

VirtualViewer 4.10 New Features and Corrected Issues

December 20, 2017

Important Phone Numbers and Links

For the most current information, please contact Snowbound Sales at

Telephone: 1-617-607-2010 or

<http://register.snowbound.com/MQL-contactUs-Website-2017.html> or

questions@snowbound.com or

<https://mylivechat.com/chatnoscript.aspx?HCCID=17729140> (sales inquiries only)

Filing a Support Ticket

<https://snowboundsupport.force.com/SupportPortal/CommunityLogin?>

Release Notes and Product Manuals

<http://www.snowbound.com/support/manuals>

Documentation Corrections

Important Information

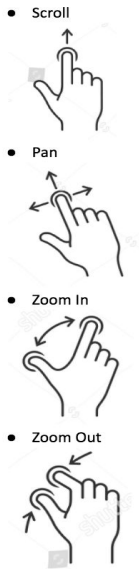
- VirtualViewer for Java now supports JRE 1.8+. Previous JRE versions are no longer tested.
- For Windows products, .NET framework versions 4.5.2 and up are now supported

New Features

Mobile Device Control Improvements

The user can pan on an image, if it's zoomed in, in all directions (imageScrollbars should be set to false in config.js for this to work).

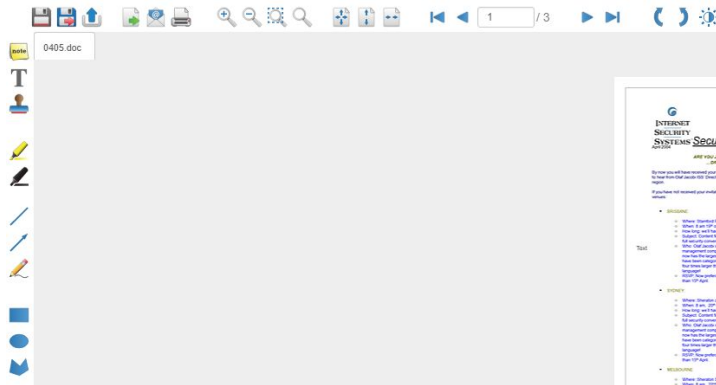
Using two fingers, the user can pinch to zoom in or zoom out on the document. The zoom motion is not animated so the increase/decrease in size will happen when the user is done pinching.



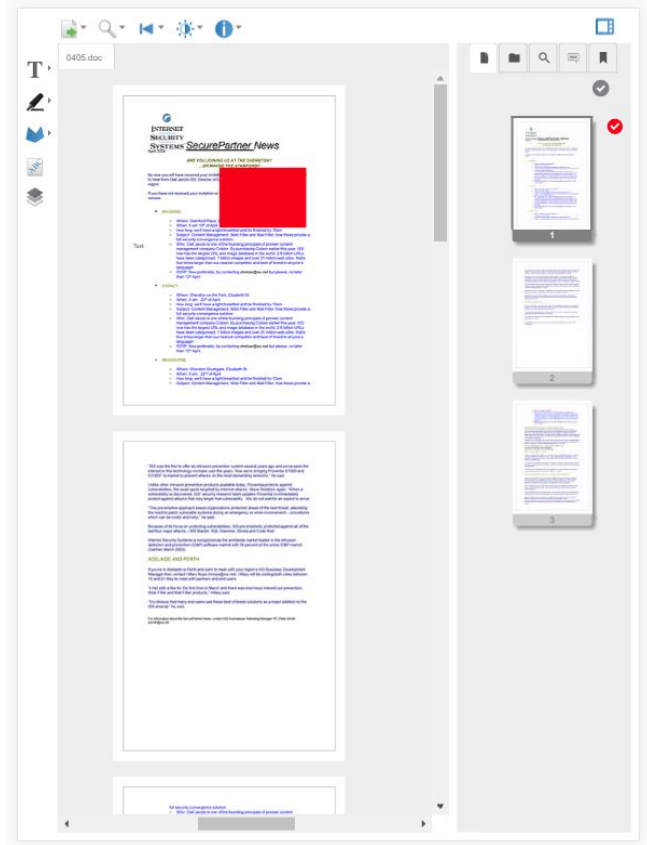
The viewer now works more elegantly in small iframes, on low-resolution monitors, and when browser windows resized smaller.

Toolbar details

On large screens, the toolbars look exactly the same as before:

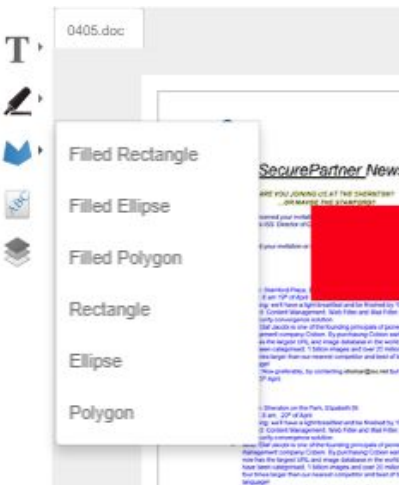


But as you shrink down, buttons collapse into menus. This happens as you shrink your monitor size, lower your resolution, or even shrink your browser. It's especially important for embedding the viewer in small iframes--like Alfresco--or for using the viewer on an iPad:



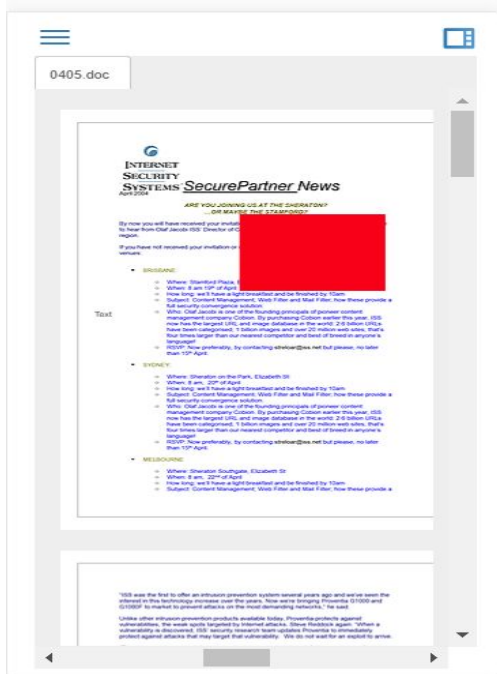
(Note that these icons are temporary.)

Here, you can see what the dropdown menus look like:

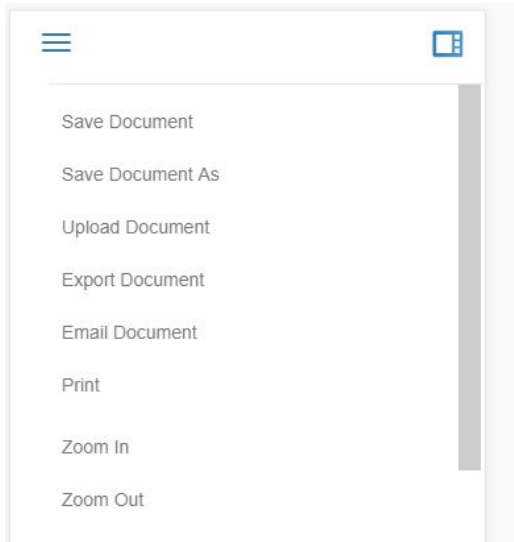


As you shrink down even further, into phone real estate or into the smallest your browser can get, we have even less space on the toolbar. So, we stuff all the options into a large menu, reachable by the three lines in the top left (called a "hamburger button"):

VirtualViewer 4.10 New Features and Corrected Issues



This menu takes up the entire screen when open:



VirtualViewer 4.10 New Features and Corrected Issues

Toolbar buttons are also now highly configurable.

A deep dive into the configuration

The file `user-config/toolbar-config.js` now defines everything about the toolbars. Customer admins can go in and manage this instead of modifying `index.html` directly: VirtualViewer code goes into this configuration file and creates a toolbar button for every configuration entry that it sees.

There are three parts to `toolbar-config.js`. First, we have the two big lists: `imageToolbarButtons` for the top toolbar and `annotationToolbarButtons` for the left-hand toolbar. Their names should categorically not be modified, particularly not in the final return statement in `toolbar-config.js`.

Take, for instance, `imageToolbarButtons`. It looks like this:

```
var imageToolbarButtons = {  
  
  "buttonKey": {button configuration},  
  
  "anotherButtonKey": {button configuration}  
  
};
```

Each "buttonKey" is a unique, descriptive key for the toolbar button. Unique because it's defining the button in the configuration, and also because it's used in the HTML. It should be letters only, and it is good practice for customers to put a prefix on this key--for instance Big Company, Inc may call a key "bcMyKey."

This is the same structure as the annotation toolbar list of buttons. Each button key should also be unique between the two lists--don't put "vvMyNewButton" as a key in the image toolbar configuration and in the annotation toolbar configuration.

Now we can look at the button configuration. In the curly brackets, there are a few items:

-localizeKey: This is a string referring to a locale file, where the text that appears in the title--what appears in the tooltip and in the dropdowns--is stored. If a customer is creating a whole new button that doesn't have an entry in a locale file, this field can be left blank, but the "name" field *must* be filled in or the button will appear blank in dropdowns.

-name: This is a string with the name of a toolbar button. VirtualViewer buttons do not use this property, since they use localized strings, but customers without an entry in a locale file must put the name of their toolbar button in this field. This should be something short and descriptive.

-clickHandler: This is the function that will be called when a user clicks on the toolbar button.

VirtualViewer 4.10 New Features and Corrected Issues

-iconImage: This is a string representing a URL to an icon image. VirtualViewer buttons use CSS files to assign icon images, but customers can list an image here and it will be used.

-addSeparatorAfter: This is a boolean value. If a group of buttons is too long, it may be necessary to add a bit of whitespace after a button to visually break up the group when the toolbar is fully expanded. If this is true, the viewer will pop some white space after the button. It is not necessary to specify "false" on this item.

-groupId: This is a string, and it refers to a group key. Groups are also configured in toolbar-config.js. If a button is in group A, it will be placed in group A in the order it appears--if the Sticky Note button is before the Rubber Stamp button in the annotation toolbar list, and they both belong to group A, the Sticky Note button will appear before the Rubber Stamp button in the UI. If an item is in the annotation toolbar list, it should be assigned to an annotation toolbar group. Additionally, this is optional--if a button has no groupId, it will essentially be in an "uncategorized" list. It will appear at the end of the toolbar and will not be included in a dropdown. (See the Layer Manager and Watermarks buttons as default examples.)

The final part of toolbar-config.js is the group configuration list, called toolbarButtonLogicalGroups. This list looks very similar to the button configuration lists. What it will do is define all the buckets that the toolbar buttons can go into.

Again, each group has a unique string key followed by a configuration object:

```
var toolbarButtonLogicalGroups = {  
  
  "myGroupKey": { configuration },  
  
  "myOtherGroupKey": { configuration }  
  
};
```

The configuration items are as follows:

-localizeKey: This is a string referring to a locale file, where the text that appears in the tooltip of the dropdown button is stored. This can be left blank.

-groupTitle: Like "name" for the buttons, this is a string with the name of the group in it. For instance, "Preferences" or "Edit" or "File." The localize key will handle this, but a customer without a locale entry could use "groupTitle".

-iconImage: This is a string representing a URL to an icon image. VirtualViewer buttons use CSS files to assign icon images, but customers can list an image here and it will be used.

-annotationToolbar: This is a boolean flag that chooses where to put the group. If true, the group is an annotation toolbar group; if false or absent, the group is in the top toolbar. Annotation toolbar buttons

should only be placed in annotation groups, and similarly, image toolbar buttons should only be placed in image toolbar groups.

New strings in vv-en.json

utilityToolbar.fileGroup: File

utilityToolbar.zoomGroup: Zoom

utilityToolbar.pagesGroup: Pages

utilityToolbar.pageManipulationGroup: Page Manipulation

utilityToolbar.infoGroup: Info and Settings

annToolbar.textAndStampsGroup: Text and Stamps

annToolbar.markupGroup: Markup

annToolbar.shapesGroup: Shapes

Unsupported Features on Mobile Devices

Drag and Drop

Search and redact are disabled if redaction is disabled

Document comparison, when offered, will not be supported for mobile devices

Though several mobile devices have been approved, we cannot guarantee that all mobile devices are supported.

Alfresco Quickshare Support

A logged-in user can create or close a public quickshare link through Alfresco. If a logged-in user accesses that quickshare link, they can see and modify the document through VirtualViewer as normal; if a guest or unauthenticated user accesses the quickshare link, they will be able to view the document but not save any modifications to the document (or annotations, notes, etc).

Currently the unauthenticated user gets the full VirtualViewer UI with a readable error message if they take any action that tries to save the document. In the future they should get a limited UI that doesn't present those options.

Alfresco Watermark Support

Added support for Watermarks in the Alfresco version of VirtualViewer allowing saving watermarks back into the Alfresco repository. All supported features or functions remain.

VirtualViewer 4.10 New Features and Corrected Issues

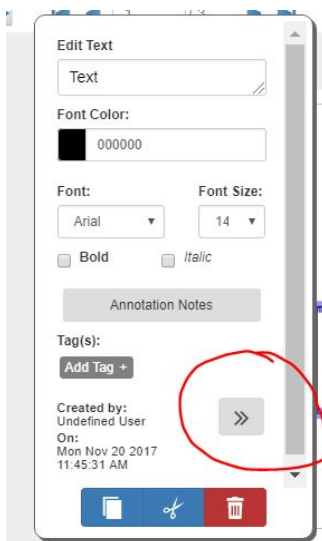
Revision history for Annotation Create Date/Time

Use Case:

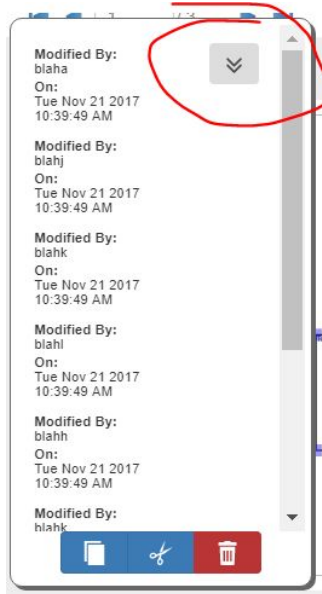
1. User 1 creates an sticky note on page 1 of a document, the userId/date/time are recorded
 2. User 2 edits that same sticky note (color, size, placement, any change really....ect)
 3. The userId/date/time are updated in a scrolling list reflecting each edit to that object.
- Works for all annotations, image stamps and redactions
 - Resizing/moving records as a change
 - Changing text records as a change
 - Page manipulations or moving pages with annotations to a new/other document will not record a change
 - The use of pages with annotations in a VD will record changes to the annotations
 - A modification will be added if the annotation is modified and then saved.
 - If the user changes the color, moves around the annotation, and expands the annotation, and then saves once, only one modification item will be saved.
 - If an annotation is pasted, it will have a clean slate--it won't keep the modifications of the original.
 - If a document is saved-as, the annotations on the new document will still have a modification trail but will not add a new modification for the saving event.

Display and Use

The revision history is displayed in the annotation popup by clicking an expando button:



And shrunk again by hitting the same button:



Filenet F_CREATOR Tag Support Added

NET 4.5.2+ Requirement Added for TLS 1.2 support

OCR Integration (beta)

This is a beta version of the OCR option in VirtualViewer. The final version is expected in the VirtualViewer v4.11 release where some alignment issues and ability to work with additional input file formats will be added. We also expect to offer a choice of OCR recognition engines in 2018.

The OCR function allows searching text in an image document (TIFF or PNG initially) as well as selecting text in the VV client after the document has been OCR'd. To OCR a document in the VV client, a user must search for text in a non-text document to get the OCR prompt. The OCR'd result is cached; while that result is cached, the user can search for and select text without a further OCR prompt. Searching is performed using the Search tab in the thumbnail panel.

The original image will overlay the OCR'd textual data to maintain the greatest similarity to the original document. The search text string will be highlighted. "Previous" and "Next" match buttons will work as normal. "Redact" and "Redact All Matches" work as normal. Applying redaction tags to results works as normal.

A wait icon will be displayed while the OCR process is running.

VirtualViewer 4.10 New Features and Corrected Issues

OCR will not be initiated if the input document is not raster. Saving to a PDF file is an option. Additional language support can be added by the customer.

The two new parameters necessary to enable OCR are in web.xml (web.config for .NET):

- `enableOcr`: Enable OCR for searching and text extraction. Must have a valid OCR configuration and licensing to function correctly. Defaults to false.
- `tesseractDataPath`: Absolute or relative path to Tesseract OCR Engine's training data. If using packed WARs in Tomcat, this needs to be changed to an external unpacked folder. Defaults to `"/tessdata"`.

Fixed Customer Issues in VV 4.10

Small screen issues fixed

- The document no longer overlaps the thumbnail panel when the viewing canvas gets very small
- Thumbnails work on iOS devices, annotations may be drawn and modified on all sizes of screen
- The thumbnail pane starts out hidden if it's going to take up too much screen real estate
- Modal dialogs are usable on small screens
- Annotation context menu is usable on small screens
- Scrolling works on small screens

Other Fixed Customer Issues in VV 4.10

- IOS Mobile: Export to Tiff/Original opens only the 1st page in new tab
 - This is an issue with IOS where it only displays the first page of a multi-page TIFF. The other pages are preserved and viewable on other applications. (This is also true when viewing such documents in Gmail.
- VV Net 4.9: Constant reloading of thumbnails
- FileAndURLRetriever class missing from VV Java v4.9
- VV: PDF Arial to Times New Roman conversion error fixed
- Tabs, canvas, and thumbs can lose sync
- Annotation Highlight Rectangle does not use highlightFillColor
- Annotations are not burnt in via export
- Text Ann Color Selection
- Confirm Changes dialog box not appearing
- Snowbound annotation tags are overwritten
- Page size increased after redaction applied
- Filenet annotation values are overwritten
- PDF document comments are not being displayed
- VV Net HTML plug issue
- PaperPort 12 PDF: Large file stalls or fails to load
- VV .NET cannot get files over HTTPS TLS 1.2 on framework 4.5
- Color is not displayed and size is shrunk in pdf document
- Cannot view annotation with permission PERM_VIEW = 40

END OF RELEASE NOTES

VirtualViewer 4.10 New Features and Corrected Issues