

## QGIS Application - Bug report #996

### Re: [Quantum GIS] #996: QGIS corrupts JPEG2000 colors

2008-03-17 07:09 AM - coatman -

<b>Status:</b> Closed	
<b>Priority:</b> Low	
<b>Assignee:</b> nobody -	
<b>Category:</b> Rasters	
<b>Affected QGIS version:</b>	<b>Regression?:</b> No
<b>Operating System:</b> All	<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b>	<b>Resolution:</b> fixed
<b>Crashes QGIS or corrupts data:</b>	<b>Copied to github as #:</b> 11056
<b>Description</b>	
<p>Under Mac OS X, working with QGIS version 0.9.2, GDAL version 1.5.0, and Kakadu version 6.0, I built the gdal_JP2KAK.dylib Kakadu-based plugin that enables QGIS to read in JPEG2000 images. My web page <a href="http://homepage.mac.com/gregcoats/jp2.html">http://homepage.mac.com/gregcoats/jp2.html</a> shows that a simple example JPEG2000 image with red, green, blue, white, cyan, magenta, yellow, and black pixels is displayed by QGIS the same size as the original JPEG2000 image, but with drastically different colors. This is highly undesirable, and is not the fault of the Kakadu version 6.0 software. Further tests show that GDAL's gdal_translate is working properly with the gdal_JP2KAK.dylib plugin. A previous version of the gdal_JP2KAK.dylib, used with earlier versions of QGIS, GDAL, and Kakadu, worked fine. So, the problem is with the new version of QGIS.</p>	

#### History

##### #1 - 2008-05-02 12:46 PM - Frank Warmerdam -

Greg,

Could you point me to the actual .jp2 file demonstrating the problem? I reviewed the jp2.html page you link to, but it did not appear to have the actual .jp2 file.

##### #2 - 2008-05-03 08:38 PM - starriver -

Please also look at ticket #1055 (Contrast Enhancement defaults to Stretch to [[MinMax]] causing raster color shifts) as it may be the same issue.

##### #3 - 2008-05-05 11:20 AM - ersts -

I cannot reproduce this on my Linux box with the current version of QGIS and GDAL 1.4.4 or 1.5.0 and the jp2 provided at the URL above.

I have not yet tried on a Windows box.

##### #4 - 2008-05-06 08:16 PM - coatman -

My web site <http://homepage.mac.com/gregcoats/jp2.html> offers a good for testing, small example [[GeoJPEG]]2000 image only 800x100 pixels, that contains red, green, blue, white, cyan, magenta, yellow, and black pixels. Please note that QGIS 0.8.0 worked well with gdal\_JP2KAK.dylib. It is QGIS 0.9.2 that radically, unacceptably alters the colors of all [[GeoJPEG]]2000 images when using gdal\_JP2KAK.dylib. Also, note that I was told that gdal\_JP2KAK.dylib must be the only entry in /Library/Application Support/GDAL/Plugins in order for the Kakadu based gdal\_JP2KAK.dylib plugin to actually be used.

The reference to "the current version of QGIS" is not helpful because it does not tell us what version of QGIS was tried. I initially reported that QGIS 0.8.0

succeeds, and QGIS 0.9.2rc1-Ganymede (8079) fails. This was confirmed by William Kyngesburye before I posted the initial report, and since then has been confirmed by starriver.

It is frustrating that this worked fine in QGIS 0.8.0, but later something was changed, and not checked, that prevents [[GeoJPEG]]2000 images from being properly displayed.

**#5 - 2008-05-06 08:26 PM - coatman -**

When adding a raster layer, QGIS 0.9.2 radically changes of colors of [[GeoJP]]2 images imported with gdal\_JP2KAK.dylib, but QGIS properly imports [[GeoTIF]] images.

**#6 - 2008-05-06 08:47 PM - ersts -**

Sorry for the confusion. By the "the current version of QGIS" on 2008-05-05 I was referring to the current version of the trunk that I tested, revision 8396, and I used the file from the provided URL:

[http://homepage.mac.com/gregcoats/jp2/images/rgbwcmk01\\_YeGeo.jp2](http://homepage.mac.com/gregcoats/jp2/images/rgbwcmk01_YeGeo.jp2)

rgbwcmk01\_YeGeo.jp2 loaded and looked correct on Linux (Ubuntu 8.04) with both GDAL 1.4.4 and 1.5.0. I have not tried on a Window or a Mac machine yet.

**#7 - 2008-05-06 09:40 PM - William Kyngesburye**

One additional test: I built the GDAL Kakadu plugin for GDAL 1.5.1 from Kakadu 5.0 - same problem. I'm running Qgis 0.10.0.

**#8 - 2008-05-06 09:50 PM - coatman -**

From <http://www.kyngchaos.com/wiki/software:qgis> I downloaded, installed, and then ran QGIS version 0.10.0-10 revision 8375 under Mac OS X. I observe no change & no improvement. As before, revision 8375 properly displays [[GeoTIF]] images, but it dramatically alters the colors of [[GeoJP]]2 files. QGIS version 0.8.0 works fine with the same files.

**#9 - 2008-05-07 08:59 AM - William Kyngesburye**

Well, I think I found the problem - Apple's GCC in Xcode 3.x has some optimization bugs, I've worked around a couple already in GDAL and [[PostGIS]], now it hits Qgis. A new Xcode 3.1 beta was released yesterday, so I'll first try that to see if Apple fixed it. Otherwise I'll have to build an un- or minimally-optimized Qgis. Unless I can figure out how to get the cmake system to apply different CXXFLAGS to certain sources (the raster reading), I'll have to apply the same optimization to all of Qgis.

**#10 - 2008-05-07 04:38 PM - William Kyngesburye**

Scrap that. I was still waking up I guess, and had the wrong GDAL-JP2 plugin enabled.

**#11 - 2008-06-08 06:28 PM - ersts -**

- Resolution set to invalid
- Status changed from Open to Closed

On Ubuntu 8.04 with GDAL 1.4 (using Jasper 1.9) the colors are correct for this image. As this color issue is also a problem in FWTools2.1 which is built with kakadu, I have to conclude that the issue has to do with either kakadu and/or GDAL and not QGIS.

As this is not an bug introduced by QGIS, I am closing this ticket and hope the devs from kakadu & GDAL will be able to solve this problem.

**#12 - 2008-06-08 06:42 PM - coatman -**

The point is that the current version of QGIS, downloaded from the official QGIS web site, does not properly display JPEG2000 images! How can that be ignored?

This is visually demonstrated on my web site

<http://homepage.mac.com/gregcoats/jp2.html>

QGIS 0.8.0 properly displayed all JPEG2000 images with both Jasper and Kakadu. That was great!

But no version of QGIS released since version 0.8.0 properly displays JPEG2000 images with Jasper nor Kakadu.

So why isn't that a problem with QGIS?

Have you viewed the details on my web page?

<http://homepage.mac.com/gregcoats/jp2.html>

Have you worked with any of the 5 example JP2 images on my web page?

Have we waited 3 months to be told that the problem clearly, visually demonstrated on

<http://homepage.mac.com/gregcoats/jp2.html>

does not exists?

**#13 - 2008-06-08 07:36 PM - coatman -**

- Resolution deleted (invalid)

- Status changed from Closed to Feedback

The problem is that QGIS 0.8.0 properly displayed JPEG2000 images. Since then, neither Jasper nor Kakadu have changed, but no version of QGIS since 0.8.0 properly displays JPEG2000 images.

All the visual evidence needed is here

<http://homepage.mac.com/gregcoats/jp2.html>

for all to see.

Again, even the official release of the latest version of QGIS for Mac OS X from the QGIS web site does not properly display JPEG2000 images. Have you downloaded any of the 5 sample JP2 images, that I was asked to create and make available for testing, and that clearly show QGIS failures since version 0.8.0?

**#14 - 2008-07-08 03:58 PM - Tim Sutton**

Coatman could you confirm whether you are testing with Toms 'All in one' or Williams 'frameworks' based application bundle on Mac? Please enumerate exact bundle source for both the QGIS 0.8 and later versions you used. This seems to be a packaging issue rather than an issue in core of QGIS. I tested here under linux and the images loaded properly. Once we have identified exactly whose packages, we should reassign this to the appropriate packager (I see William has already participated in this thread).

I'm keeping this milestone for 1.0 for the time being since we need to get 0.11.0 released asap.

**#15 - 2008-07-08 05:22 PM - William Kyngesburye**

I don't think it's a packaging problem - I compiled a kakadu v5 plugin for Tim's GDAL and his Qgis has the same color issue. My old Qgis 0.8 that Greg says works, also used an older GDAL (1.4 I think), so 2 items have changed here. I tested with the same older Kakadu, though Greg has the latest v6.

**#16 - 2008-07-08 08:36 PM - coatman -**

I do not think this is a packaging problem. There is very substantial information posted on my web page <http://homepage.mac.com/gregcoats/jp2.html> including 7 sample pairs of JPEG2000 images, showing how they are improperly displayed by QGIS and how they appear when properly displayed. QGIS version 0.8.0 properly displayed JPEG2000 images via `[[JasPer]]` and Kakadu. The current QGIS does not properly display JPEG2000 when relying upon `[[JasPer]]` nor when relying upon Kakadu. There is nothing wrong with `[[JasPer]]` or Kakadu.

**#17 - 2008-07-09 01:23 AM - Tim Sutton**

Hi. Yes before making my previous comment, I read through your web page <http://homepage.mac.com/gregcoats/jp2.html> in detail (great presentation of the issue by the way!), tested each of the available image and they all displayed identically in both tif and jp2 variants under QGIS svn trunk on ubuntu heron using the following packages:

ii libgdal1-1.4.0	1.4.4-1ubuntu3	Geospatial Data Abstraction Library
ii libjasper1	1.900.1-3	The <code>[[JasPer]]</code> JPEG-2000 runtime library
ii libqt4-core	4.4.0-1ubuntu3~hardy1	transitional package for Qt 4

I could not replicate the issue in any case. This leads me to believe that one of the dependencies being included with the QGIS package on other platforms has changed and is causing the issue.

What license are these images under? If you agree I will create a unit test to compare the images and we can formalise the testing a little more. I think we should ask Marco Pasetti (windows packager) to do some testing with his most recent packages too so we can see what the status is with his windows packages before we put out the release.

Regards

Tim

**#18 - 2008-07-09 05:23 AM - coatman -**

You are welcome to use the images available on my web page to further diagnose the problem. I interpret your comments to mean that you successfully displayed those images with a new version of QGIS depending upon the `[[JasPer]]` JPEG2000 code, but did not test QGIS depending upon Kakadu to display JPEG2000 images.

**#19 - 2008-07-09 07:23 AM - Tim Sutton**

Hi

Ah excuse my ignorance - I hadn't realised that jasper and kakadu were different implementations - I will try to install kakadu jp2 support on ubuntu if packages are available and retest.

I will create a unit test using your images that will run conditionally on presence of jp2 support.

I'm also not aware of how to tell gdal to use kakadu rather than jasper for jp2 support so I'll need to investigate this all further.

Peter on your ubuntu testing were you using Jasper or Kakadu?

Regards

Tim

**#20 - 2008-07-09 07:35 AM - William Kyngesburye**

Kakadu is not open/free, so you would have to get the source yourself. Though it is possible to distribute binaries based on Kakadu for open source projects. Kinda like the [[MrSID]] and ECW FOSS licenses, except that the source costs \$. Which is really a bummer, because Kakadu seems to be the only one left standing with currently-maintained and powerful JP2 support.

As for GDAL loading, GDAL load external plugins first, so if Kakadu is compiled as a plugin it will override Jasper if that is built into GDAL. The other way is to use GDAL\_SKIP env var to disable the Jasper driver.

**#21 - 2008-07-09 08:42 AM - coatman -**

The person who was working on [[GeoJasPer]] ceased development 3 years ago. Kakadu continues to be maintained, updated, and enhanced. Kakadu supports hardware acceleration via Intel SSE, and via multiple processors and multiple cores. As a practical matter, Kakadu is an order of magnitude faster than [[GeoJasPer]] and requires an order of magnitude less memory than [[GeoJasPer]]. It is because [[GeoJasPer]] was completely free and Kakadu charges a small amount that we have the current situation. GDAL (Frank Warmerdam) has supported Kakadu for 5 years. QGIS version 0.8.0 properly displayed JPEG2000 images via [[GeoJasPer]] and Kakadu! I am simply trying to suggest that we return this functionality to QGIS. The cost of the Kakadu source code for use in the non-commercial QGIS is less than the cost an Adobe Photoshop upgrade. I have offered to pay the cost of an additional Kakadu license for the person who would include in QGIS Kakadu's support for hardware acceleration.

**#22 - 2008-07-09 09:05 AM - coatman -**

The Summary / Subject got corrupted in my previous posting. Sorry.

**#23 - 2008-07-09 09:50 AM - ersts -**

On Ubuntu I used jasper because as indicated in a subsequent response, Kakadu is not free.

However, I tried the test images with binary distribution of FWTools v2.2.1 (for windows), which is built with kakadu, and it also corrupted the colors.

This is what led me to conclude that it is not an error directly introduced by QGIS, but some kinda of GDAL<->Kakadu conflict/miscommunication.

**#24 - 2008-08-02 02:35 AM - Tim Sutton**

I have removed the 'must fix' for release denotation for this bug since it requires using a non free library, and the underlying cause of the issue is unlikely to be resolved for the release. We can pursue the issue further in one of the 1.x.x bug fix releases.

My apologies since kakadu support is no doubt important to the original ticket submitter but we need to prioritise what we can reasonably achieve for the release of 1.0

Regards

Tim

**#25 - 2008-12-06 02:32 PM - coatman -**

The first several releases of QGIS properly displayed JPEG2000 and [[GeoJPEG]]2000 images, using both Jasper and Kakadu. Then QGIS was "upgraded", with the result being that QGIS no longer properly displays JPEG2000 nor [[GeoJPEG]]2000 images. Not using Jasper, and not using

Kakadu. This problem is well documented and visible at <http://homepage.mac.com/gregcoats/jp2.html>

9 months later it was simply decreed that QGIS would be released without the ability its early versions had to display JPEG2000 images. The author of Kakadu offered the Kakadu source code for free to this project. I offered to pay a programmer. So, I am at a loss to understand why nothing has been done, and why there is no plan to do anything?

I offer \$2,500.00 for a version of QGIS, compatible with Mac OS X 10.5.5 and Kakadu 6.1.1, that works essentially similar to the free application kdu\_macshow, and is released by 23 Dec 2008.

Greg Coats

**#26 - 2008-12-06 08:21 PM - Tim Sutton**

Hi Greg

As an open source project, we cannot mandate what our developers work on. In planning a release we need to look at the features & bugfixes that have been implemented up to a certain point in time and use that as the basis for a release. If a given feature / bugfix is not implemented at the point of going to release, we need to make hard decisions.

Furthermore since kakadu is not freely available there is higher than normal barrier to entry to resolving your issue. I was not aware that the kakadu author had offered a copy of his sources to QGIS for us to test with. If this is the case I would certainly be happy to test on the Linux platform as probably would Peter Ersts. However resolving the issue on the OS X platform I suspect will take additional input from William and or Frank Warmerdam. Neither myself or as far as I know, Peter Ersts have an OS X development platform to test the issue on at this stage.

Perhaps William will find your kind bug bounty offer an incentive to look at this problem further.

Once again let me state that we are not unsympathetic to your desire to have the issue fixed and it was only moved to 1.1 for pragmatic reasons rather than any desire to dismiss this ticket as a non issue.

**#27 - 2008-12-06 09:04 PM - coatman -**

Please remember that the first several versions of GDAL that were released, properly displayed JPEG2000 images under Mac OS X, using BOTH Jasper and Kakadu. I became interested in GDAL precisely because it properly displayed JPEG2000 images. William Kyngesburye has said to me several times he does not accept money to do anything under any circumstances.

There seems to be a strong bias here (that I do not understand) against Kakadu, even though the Kakadu executables have always been free to everyone, and free for all of these platforms: Windows, Linux, Solaris, and Mac OS X. Professor Taubman offered the Kakadu source code for free to this project. But even if that offer is rejected for a reason I do not understand, the cost of purchasing the Kakadu source code with a lifetime of free updates is less than the cost of buying one Photoshop upgrade, that contains executables, but no source code. And I would pay for the cost, but it is free for this project.

So, can we at least all agree to move forward now to make available a version of GDAL for Mac OS X that properly displays JPEG2000 images, using at least Jasper? Surely no one objects to that. But note that Jasper and [[GeoJasper]] were both been abandoned years ago by their authors, and the last time I compiled them I got more than 500 warnings. Kakadu compiles without any warnings, for all platforms, so it may be easier, and faster to use.

If no one will accept money for a GDAL compatible with Mac OS X, then I will have no choice but to abandon GDAL, withdraw my financial support, and give my money before the end of the year to ESRI. I do not want to do that. But before the end of the year, I need a GIS that displays JPEG2000 images, which the initial versions of GDAL did.

Please understand that I am trying to help, and just ask that a functionality that was in the first version of GDAL be restored to the current version. I am not requesting a new feature that has never been used.

#28 - 2008-12-06 10:55 PM - ersts -

Replying to [comment:27 Coatman]:

*Please remember that the first several versions of GDAL that were released, properly displayed JPEG2000 images under Mac OS X, using BOTH Jasper and Kakadu.*

As I have noted in previous posts, displaying JPEG2000 images in QGIS on Ubuntu versions <= 8.04 (with jasper) work just fine, while other platforms it did not. As of Ubuntu 8.10 that is not longer the case. So now, the JPEG2000 display errors are present on all platforms, but is not specific to QGIS ( you will also experience these display issues in FWTools [compiled with kakadu] or with gdal\_translate [native 8.10 compiled with jasper]). Thus the problem has to do with, or a combination of, libjpeg, jasper and/or kakadu, and GDAL.

*There seems to be a strong bias here (that I do not understand) against Kakadu*

That is simply not the case, it is a broader issues that has been documented with other libs several times in this and related posts.

*Please understand that I am trying to help, and just ask that a functionality that was in the first version of GDAL be restored to the current version. I am not requesting a new feature that has never been used.*

Your interest is acknowledged but your passion maybe misdirected and misunderstood. The QGIS raster capabilities are tightly wrapped around GDAL which itself relies on numerous other libraries. While we all want the best app possible, many folks contribute only as volunteers and each QGIS dependency has numerous outstanding bugs themselves; the ones which are most important to us individually ultimately seem to be the ones take the longest to fix.

I am sure in spirit of the open-source paradigm we can all contribute toward the golden goal and I openly ask for any patches and ideas folks may have that may bring a conclusion to this problem.

cheers.

#29 - 2008-12-07 11:25 AM - William Kyngesburye

Comments and news:

*... even though the Kakadu executables have always been free to everyone, and free for all of these platforms: Windows, Linux, Solaris, and Mac OS X.*

These executables are just that, executables. There are no headers, thus the library binary is not usable in other projects. As stated on the kakadu download page: they are for demonstration purposes.

*William Kyngesburye has said to me several times he does not accept money to do anything under any circumstances.*

As I've also said, it's not a question of accepting money (I can't), it's a question of skill - I'm not a C++ programmer. I can poke and prod, and have a fair chance of figuring out problems, but this one is too deep for me.

---

Anyways, I wrestled a little more with the ECW library (poking and prodding), which includes its own custom JP2 library. ECW on OSX has always been a troublemaker - getting it working at all, getting it working for universal OSX, getting it working in 64bit OSX. Up until now, all I had working was ECW

read-only, universal/64bit. At one time the ECWJP2 was working, but it broke a while back.

I just got ECWJP2 working again, but only in 32bit Intel OSX (I can't test PPC right now, but it likely also works). This is OK for Qgis, as 64bit Qgis on OSX is possible but still needs testing (and probably work). There are no color problems with Greg's test JP2 image using the ECW JP2 library.

I'm working with GDAL 1.6 right now. I can't get the ECW plugin to load in GDAL 1.5 for some reason. My current Qgis uses GDAL 1.5, and I'm waiting for GDAL 1.6 to finalize this coming week.

**#30 - 2008-12-07 01:31 PM - coatman -**

I take this as good news, because we can conclude that the present inability of QGIS to properly display JPEG2000 images has nothing to do with Mac OS X, and nothing to do with Kakadu. So it could be fixed in another Unix environment, like Ubuntu, and then it may again work with Mac OS X and Kakadu. Again, please remember that QGIS initially properly displayed JP2 images with Jasper and Kakadu. libjpeg is not involved in any way with JP2 images. Jasper has not changed in years. I can use with QGIS the same version of Kakadu, as when QGIS properly displayed JP2 images. So, the source of the problem is in a part of some code that **changed**, and that handles reading in JP2 images. Is Subversion, or something like it, available to point to these code changes?

**#31 - 2008-12-07 07:14 PM - Frank Warmerdam -**

Greg,

I'm quite lost in this ticket. I mostly just wanted to caution you with regard to purchasing ESRI software for jpeg2000 support - in that it uses GDAL (JP2KAK driver) and assuming the issues are not entirely platform dependent, you might also encounter them in the [[ArcGIS]] context.

**#32 - 2008-12-08 06:28 PM - coatman -**

Thanks for the warning. So, as a test, today I went to a place where I was able to use ESRI [[ArcGIS]] version 9.2 and 9.3. Both versions properly displayed all of the [[GeoJPEG]]2000 .jp2 files I had.

Would someone please offer an estimate as to when QGIS will be able to properly display JPEG2000 images, under Ubuntu and Mac OS X, as it was originally did?

Greg

**#33 - 2008-12-09 01:10 PM - coatman -**

The original QGIS properly displayed JPEG2000 images with Kakadu and with Jasper. This was fantastic. Since an update, QGIS no longer properly displays JPEG2000 images with Kakadu nor with Jasper. This is not because of a bug in Kakadu. And since Ubuntu exhibits the same problem, this is not because of a bug in Mac OS X. My web page <http://homepage.mac.com/gregcoats/jp2.html> shows that QGIS displays JP2 images with the correct number of columns, and the correct number of rows. Only the colors are improperly displayed. I have waited 9 months, and I have offered to donate \$2,500.00 for a version of QGIS that displays JP2 images. Will someone please inform me when I can anticipate having a version of QGIS that again displays JP2 images with the proper colors?

**#34 - 2008-12-10 07:14 AM - Tim Sutton**

Hi

Could I repeat my request to make the source code of kakadu available so that we can test. In an earlier post you mentioned a bias against kakadu - while



I am not aware of any overt bias, not having ready access to the sources so that we can compile kakadu enabled versions of gdal makes it difficult to test.

Thanks

Regards

Tim

**#35 - 2008-12-10 08:29 AM - coatman -**

The tests run, the results shown on my web page [homepage.mac.com/gregcoats/jp2.html](http://homepage.mac.com/gregcoats/jp2.html) and the comments above by ersts and kyngchaos show that the current QGIS does not properly display JPEG2000 images using Jasper, nor does the current QGIS properly display JPEG2000 images using Kakadu, under (at least) Ubuntu 8.10 and Mac OS X 10.4 and 10.5. Since this team is more familiar with Jasper, I respectfully suggest that the first step is to get QGIS properly displaying JPEG2000 images with Jasper. Then I could then see if fixing QGIS with Jasper results in QGIS also being fixed with Kakadu.

Frank and I already have the Kakadu source code. I offered to pay for a copy for William, but he is unable to accept money. I paid for my copy of Kakadu. Professor Taubman has offered to supply an additional copy of Kakadu, free, and I have relayed that offer here, but no one has asked me for this additional copy of Kakadu, so I have been unable to pass along a request with contact info to the Kakadu licensing folks in Australia.

It seems that the best approach now is to get QGIS properly displaying JPEG2000 images with Jasper, and after that I would personally take on the task of at least seeing if QGIS would properly displays JPEG2000 images with Kakadu. Plus if QGIS properly displayed JP2 images with Jasper but not with Kakadu, then I could get the Kakadu group to investigate and fix that situation. But it is essentially a prerequisite that we have QGIS properly displaying JP2 images with Jasper.

QGIS is currently displaying [[GeoJP]]2 images properly georeferenced, with the proper number of columns, and the proper number of rows, but with altered colors. So it is reasonable to think that since displaying JP2 images with both Jasper and Kakadu exhibits the same altered colors symptoms, a change that works with Jasper will also likely work with Kakadu.

The altered colors are not in [[ArcGIS]], nor in the early versions of QGIS. Since I have waited 9 months, and recently offered money, I do not know what else I can do, particularly in this forum. I would be glad to talk more directly with anyone interested in a resolution. My email is [gregcoats@mac.com](mailto:gregcoats@mac.com) I can telephone any one, any where, and am available for an iChat with audio or audio+video. Without some resolution of the displaying altered colors, the only solution I see is buying [[ArcGIS]].

**#36 - 2008-12-10 10:00 AM - Jürgen Fischer**

Yesterday I compiled GDAL 1.4.3 and GDAL 1.5.2 (both from debian package sources I had lying around) with jasper 1.9 and tried to convert the first jp2 from the page to tif using `gdal_translate` from either gdal versions.

Findings:

- `qiv` from jasper displays the correct colors from the jp2
- the tif from gdal 1.4.3 also has the correct colors (tested with `qiv`),
- the tif from gdal 1.5.2 is all black (also tested with `qiv`)

So GDAL seems to be the problem and neither QGIS nor jasper/kakadu. I didn't compile qgis with gdal 1.4 yet, but QGIS with GDAL 1.5 also shows the jp2 all black.

Just my 2c.

**#37 - 2008-12-10 10:04 AM - Frank Warmerdam -**

Jef / Greg,

A focused GDAL ticket should likely be filed on this. Note, I don't maintain the [[JasPer]] driver, so if you approach it from that angle it will be up to Andrey Kiselev to decide if he is interested in pursuing it. If it is reproduced with the JP2KAK driver, I can try to look into it.

**#38 - 2008-12-10 10:12 AM - William Kyngesburye**

The [[JasPer]] and Kakadu problems appear to be different.

For [[JasPer]], it consistently appears in anything based on GDAL. I have a bug report for that:

<http://trac.osgeo.org/gdal/ticket/2399>

The Kakadu problem appears in some particular way of using GDAL - here in Qgis, and in FWTools as noted by ersts. Other direct use in the GDAL tools is not affected.

**#39 - 2008-12-10 10:18 AM - coatman -**

Jef, Thanks for your helpful report. I think it is reasonable for us to conclude that Jasper and Kakadu properly read, write, and display JPEG2000 images. I think that QGIS depends upon GDAL to display JP2 images, so there may be no QGIS code to change to get QGIS to display JP2 images with the correct colors.

**#40 - 2008-12-10 10:27 AM - coatman -**

Hi Frank,

Thanks for your suggestion.

I would appreciate it if you would please suggest where to go to file a focused GDAL ticket.

**#41 - 2008-12-10 11:09 AM - coatman -**

Doing a Unix diff on frmts/jp2kak/jp2kakdataset.cpp for gdal-1.6.0 (a version of GDAL that I think is displaying altered colors for JP2 images with JP2KAK) versus gdal-1.5.0 (a version of GDAL that I think properly displayed colors in JP2 images with JP2KAK), I notice that in the section following the line 312 comment "Figure out the color interpretation for this band." gdal-1.5.0 provides two possibilities

```
    if( oJP2Channels.get_num_colours() 3 )    {
#ifdef KAKADU41
    oJP2Channels.get_colour_mapping( 0, nRedIndex, nLutIndex, nCSI );
    oJP2Channels.get_colour_mapping( 1, nGreenIndex, nLutIndex, nCSI );
    oJP2Channels.get_colour_mapping( 2, nBlueIndex, nLutIndex, nCSI );
#else
    oJP2Channels.get_colour_mapping( 0, nRedIndex, nLutIndex );
    oJP2Channels.get_colour_mapping( 1, nGreenIndex, nLutIndex );
    oJP2Channels.get_colour_mapping( 2, nBlueIndex, nLutIndex );
#endif

```

while gdal-1.6.0 provides only one possibility

```
    if( oJP2Channels.get_num_colours() 3 )    {
    oJP2Channels.get_colour_mapping( 0, nRedIndex, nLutIndex, nCSI );
    oJP2Channels.get_colour_mapping( 1, nGreenIndex, nLutIndex, nCSI );

```

```
oJP2Channels.get_colour_mapping( 2, nBlueIndex, nLutIndex, nCSI );
```

Since what I see as a result of QGIS displaying a JP2 image with JP2KAK, is an image with the proper number of columns, and proper number of rows, but with the colors altered as through they are being run through a color Look Up Table or Lut, I thought this observation worth reporting.

**#42 - 2008-12-10 02:17 PM - Frank Warmerdam -**

Replying to [comment:40 Coatman]:

*Hi Frank,*

*Thanks for your suggestion.*

*I would appreciate it if you would please suggest where to go to file a focused GDAL ticket.*

At the risk of being flippant, a focused GDAL ticket should be filed in the GDAL bug tracker. (<http://trac.osgeo.org/gdal>).

It needs to demonstrate the problem with GDAL utilities or some small sample program using the API. As there is already a ticket languishing on the [[JasPer]] issue there is no point in filing another. If you were to file one on the JP2KAK problem, it would need to be based on something reproducible with GDAL utilities (ie. `gdal_translate`). It appears from the comments above that this is not so easy.

I do think the color mapping stuff is a promising angle, though it isn't particularly clear to me why it would work for `gdal_translate`, and not for `qgis`.

**#43 - 2008-12-10 04:03 PM - Jürgen Fischer**

Replying to [comment:36 jef]:

*I didn't compile qgis with gdal 1.4 yet, but QGIS with GDAL 1.5 also shows the jp2 all black.*

just to complete the findings: `qgis` compiled with `gdal 1.4` display the sample `jp2` ok.

**#44 - 2008-12-10 06:28 PM - coatman -**

Jef, Thanks for reporting your findings. I think it is now reasonable to conclude by a preponderance of the evidence reported here that

1. When QGIS depends upon a version of GDAL that is older than about 10 months, then QGIS properly displays JPEG2000 images thru Kakadu and thru Jasper
2. When QGIS depends upon a version of GDAL that is newer than about 10 months, JP2 images are properly georeferenced, and displayed with the proper number of columns and rows, but the colors are altered. Thru Kakadu, the colors appear to be run through a color Look Up Table (LUT). Thru Jasper, the colors can be all black. More examples are at <http://homepage.mac.com/gregcoats/jp2.html>

In comparing the code in GDAL 1.6.0 to 1.5.0, I find that the Kakadu based `frmts/jp2kak/jp2kakdataset.cpp` in 1.6.0 has different code than 1.5.0 after the comment in 1.6.0 on line 312 "Figure out the color interpretation for this band."

I find that the Jasper based `frmts/jpeg2000/jpeg2000dataset.cpp` in 1.6.0 has different code than 1.5.0 around the comment in 1.6.0 on line 738 "JPEG2000 driver ignores color table. Consider using color table expansion (-expand option in `gdal_translate`)".

Jef, If you do not already have Kakadu and are interested (Kakadu is much faster than Jasper), then please send contact info to [gregcoats@mac.com](mailto:gregcoats@mac.com)

Thanks to everyone for their contributions. I hope now someone will look into GDAL.

**#45 - 2008-12-10 09:28 PM - Frank Warmerdam -**

Greg,

Note: I'm still waiting for a GDAL ticket to be filed. It should include all relevant details, including how to reproduce the problem with GDAL utilities, what version of GDAL works, what one does not. What version of Kakadu you are using to establish this. And of course how you are determining success and failure.

At that point I'll briefly try to reproduce what you report and see where we go next.

**#46 - 2008-12-11 07:08 AM - coatman -**

Hi Frank, I posted a focused GDAL ticket about 3 hours before this message from you. It is titled JPEG2000 images improperly displayed, colors corrupted and the URL is <http://trac.osgeo.org/gdal/ticket/2732>

More detailed information, and most importantly small JP2 images created specifically for testing with this ticket are available for free download from <http://homepage.mac.com/gregcoats/jp2.html>. Success is properly displaying these JP2 test images. Failure is improperly displaying these test images, especially displaying these JP2 images with altered colors, and in particular changing the colors of these JP2 test images by running them through a color Look Up Table (LUT).

Regarding Kakadu version: The original report posted here at May 2, 2008 3:46:51 PM EDT specified in its first line that we were using Kakadu version 6.0, the current version of Kakadu at that time. As of 8 December 2008, the current version of Kakadu is 6.1.1, which has several improvements and no known, reported bugs, so I would recommend working with Kakadu 6.1.1, but a solution with either version of Kakadu would be most welcome. Thanks for your help.

**#47 - 2008-12-11 07:54 AM - coatman -**

Frank writes: "please reference one specific jpeg2000 file demonstrating a problem, and ideally a reference tiff or png demonstrating what the image should look like."

As I have said here what seems like many times since 2 May 2008, several image files are available for testing at, and can be downloaded, for free, from <http://homepage.mac.com/gregcoats/jp2.html>

All of these image files are available as .tif, .jp2, and .png files. I believe this more than meets the requirement to reference one online image that demonstrates the problem. Please move forward.

**#48 - 2008-12-11 08:34 AM - coatman -**

Perhaps being more specific will help.

As an aid to narrowly focusing this problem, I am making available as a free download from <http://homepage.mac.com/gregcoats/jp2.html> an 800x100 pixel and a 313x313 pixel .jp2 files showing [[GeoJPEG]]2000 images properly georeferenced, and displayed with the proper number of columns, and the proper number of rows, but with the colors altered, as though they are being run through a Color Look Up Table, that are easily visually compared with these same images as .tif files that are properly displayed [[GeoTIFF]] images, alone with a reference, non-georeferenced, .png file.

**#49 - 2008-12-11 10:08 AM - coatman -**

I am new to Trac, and warmly welcome contributions from others who could better explain & focus this ticket.

Available at <http://homepage.mac.com/gregcoats/jp2.html> for testing, is a 800x100 image that is a composite of squares with (from left to right) pure red, green, blue, white, cyan, magenta, yellow, and black pixels. Here are the direct URLs for the 800 columns x 100 rows [[GeoJP]]2, [[GeoTIF]], and PNG files.

[http://homepage.mac.com/gregcoats/jp2/images/rgbwcmkyk01\\_YeGeo.jp2](http://homepage.mac.com/gregcoats/jp2/images/rgbwcmkyk01_YeGeo.jp2)

[http://homepage.mac.com/gregcoats/jp2/images/rgbwcmkyk01\\_YeGeo.tif](http://homepage.mac.com/gregcoats/jp2/images/rgbwcmkyk01_YeGeo.tif)

<http://homepage.mac.com/gregcoats/jp2/images/rgbwcmkyk01.png>

Also available is a 313x313 natural color aerial photograph. Here are the direct URLs for the 313 columns x 313 rows [[GeoJP]]2, [[GeoTIF]], and PNG files.

[http://homepage.mac.com/gregcoats/jp2/images/18stj940125\\_reduce4\\_reversible.jp2](http://homepage.mac.com/gregcoats/jp2/images/18stj940125_reduce4_reversible.jp2)

[http://homepage.mac.com/gregcoats/jp2/images/18stj940125\\_reduce4.tif](http://homepage.mac.com/gregcoats/jp2/images/18stj940125_reduce4.tif)

[http://homepage.mac.com/gregcoats/jp2/images/18stj940125\\_reduce4.tif.png](http://homepage.mac.com/gregcoats/jp2/images/18stj940125_reduce4.tif.png)

**#50 - 2009-01-01 12:32 PM - coatman -**

GDAL supports JPEG2000 images using either Kakadu based JP2KAK or Jasper based JPEG2000

`gdal_translate --version`

GDAL 1.6.0, released 2008/12/04

`gdal_translate --formats | grep 2000`

JP2KAK (rw): JPEG-2000 (based on Kakadu)

JPEG2000 (rw): JPEG-2000 part 1 (ISO/IEC 15444-1)

When writing a JPEG2000 image, GDAL `gdal_translate` lets the user specify `-of JP2KAK` to use Kakadu, and lets the user specify `-of JPEG2000` to use Jasper. When reading a JPEG2000 image, GDAL lets the user specify `export GDAL_SKIP=JPEG2000` to not use Jasper, and instead use Kakadu; and GDAL lets the user specify `export GDAL_SKIP=JP2KAK` to not use Kakadu, and instead use Jasper.

Even though I specified use Kakadu by setting the environment variable `export GDAL_SKIP=JPEG2000`, in QGIS, after adding a JPEG2000 image via Layer -> Add Raster Layer..., I find that Layer -> Properties... -> Raster Layer Properties -> Metadata shows that QGIS is using Jasper based JPEG2000, instead of using Kakadu based JP2KAK. QGIS initially shipped properly displaying JPEG2000 images using Jasper, but for the past 10 months QGIS has not properly displayed JPEG2000 images using Jasper, and there is no effort to re-enable QGIS to use Jasper to properly display JPEG2000 images.

So, I desperately need a way to specify to QGIS that it use the GDAL Kakadu based JP2KAK rather than the Jasper based JPEG2000. Since I do not think QGIS pays attention to the environment variable `GDAL_SKIP`,

How do I tell QGIS to use the GDAL Kakadu based JP2KAK, rather than Jasper, to display JPEG2000 images?

**#51 - 2009-01-01 06:25 PM - coatman -**

The Quantum GIS Downloads web page offers 2 flavors of QGIS version 1.0.0-Kore-preview2 for Mac OS X. I have downloaded, installed, and used both. Unfortunately, both improperly display corrupt colors for JPEG2000 images, that are properly geo-referenced, and displayed with the proper number of columns and rows. This serious problem was reported here 10 months ago, and is extremely well documented at this web site

<http://homepage.mac.com/gregcoats/jp2.html>

I was delighted that early releases of QGIS properly displayed JPEG2000 images. I strongly urge QGIS management to halt its practice of knowingly releasing versions of QGIS that display corrupt colors for JPEG2000 images, and to quickly restore to QGIS the ability to properly display the colors in JPEG2000 images. I have waited 10 months, and I have offered \$2,500.00 for a version of QGIS, compatible with Mac OS X 10.5.5 and Kakadu 6.1.1, that would, like the early versions of QGIS, simply properly display the colors in JPEG2000 images, but QGIS remains unable to properly display JPEG2000 images.

**#52 - 2009-01-03 12:49 PM - coatman -**

Using the GDAL Jasper-based JPEG2000, and without using Kakadu, I used GDAL 1.6 `gdal_translate` to create from a `[[GeoTIF]]` a `[[GeoJP]]2`, and then to create from that `[[GeoJP]]2` a `[[GeoTIF]]`. QGIS version 1.0.0-Kore-preview2, for Mac OS X 10.5, available from the Quantum GIS Downloads web site, and from the kyngchaos web site, both display the `[[GeoTIF]]` created by `gdal_translate` properly geo-referenced, with the proper number of columns and rows, but with the colors being ALL BLACK. Additionally, all of the other image viewing applications that I have display the `[[GeoTIF]]` created by `gdal_translate` as all black.

To summarize, I am reporting that a *GeoTIF*, created by `gdal_translate` from a `[[GeoJP]]2` that `gdal_translate` created, is being displayed as all black, by QGIS and by 5 other image viewing applications.

The sample `[[GeoTIF]]`, 800 columns by 100 rows, that `gdal_translate` converts to `[[GeoJP]]2`, can be downloaded from [http://homepage.mac.com/gregcoats/jp2/images/rgbwcmk01\\_YeGeo.tif](http://homepage.mac.com/gregcoats/jp2/images/rgbwcmk01_YeGeo.tif)

```
gdal_translate --version
```

```
GDAL 1.6.0, released 2008/12/04
```

```
gdal_translate --formats | grep 2000
```

```
JP2KAK (rw): JPEG-2000 (based on Kakadu)
```

```
  JPEG2000 (rw): JPEG-2000 part 1 (ISO/IEC 15444-1)
```

```
gdal_translate -ot Byte -of JPEG2000 -co "FORMAT=JP2" -co "mode=int" rgbwcmk01_YeGeo.tif rgbwcmk01_YeGeo_GDAL16_JPEG2000.jp2
```

```
Input file size is 800, 100
```

```
0...10...20...30...40...50...60...70...80...90...100 - done.
```

```
export GDAL_SKIP=JP2KAK
```

```
gdal_translate -ot Byte -of GTiff -co "COMPRESS=NONE" rgbwcmk01_YeGeo_GDAL16_JPEG2000.jp2 rgbwcmk01_YeGeo_GDAL16_JPEG2000.tif
```

```
Input file size is 800, 100
```

```
0...10...20...30...40...50...60...70...80...90...100 - done.
```

**#53 - 2009-01-03 02:51 PM - Jürgen Fischer**

Replying to [comment:52 Coatman]:

*To summarize, I am reporting that a GeoTIF, created by gdal\_translate from a [[GeoJP]]2 that gdal\_translate created, is being displayed as all black, by QGIS and by 5 other image viewing applications.*

So you also have verified that this is a GDAL and no QGIS bug. Therefore I'm tempted to close this ticket.

Please follow up the ticket you filed in the GDAL TRAC. Your latest comment looks like the explanation Frank Warmerdam requested there and I bet the GDAL developers are following their TRAC more closely than ours.

**#54 - 2009-01-03 03:15 PM - coatman -**

A byte by byte comparison of the original uncompressed .tif to the uncompressed .tif created by `gdal_translate` using Jasper, from a .jp2 created by `gdal_translate` from the original .tif, shows that the basic tiffinfo and listgeo metadata is identical (this is good). But od shows that this 800 column by 100

row by RGB image has  $800 * 100 = 80,000$  consecutive 00 00 00 pixel triples, representing black, which results in image viewing applications displaying the .tif created by gdal\_translate as all black.

**#55 - 2009-01-03 05:52 PM - coatman -**

When gdal\_translate depends upon Jasper based -of JPEG000 to read a .tif, write a .jp2, and then write a .tif, the final .tif contains all black pixels, but when gdal\_translate depends upon Kakadu based -of JP2KAK to read a .tif, write a .jp2, and then write a .tif, the final .tif is properly written and properly displays with all of the image viewing applications I have. My GDAL has both Jasper JPEG2000 and Kakadu JP2KAK. With gdal\_translate, when writing a .jp2, the user specifies the use of Jasper or Kakadu with -of JPEG2000 or -of JP2KAK. With gdal\_translate, when reading a .jp2, the user specifies the use of Jasper or Kakadu with export GDAL\_SKIP=JP2KAK or export GDAL\_SKIP=JPEG2000.

I could use QGIS now if I knew how to specify to QGIS that it use Kakadu JP2KAK instead of using the default Jasper JPEG2000. Since gdal\_translate Jasper JPEG2000 is not working, but gdal\_translate Kakadu JP2KAK is working, P-L-E-A-S-E tell me

How to specify that QGIS use Kakadu JP2KAK, and not use Jasper JPEG2000?

Greg

```
gdal_translate -ot Byte -of JPEG2000 -co "mode=int" rgbwcmk01_YeGeo.tif rgbwcmk01_YeGeo_GDAL16_JPEG2000.jp2
```

```
export GDAL_SKIP=JP2KAK
```

```
gdal_translate -ot Byte -of GTiff -co "COMPRESS=NONE" rgbwcmk01_YeGeo_GDAL16_JPEG2000.jp2 rgbwcmk01_YeGeo_GDAL16_JPEG2000.tif
```

The resulting, Jasper dependent, .tif contains only all black pixels.

```
gdal_translate -ot Byte -of JP2KAK -co "QUALITY=100" rgbwcmk01_YeGeo.tif rgbwcmk01_YeGeo_GDAL16_JP2KAK.jp2
```

```
export GDAL_SKIP=JPEG2000
```

```
gdal_translate -ot Byte -of GTiff -co "COMPRESS=NONE" rgbwcmk01_YeGeo_GDAL16_JP2KAK.jp2 rgbwcmk01_YeGeo_GDAL16_JP2KAK.tif
```

The resulting, Kakadu dependent, .tif properly displays.

**#56 - 2009-01-04 06:28 AM - Jürgen Fischer**

- Resolution set to fixed

- Status changed from Feedback to Closed

The jasper support in GDAL was fixed (see [GDAL Ticket #2399](#)) and updated GDAL package for Ubuntu is available on [launchpad](#).

I filed the new ticket #1474 with a request to make the gdal driver selectable.

**#57 - 2009-01-05 08:31 AM - Jürgen Fischer**

Replying to [comment:56 jef]:

The jasper support in GDAL was fixed (see [GDAL Ticket #2399](#)) and updated GDAL package for Ubuntu is available on [launchpad](#).

I filed the new ticket #1474 with a request to make the gdal driver selectable.

Which was a duplicate of #182

**#58 - 2009-01-05 07:10 PM - coatman -**

Regarding <http://trac.osgeo.org/gdal/ticket/2732#comment:14>

The color display problem with GDAL Kakadu based JP2KAK is fixed by changing line 1562 of jp2kakdataset.cpp from

```
KDU_WANT_CODESTREAM_COMPONENTS);
```

to

```
KDU_WANT_OUTPUT_COMPONENTS);
```

I am pleased to report that, under Mac OS X 10.5, I changed, as directed, line 1562 of 1.6/gdal/frmts/jp2kak/jp2kakdataset.cpp and then built a new gdal\_JP2KAK.dylib plugin. Now QGIS, using Kakadu based JP2KAK, properly displays JPEG2000 images.

So a new compilation of QGIS with the updated GDAL Kakadu based JP2KAK jp2kakdataset.cpp and the updated GDAL Jasper based JPEG2000 jpeg2000dataset.cpp provides two facilities for QGIS to properly display JPEG2000 images. The JP2KAK plugin for 32 bit and 64 bit Intel Macs can be downloaded from my web site. Hooray! Thanks!

**#59 - 2009-08-22 01:01 AM - Anonymous**

Milestone Version 1.0.1 deleted