Image display improvements

Priority	Area	Description	Comments	Ticket(s)	(orig. #)
	Layer dialog	Use a toggle for show/hide-all instead of two buttons. (Could be a state-changing single button or a tri-state checkbox.)	Consider making the layer dialog hierarchical, so that, for example, all the mask layers can be grouped, all the instrument footprints can be grouped, etc.		8
	Layer dialog	Allow creating a group of uploaded regions via the Python API.	Needs further discussion of whether the individual regions should still be visible in the layer dialog. May just be a question of what the afw.display interface for multiple-region upload should look like.		1
	Color naming	In all APIs that take color names, we should use the current W3C/CSS3 interpretation.	Regarding X11-vs-W3C conflicts, see Clashes between web and X11 colors in the CSS color scheme; we will standardize on W3C.		2
	API	Provide an API for setting the "target" for any uploaded image, so that it will be used by the Firefly "center image" button. ("Center image: Currently we center the image at the last input position or image center. Maybe allow user to input a position to center the image on. For API, allow user to give position as parameter to center on.)	This should perhaps be thought about together with the request for a pan-to-specified position UI element (most useful in HiPS, of course).		3
	Lock/pan /zoom /WCS match	Please separate locking of pan/zoom from the stretch and colorization settings.	Already under review in-house; we agree.		4
	Region display	After multiple zoom-in / zoom-out operations, the image and region layer positions can get out of sync.	A reproducible example would help.		7
	Mask coloring / API	Confusion between alpha and transparency in API. afw.display uses transparency.	Perhaps afw.display should also be able to accept alpha?	⚠ DM-15438 - Jira project doesn't exist or you don't have permission to view it.	9
	Coordinate readout	Add a display of the masks that are set on the current pixel in the coordinate display.	Possibly display an abbreviated form by default, with the full list as a tooltip (and therefore only available when in lock-by-click mode, since the mouse is by definition not over the display!).		11
	Coordinate readout	Format of flux values: always-on exponential notation is hard to read. Consider something like FORTRAN "G" format.	Currently it appears that the "E" appears when values are greater than 1000. With the dynamic range and offset of LSST data, this will be much of the time...		19
	Mask coloring	Bug: When open the layer dialog and tried to change the color for one mask, the color picker dialog showed a different color from the original. (Xiuqin can't reproduce it directly in my local Firefly in Aug 8 Git commit e6a11ce67 build.)			10
	Image inset	Try to always display at full resolution, i.e., with visible single pixels (zoom level 3x or 4x). May require actually doing that only on mouse-pause during panning, as a server call is required to get the full-resolution data.	Perhaps this behavior should be a selectable alternative to the current one of displaying the inset at double the zoom of the main window.		5
	Image inset	Highlight the mouse position on the main image in the inset image. Display all the layers of the main image on the inset, if possible. Specifically requested: masks. (Should also work for (LSST) footprints.)	Highlight could be a crosshair or a target circle. Be careful that this works properly near the edge of an image. Highlight should also work for		5

	Image performance	Pan and zoom is slow, especially on the Kubernetes deployment of Firefly in `lsst-lspdev`. Look into possible causes: 1. K8s network overlay 2. Handling of memory & swap interaction with K8s. Are we getting real or virtual memory? 3. File system mount: nfs, gpfs. Will the direct local working directory help? 4. Request SSD for Firefly cache			6
	Image performance	Pan and zoom are especially slow when masks are displayed. Can be difficult to tell when an update is complete. Can the "working" indication be improved?			15
	Mask coloring	Provide a simpler / more direct UI means to change only the transparency of a mask.	What about a special control for the whole image mask group in the layer dialog, which changes only the alpha?		12
	Python API behavior	Enable user to change Firefly server on the fly, repoint at firefly_display level. reason: notebook users may want to access different Firefly server or session/"channel" at different sections of a notebook.			13
	Stretch issues	Bug: Strange behavior of the checkbox for "Z-scale" in the asinh dialog.	Hard to reproduce. Xiuqin may have seen it once.		new
	Stretch issues	Consistency of when "Refresh" (should it be called "Apply"?) is needed: 1. For the new asinh stretch, changing parameter Q results in the stretch change in image 2. But in other types of stretch, "Apply" button has to be clicked for the change to take effect.		⚠️ DM-15205 - Jira project doesn't exist or you don't have permission to view it.	16
	Region display	Bug: At high zoom levels, sometimes the edges of regions seem to disappear. Seems to happen when all four corners of a region are outside the displayed area of the image.	Has been difficult to reproduce in-house. Would be useful to get RHL's notebook.		17
	Coordinate readout	Capture arrow-key events and use them to move the mouse when it is over an image. Moves the mouse-selected point to the next pixel center, then moves by whole pixels. Should affect the zoomed inset.	(Relies on LSST and FITS conventions where the pixel center is at an integer (not half-integer) value.) What should this do when lock-by-click is in effect? Move the locked position, presumably. Does this affect the zoomed inset? Proposed behavior: 1. When lock-by-click is OFF, the arrow keys just move the mouse: for FITS, by single source image pixels; and for HiPS, by the size of the displayed HiPS pixel at the center of the image window (i.e., 1/512th of the current tile size) in the vertical or horizontal directions in the as-reprojected image. The mouse actually moves on the screen. All behaviors of the coordinate readout and image inset are as normal for mouse movement. 2. When lock-by-click is ON but no position has been selected yet, the arrow keys still move the mouse as above. The space bar can be used as an equivalent to the primary mouse button (i.e., to do the locking). 3. When lock-by-click is ON and a position has been locked, the arrow keys move the locked position by single pixels, as above, **but do not move the mouse**. The image inset works as it currently does when the mouse is over the image and the mouse is moved, i.e., the inset follows the mouse but the locked coordinates do not. When the arrows keys are used, and the locked coordinates change, the image inset jumps to be centered on the locked position, not the mouse. 4. In the inset, both the mouse position AND the locked position are shown if possible, and distinct marks are used. The mark for the locked position should be the same one as used in the regular image (currently a yellow square). The mouse position could be shown with a crosshair or target circle, as mentioned above. (Original #5 in this list.) Q: What should we do if a FITS image is rotated? Does the horizontal arrow always move by pixel in the serial (NAXIS1) direction, even if that happens to be, say, up?		18

	Image subwindow headers	When there are two images in the single-image mode (regular or expanded), there is only one arrow to go through them. There should still be two arrows for consistency.	This needs a bit more thought about how the "next image" names are displayed in expanded mode.		20
	Image subwindow headers	The locations of the next/previous-image arrows, and the location and presence of the single/grid/selector buttons, need to be consistent across all modes and between "slate" and "portal" modes.			new
	Application-level appearance	The "channel name" should be available somewhere in the UI and should also be in the <head><title> element so that it appears in tab names in browsers.	(gpdf comment)		new
	Application-level appearance	Allow the image toolbar to be moved to the side (left, perhaps alternatively right) of the browser window. Allows the user to have a larger square region for image-viewing.			21
	Image titling	There should be API-level control of what appears and does not appear in the image title (the upper-left strip), e.g., whether to display the FOV, the zoom level, etc.			14
	Image titling	One element that should be displayable in the image title field is the `afw.display` image viewport number (the "frame" parameter).	Can be achieved by prepending the frame ID to the image name for transmission to Firefly. May be reasonable to have more explicit support in Firefly for giving each viewport in a slate a name. (Discuss with David Shupe.)		new