

User Manual for Kakadu's "kdu_show" Tool

(last updated for version v4.1)

David Taubman, UNSW

February 10, 2004

1 Basic Operations

1.1 Opening and Closing Files

The File menu may be used to open and close files. Alternatively, you may find the "o" accelerator key useful for opening files quickly. Opening a new file automatically closes any existing open file. You can supply the name of a file to "*kdu-show*" on the command line, in which case that file will be opened on start up and its dimensions will guide the initial size of the view window.

The "*kdu-show*" tool can open JP2 files, JPX files and unwrapped JPEG2000 codestreams. At some point in the future, MJ2 files will most likely be added to this list.

1.2 Panning, Zoom and Re-orienting

The "*kdu-show*" tool started out as a sophisticated demonstration application to showcase some of Kakadu's capabilities for incremental decompression of large images, based on spatial regions of interest (view ports), as well as Kakadu's ability to decompress images in a geometrically transformed domain. In fact, Kakadu's codestream management architecture has always allowed compressed JPEG2000 images to be accessed as though they had been compressed as smaller image, rotated and/or cropped versions of themselves. As a result, the core "kdu-show" features have always been those of panning, zooming, rotating and flipping a viewport into the compressed image.

Panning may be accomplished via the scroll-bars, or using the arrow keys. The PageUP and PageDown keys may also be used for panning. The image may also be dragged around with the mouse while the left mouse button is depressed and the shift key held down.

Zooming may be accomplished via the View menu, or with the "z" (zoom in) and "ctrl-z" (zoom out) accelerators. By default, zooming is centred about the middle of the current viewport, but you may define an alternate focus box using the method described below. Zooming out is accomplished using the multi-resolution attributes of the discrete wavelet transform (DWT). As such,

the degree to which you can zoom out is limited by the number of DWT levels created when the image was compressed. Although this limit could readily be circumvented by rescaling the decompressed image, it provides useful feedback to content creators, making them aware of the limitations associated with using too few DWT levels.

Rotation and flipping may be accomplished via the View menu, or with the “[”, “]”, “_” and “|” accelerators. Note, however, that any rotation which involves image transposition (not just horizontal and vertical flipping) may slightly alter the displayed imagery if it happens to have been compressed using JPEG2000’s reversible processing path (Kakadu’s “Creversible=yes” attribute). The reason for this is that horizontal and vertical aspects of the wavelet transform do not completely commute when the reversible transform is used, due to LSB rounding effects. Kakadu accomplishes rotation by decompressing the imagery in a reverse order and performing the wavelet transform in an alternate order, rather than just decompressing the image and then subsequently reorienting it.

The application automatically resizes its window to match the image size, unless the image is too large to be viewed in the current window, in which case a restricted viewport is displayed. You may adjust the window size via the usual resize handles. Alternatively, and often more conveniently, you may enlarge or shrink the view window quickly using the “w” and “s” accelerator keys.

1.3 Focus Boxes

Focus boxes have at least 3 uses in “kdu-show”: 1) they define a centre for zooming; 2) they define the region of interest during remote image browsing with JPIP (see Section 3); and 3) they define spatial regions to be associated with metadata added using the methods described in Section 5.

To define a focus box, left click the mouse within the view window, drag the focus box, and then left click again to complete the box. If your mind after the first mouse click, you may hit the right mouse button, or the escape key. This will leave any previously existing focus box intact.

To remove a focus box, you may do any of the following: 1) double-click in the view window; 2) use the Focus menu; or 3) hit the “f” accelerator key.

By default, the focus box will be displayed using both a highlighting technique and a dashed outline. The highlighting technique makes the rest of the image slightly darker, and the focus box slightly lighter. You can turn off the highlighting by using the Focus menu, or hitting the “h” accelerator key.

You can widen or shrink an existing focus box using the Focus menu, or the “shift-w” and “shift-s” accelerators.

You can pan the focus box using the arrow keys, while holding down the shift key. Basically, the accelerators which affect the focus box are identical to those used to manipulate the view window, except that the shift key is held down.

1.4 Quality Layers

JPEG2000 codestreams may contain multiple embedded image qualities, which form the foundation of quality progressive remote image browsing. By default, local files are opened at maximum quality, but it can be useful to see the visual quality associated with each compressed quality layer. To reduce the number of quality layers you may use the Quality menu or the “<” accelerator. You may increase the quality again using the Quality menu using the “>” accelerator. These manipulations automatically change the status bar to one which displays the number of quality layers which are in use.

1.5 Navigating Between Image Elements

The “kdu-show” tool has 3 fundamental modes of image display, as follows:

Single Component Mode: In this mode, only a single image component is displayed as a greyscale image. Image components typically correspond to colour channels (e.g., red, green, blue, or luminance, and chrominance channels), but they may correspond to alpha blending channels, palette indices, or anything else.

To enter single component mode, use the View menu or hit either of the “1” or “+” accelerator keys. While in this mode, you may navigate forward and backward within the image components of a given codestream by using the View menu or the “+” and “-” accelerators.

While in single component mode, you may navigate amongst the codestreams in the file (JPX files may have any number of codestreams) using the RETURN and BACKSPACE keys. Alternatively, you may use the View menu.

Single Compositing Layer Mode: In this mode, only one compositing layer is displayed, usually as a colour image, or alpha blended against a solid white background. A JP2 file contains exactly one compositing layer, which is the rendered colour image. A JPX file may contain any number of compositing layers. A raw codestream is treated as having exactly one compositing layer, corresponding to our best guess regarding the intended display image.

To enter single compositing layer mode, use the View menu or hit the “L” accelerator key. While in this mode, you may navigate forward and backward amongst the compositing layers by using the RETURN and BACKSPACE keys, or the View menu.

You may also enter single compositing layer mode by holding both the “shift” and “control” keys down and left clicking the mouse, while it is over any particular compositing layer. This method allows you to quickly navigate to a particular compositing layer within a complete composition.

Composited Image Mode: In this mode, the application displays what it believes to be the intended image. For JP2 files and unwrapped raw code-streams, this is identical to the result displayed in single compositing layer mode. If a JPX file contains a composition box, however, the composition instructions are used to build the composited image which was intended by the content creator.

When a new file is opened, “kdu-show” starts out in the composited image mode, but if you have changed the mode to single component or single compositing layer mode, you may get back to composited image mode by using the View menu or hitting the “c” accelerator.

While in composited image mode, you may navigate forward and backward between composited frames (animations contain multiple frames, each of which is a composited image) by using the RETURN and BACKSPACE keys, or via the View menu.

2 Status Information

Some summary information is displayed at the bottom of the application window on its status bar. The status bar is divided into fields. The information displayed in these fields also depends upon the status mode, which may be toggled using the Status menu, or via the “t” accelerator key. In particular, you can currently toggle between the following three status modes:

1. Left field displays current zoom factor (percent) and image element; right field displays the number of quality layers. The image element depends on the image display mode. In single component mode, the component index and codestream index are displayed here; in single compositing layer mode, the compositing layer index is displayed here; in composited image mode, the animation frame index is displayed here.
2. Left field is as above; right field displays the full size of the image at the current resolution, measured in pixels.
3. Left field displays the amount of working memory used by the codestream management machinery (for all codestreams currently loaded into memory). This value is generally very small if the codestreams are enabled for random access (precincts, PLT marker segments and a layer-minor progression order).
4. (available only when connected to a JPIP server): left field displays active/idle transmission status (idle when all requested imagery for the current view/focus window has been received); centre field displays the total number of bytes received from the server; right field displays the average data rate at which transmitted bytes have arrived, averaged only over non-idle periods.

3 Remote Image Browsing with JPIP

By and large, images may be accessed remotely from a JPIP server (such as that implemented by the “*kdu-server*” utility) exactly as if they were local files. Rather than experiencing significant delays while the image downloads, you will instead observe incrementally improving quality as image data is incrementally transferred by the server. JPIP servers focus the transmitted data around the region defined by the focus window so the quality in that region increases more rapidly. If no focus window is defined, it is taken to be the current view window. Once sufficient data has been received to reconstruct the image within the focus window without loss, at the current display resolution, the server enters the idle mode and will not transmit any further data until the image region, resolution (zoom factor), codestream, image components, compositing layer or animation frame are changed.

It is important to realize that the viewing tool is only loosely (asynchronously) coupled to the client component and, ultimately, to the server utility. You are always free to pan, zoom and otherwise navigate the display to any part of the image, regardless of whether any data has been received from the server to render that region. After some time, the server will adjust its transmission pattern to serve the new image region/resolution which is currently of interest. Of course, if the server has disconnected, no amount of navigation will bring in new data, so you will generally be left with non-uniform image quality which reflects the amount of time spent browsing in different image regions.

It is also important to realize that the JPIP server is allowed to shrink the region of the image for which it will serve data, if the region is too large to be served in a quality progressive fashion without consuming excessive server memory resources. If this happens, the changed region will be reflected on the view window through a new focus box. You may move this modified focus box around, to change the region to which the server’s transmission is customized, but you may not enlarge the focus box without having the server force it back down again to its size limit.

3.1 Regular Image Browsing

To open a remote image, use the File->OpenURL menu item, or start “kdu-show” with an appropriate URL on the command line. The File->OpenURL dialog box provides several fields for you to fill out. For regular image browsing, enter the name of the file you want to access in the “Object requested from server” field and enter the name or IP address of the server machine in the “Server name or address” field. You may append an optional colon-separated port number to the server name or address, if your server is not listening on the default port 80.

One of three closely related JPIP transfer protocols may be selected via the “Channels and Sessions” drop-down list. For the most efficient communication, the “http-tcp” option is recommended; however, this requires the establishment of both an HTTP and a regular TCP channel, which might not be permitted

by some institutional firewalls. In this case, select “http” as the next best option. The “none” option is primarily for experimental purposes. It is much less efficient, especially for the server.

If your organization requires all external traffic to flow through an HTTP proxy, select the “http” option and put the name or address of your HTTP proxy server in the “Proxy name” field – you may need to consult your system administrator to find out the name of your HTTP proxy. Proxies reduce the communication efficiency and responsiveness, so use them only if necessary.

You may elect to have all your image browsing results saved in a local cache directory, by entering the name of this directory into the “Cache dir” field. The advantage of this is that when you open the same image again on the same server, or a compatible server (one which assigns the same unique target identifier to the image resource), the previous browsing results will be reused and that data will not be sent over again by the server.

An alternate way to open an image on a remote JPIP server is to supply the URL on the command line. The URL must commence with either “http://” or “jpip://” as the protocol identifier prefix, followed by the server host name (or IP address), a single “/” character, and then the resource string. The syntax for the resource string and the server name are identical to those used by the File->OpenURL dialog box. The alternate “jpip://” protocol identifier is provided so that the “kdu-show” utility can be fired up automatically from a web browser whenever a URL with this protocol identifier is encountered. In fact “kdu-show” registers itself with the Windows Internet Explorer as the target for such URLs, the first time it is ever used. You should be aware that when URLs are supplied on the command line, “kdu-show” obtains the remaining information (proxy server, cache directory, protocol variant) from the registry, based on the last values supplied via the File->OpenURL dialog. For this reason, it is advisable to use this dialog to configure the application first, before opening URL’s from the command line or by clicking on JPIP URLs on a web page.

You may explicitly disconnect from a JPIP server, without actually closing the image, by using the File->Disconnect menu item.

3.2 One-Shot Image Browsing

To implement the functionality of an image browser, “kdu-show” uses Kakadu’s *kdu-client* object to generate a sequence of much more complex HTTP requests than that supplied via a typical command line URL. These much richer requests are generated using either HTTP POST or HTTP GET with query parameters appended to the URL, separated by the usual “?” query separator.

As an alternative to interactive image browsing, you may supply one of these more complicated URLs directly by including the “?” query separator and the query fields of interest. In this case, only one request will be delivered to the server and no amount of interactive navigation within the image will cause any further requests to be sent to the server. The server will continue to transmit data relevant to the original request until all relevant information has been sent or the server times out or is disconnected. As a result, interactive navigation and

image data delivery are entirely disconnected processes; you may be receiving data relevant to a small image region at high resolution, but be viewing the image at low resolution (zoom factor) or vice-versa. You may even be viewing a completely different codestream within the data source and so not see the effects of newly arrived image data.

The main purpose of this mode of behaviour is for server debugging and for modeling the behaviour of JPIP when all imagery is generated in response to a static URL, which might be embedded in an HTML page.

You may also enter single-shot URLs via the File->OpenURL dialog. To do so, simply append the “?” query separator and relevant query parameters to the string entered in the “Object requested from server” field of the dialog box.

To use single shot URLs you will need to be familiar with the JPIP syntax. It is not the purpose of this document to describe this syntax, but the JPIP committee draft may be downloaded from “<http://www.jpeg.org/CDs15444.html>.” You may also examine the JPIP requests delivered to the Kakadu server during normal interactive image browsing by supplying the ‘-record’ command line option to “*kdu-server*”.

4 Viewing Codestream and File Properties

Use the File->Properties menu item or the “p” accelerator to view the properties of the current codestream. Where more than one codestream is involved in the current image display, the application chooses one of them to display its attributes. Codestream properties include comment marker segments, coding parameter marker segments and tiling attributes. You may double-click on any attribute to read a detailed description of that attribute’s interpretation, as provided by the Kakadu core system.

Additional attributes are provided by the file format in which codestreams are embedded. To examine the file format itself, use the View menu or the “m” accelerator to activate the “metashow” tool. You may click on various nodes within the metashow tool’s tree view to obtain additional details or to navigate to have the main window navigate to corresponding image elements. For example, clicking on the second compositing layer header box will cause the main window to enter single compositing layer mode and display the second compositing layer. Clicking on a contiguous codestream box or codestream header box will cause the main application to enter single component mode, while clicking on the composition box, if any, will cause the application to enter the composited image mode.

Clicking on some leaf nodes in the metadata tree will cause their contents to be expanded as text, while for other boxes a generic hex dump is provided.

5 Metadata Overlays and Editing

As of Kakadu v4.1, you may use “kdu-show” to edit and display spatially sensitive metadata in a JPX file. You may also add metadata to a JP2 file and resave it as JPX.

To add metadata use the Edit menu or the “a” accelerator. This opens up the metadata editing dialog, which allows you to enter an arbitrary text label. If no focus box has been selected, the metadata will be associated by default with the top-most current image entity (compositing layer or codestream), depending on the current image display mode (see Section 1.5). If a focus box was selected when you hit the “a” accelerator, the metadata will be associated by default with the top-most codestream which overlaps with the focus region; it will also be associated explicitly with that region. These associations are achieved by embedding appropriate number list and ROI description boxes in the JPX file (when you save the edited result).

You may alter any of the above associations within the metadata editing dialog. It will explicitly exclude options which are illegal. For example, metadata associated with a focus region must also be associated with at least one codestream, not just with compositing layers. You will see that you can add multiple associations, but it is up to you to ensure that this is meaningful.

Once you have some spatially associated metadata, you will notice that “kdu-show” highlights its presence via partially transparent overlays. You may use the Overlay menu to control the appearance of such overlays. Most usefully, the “ctrl-o” accelerator may be used to toggle the overlay mode between *flashing*, *static*, and *no overlays*. You may also control the darkness/lightness of overlays via the Overlay menu, or by using the “ctrl-d” and “ctrl-b” accelerator keys. Finally, you may control the minimum size of elements which will appear on overlays.

In the flashing or static overlay mode, try holding the control key down and moving the mouse cursor over regions which are associated with metadata. You will notice that the cursor changes to one which contains an “M” and the application displays temporary labels where appropriate. Keeping the control key down, you may also left click to open up the metadata editor again. This allows you to modify or delete existing metadata.

If there is any metadata associated with a codestream or compositing layer as a whole (e.g., the name of an image within a photo album), this metadata may be viewed and edited in a similar manner to region-specific metadata, described above. In particular, holding the control key down while the mouse is positioned over a compositing layer or codestream which contains metadata labels will cause the “M” cursor to appear, and any relevant label to be displayed temporarily. By left-clicking with the control key held down, you enter the metadata editing dialog. While there can be many metadata items associated with a codestream or compositing layer as a whole, only the “top-most” one associated with any given mouse location can be displayed and edited. The “top-most” one is determined based on the order of compositing layers. If no relevant compositing layer has metadata, individual code-streams which contribute to

visible layers are examined.

6 Saving Files

You may save images back out of “*kdu-show*” using the File->Save menu item. This is of interest particularly if you have edited the image’s metadata or if the image has been obtained from a remote JPIP server – in that case, the saved file will contain whatever information had been received from the server up to the point when it is saved.

You may edit and resave a file over the top of the file you already have open. In this case, however, the file is actually saved under a different name, formed by appending the “~” character to the name of the currently open file rather than overwriting it. When you close the open file, any such saved copy is automatically renamed so as to replace the one you had open. This happens even if you close the application (by any of the usual means, or by hitting the “q” accelerator).