Dynamic resize works in low level API by converting jagged array into table structure. If custom behavior is required, such as show only top X rows, plugin can be registered to transform input value.

|  |
| --- |
| [[dynamic]:top(10)] |

Processor plugins works in high level API by applying transformations on low level API. Since custom plugins are called before built-in plugins, we can use it for various transformations

|  |  |  |
| --- | --- | --- |
| Column 1 | Second column | Third column |
| [[fixed.column1]:limit(5)] | [[fixed.column2]] | [[fixed.column3]] |

Although the previous example works due to mutations, it’s hacky way to get the desired behavior. The suggested way is via new v5 navigation API. Ideally paired with aliases so that tags are still readable. Navigation plugin requires activation by specifying separator char (in this case : )

Due to usage of extra expression in path, they will be considered different (fixed vs fixed:limit(3) )

|  |  |  |
| --- | --- | --- |
| Column 1 | Second column | Third column |
| [[fixed:limit(3).column1]] | [[fixed:limit(3).column2]] | [[fixed:limit(3).column3]] |

Benefits of navigation API shine when input needs to be transformed, e.g. a simple list needs to be grouped by property into groups.

|  |
| --- |
| [[list:group(A).key]] |
| * Details: [[list:group(A).value:limit(3).B]] at [[list:group(A).value:limit(3).C]] |