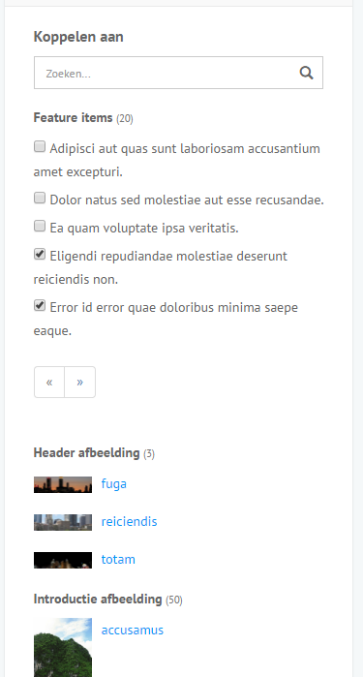
# INTEGRATED-416 Allow inline use of images



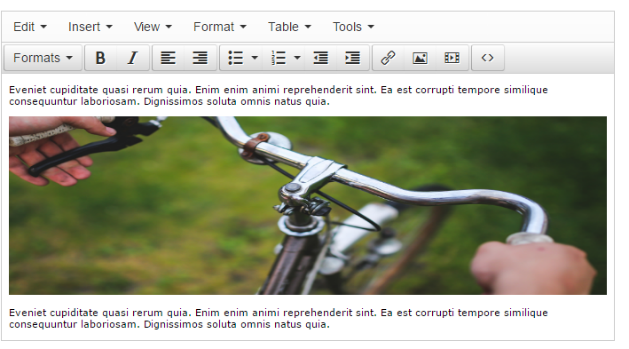
## Current situation

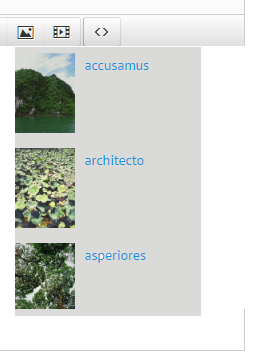
* Images are linked to articles using a relation (image 1).
* The website (or other kind of channels) render articles with images on article-level.
* Content integrity is guaranteed (when an image is removed or changed the display of the article will be adjusted, because it uses the Image object.
* Images can be moved. The StorageBundle allows storage migration. This is not a problem at all in the current situation.





## New requirement

* A specified image position within the article need to be possible, somewhere in the middle of the the text content.
* The user is able to insert the image at any position in the text, but this well result in a full width image view (or maybe placeholder that will be designed). So it is not possible to put an image next to the text. It is also not possible to put multiple images next to each other. This is because the way of display need to be implemented by the channel (for example the website). So we only need to know the image position. And we need a nice way to store it.
* Content integrity need to stay guaranteed. This means only images that have been linked to the article can be used in the content.
* The display of the images still need to be implemented on the channel. So (for example) the website decides if multiple images are displayed next to each other, which width is used, if an zoom option is available, etc.
* Storage migration still need to be supported.



## Solution approach

* The editor will have an image insert button. The user can choose an image from the linked ones (see the ugly example). The image will be inserted on the cursor position.
* The image reference need to be stored in the content. One way can be to use an <img> tag with “data-id” attribute and “class” attribute to be able to style it.
* The rendering on the website front-end need to be adjusted to support inline images: inline images need to be excluded from the article-level images and displayed in the correct way. The StorageBundle need to be used to use the right storage location and an image resize is needed as well (Gregwar). When using an <img> tag the HTML need to be parsed for this.

## Comment about article body content

* The article body content is currently saved as limited HTML. Edit editor supports H1, H2, H3, H4, H5, H6, Preformatted, Strong, Italic, Underline, Bullet list, Numbered list, Link, Table.
* Style classes for the editor can be configured in the application.
* It would be nice to define and document the Integrated body content language.

## Needed in a technical design

* What is the most excellent way to storage the image references?
* What is the most excellent way to do the front-end rendering?
* Which “objects” are needed?
* How to handle deleted relations or images?
* The way of front-end implementation is less important and can be skipped in the technical design. This can be done with a TinyMCE plugin and jQuery.
* How to use this for other file content types, for example Video or PDF?
* How to use this for other content types like tweets?