Graph Sequences Viewer Users Guide

Overview

Graph Sequences Viewer (GSV) is a tool based on cytoscape.js and dedicated to have a graphical represantation of mapping reads in graph. Each reads is a node and overlaps between two reads is an edge. Graph representation for mapping reads allow to view quickly each way can be possible to extend a sequence. The input can be a simple JSON with graph description(nodes and edges). But it's recommended to use the output of the tool Mapsembler. Mapsembler take one or many references sequences and try to extends all of them with sets of reads. Some outputs are available in Mapsembler whose json graph output. This output is a specific JSON formated that allow to use full possibilities of GSV.

Graph Sequences Viewer allow to explore the graph and help to find elements with a particalary interest through a vizmapper. Vizmapper allow to apply style (shape, color , size) on graph elements in function of data properties of this elements (sequences length, coverage,...). Each elements of the graph are clickable and allowed to see various information in a rectractable panel. This panel have some functions, especialy for nodes, for use the data displayed like sequences concatenation, comment and hightlight. And it's possible to export all sequences displayed in this panel (nodes squences and concatenated sequences) in text files.

Compatibility table

4.0+ 5.0+ 5.1.7+ 9.0+ 11.6+

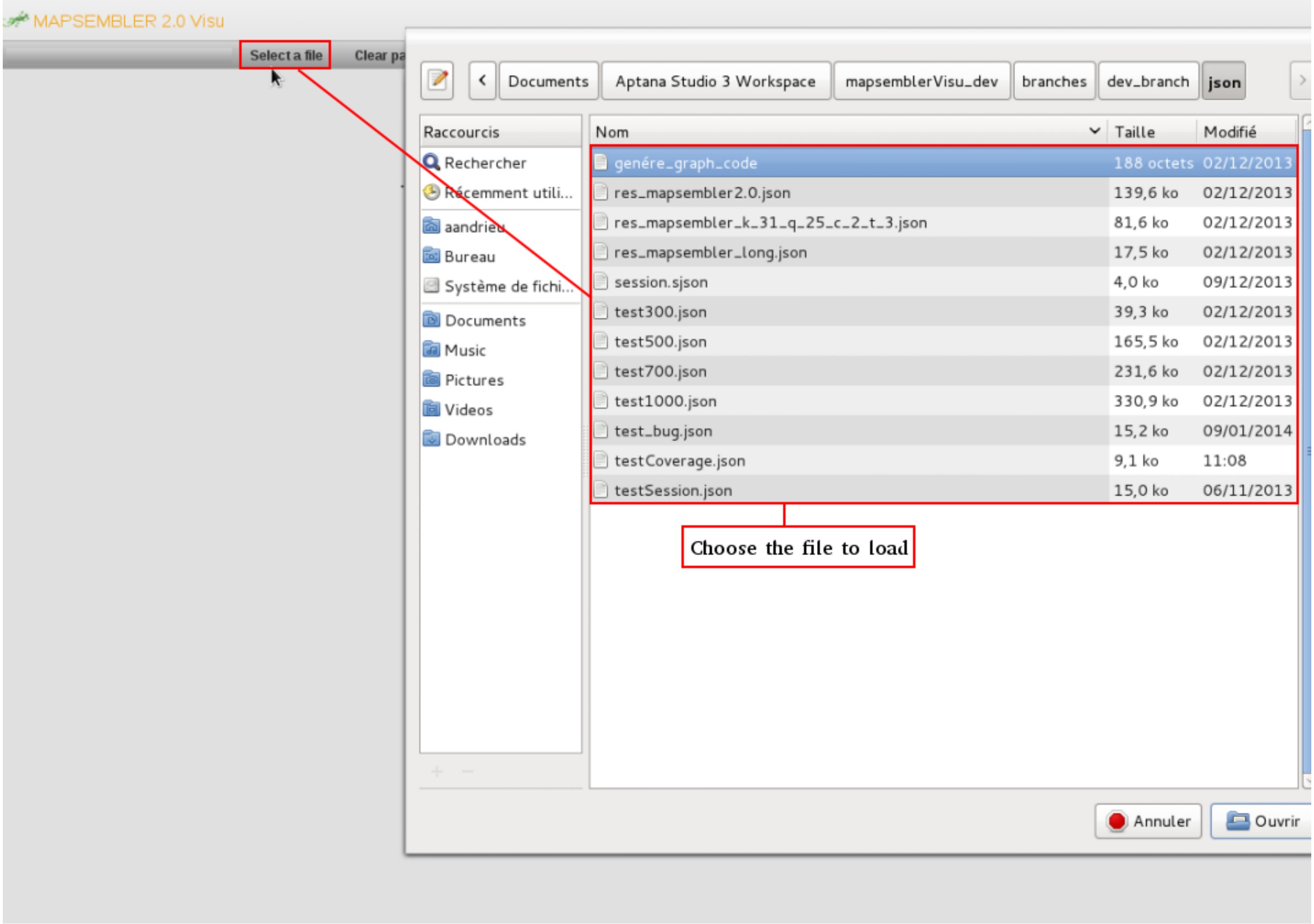
Compatibility table is valid on MacOS X, Windows and Linux.
But several tests had been made only on :
- Windows 7 with IE 10, IE 11, Firefox 26, Chrome 31, Opera 12.16
- MacOS X Montain Lion with Safari 6.0.2, FireFox 26, Firefox 27, Chrome 30, Chrome 31 and Opera 12.16
- MacOS X Maverick with Safari 7.0, FireFox 26, Firefox 27, Chrome 31
- Fedora 17 with FireFox 15, Chrome 22

Some problems are present with Chrome >= 31 on MacOS X, please use an other browser. For others compatibilities problems are questions, please contact alexan.andrieux@inria.fr.

1. Start Page

1.1. Load a File

The start page allow to load a graph file (.json) or a session file (.sjson). If the application is launched in standalone, any compatible file save in your hard drive can be open. But if the application is online, only the file present on the server of the application can be open. So with an online instance, it's necessary to upload files on server in a directory with good right (For example: json directory in root directory of the application).



Nom	Taille	Modifié
genére_graph_code	188 octets	02/12/2013
res_mapsembler2.0.json	139,6 ko	02/12/2013
res_mapsembler_k_31_q_25_c_2_t_3.json	81,6 ko	02/12/2013
res_mapsembler_long.json	17,5 ko	02/12/2013
session.sjson	4,0 ko	09/12/2013
test300.json	39,3 ko	02/12/2013
test500.json	165,5 ko	02/12/2013
test700.json	231,6 ko	02/12/2013
test1000.json	330,9 ko	02/12/2013
test_bug.json	15,2 ko	09/01/2014
testCoverage.json	9,1 ko	11:08
testSession.json	15,0 ko	06/11/2013

Choose the file to load

a. Graph File

The graph file can be output of Mapsembler, but also a simple json. In the minimal structure for a compatible json is:

```
{
"nodes": [
    {"data": {"id": "0","sequence": "ATGC"}},
    {"data": {"id": "1","sequence": "ATGC"}},
    {"data": {"id": "2","sequence": "ATGC" }},
    {"data": {"id": "3","sequence": "ATGC"}},
    {"data": {"id": "4","sequence": "ATGC"}}
],
"edges": [
    {"data": {"id": "e0","source": "0","target": "1","direction": "FF"}},
    {"data":{"id": "e1","source": "1","target": "0","direction": "RR"}},
    { "data": {"id": "e2","source": "0","target": "3","direction": "FF"}},
    { "data": {"id": "e3","source": "3","target": "0","direction": "RR"}},
    {"data": {"id": "e4","source": "1","target": "2","direction": "FF"}},
    {"data": {"id": "e5","source": "2","target": "1","direction": "RR"}},
    {"data": {"id": "e6","source": "3","target": "4","direction": "FF"}},
    {"data": {"id": "e7","source": "4","target": "3","direction": "RR"}}
]
}
```

For better performances, the property "length", refered to the size of the sequence, can be add in nodes data. In this way, the application must not calculate size of sequences and save several time, especially with large graph.

```
    {"data": {"id": "0","length":4 "sequence": "ATGC"}},
```

Data nodes and edges can contains also the property coverage. Coverage is an average come from differents files. So coverage property is an array containing identity of files and value of average coverage:

```
{"data": {"id": "0","length":4 "sequence": "ATGC",
coverage":[
    {"id":"file_1","avg_coverage": 106.47},
    {"id":"file_2","avg_coverage": 106.47}}
]
```

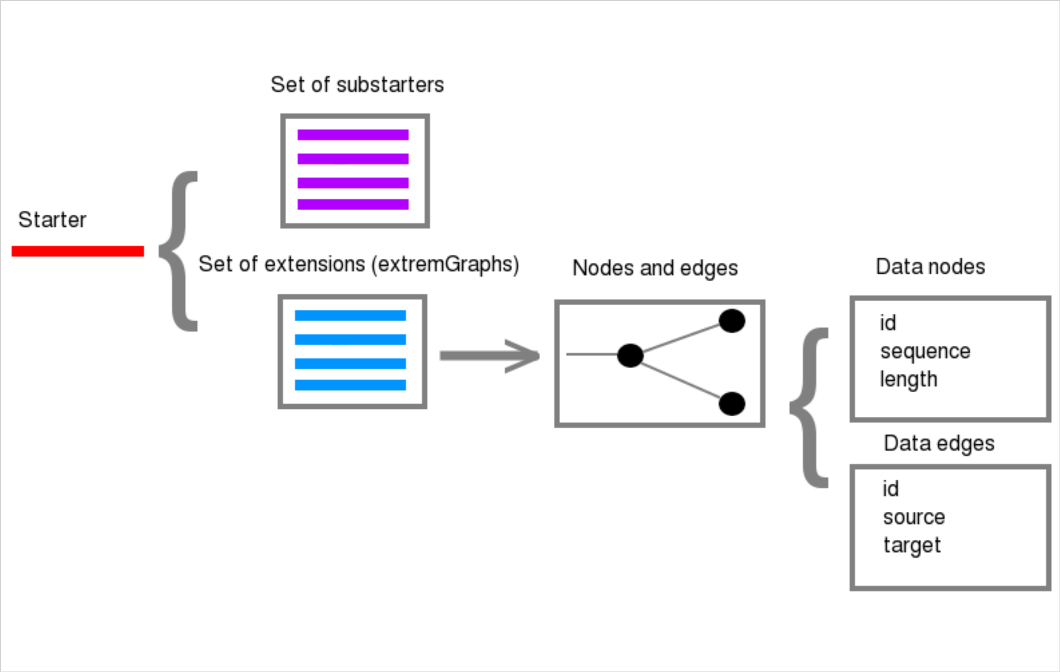
To use concatenation function , it's necessary to add in nodes data the property k, refered to the overlap between sequences of nodes linked.

```
    {"data": {"id": "0","length":4 "sequence": "ATGC", "k":3}},
```

When minimal json is loaded, the graph viewer is display automatically.

Mapsembler json output has a complex structure and can include several graphs. With this json file, it's allow to make choices on starters, substarters and coverage files to take into consideration for the graph wanted. For each starter, there are at least one substarter (the substarter can be the starter himself) and for each substarter there are one are two extensions. So the Mapsembler output looks like:

Shema :



Code :

```
{
"Starter_0":{"id":"S0", "sequence":"ATGC",
"substarters": [
    { "data": { "id":"s0", "sequence":"ATGC", "length":4, "extremGraphRight":"k0",
    "extremGraphLeft":"none"}}
],
"extremGraphs":[ {"data":{"id":"k0", "sequence":"ATGC", "direction":"RIGHT",
"firstNodeId":"n0",
"nodes": [
    {"data": {"id": "0","sequence": "ATGC"}},
    {"data": {"id": "1","sequence": "ATGC"}},
    {"data": {"id": "2","sequence": "ATGC" }},
    {"data": {"id": "3","sequence": "ATGC"}},
    {"data": {"id": "4","sequence": "ATGC"}}
],
"edges": [
    {"data": {"id": "e0","source": "0","target": "1","direction": "FF"}},
    {"data":{"id": "e1","source": "1","target": "0","direction": "RR"}},
    { "data": {"id": "e2","source": "0","target": "3","direction": "FF"}},
    { "data": {"id": "e3","source": "3","target": "0","direction": "RR"}},
    {"data": {"id": "e4","source": "1","target": "2","direction": "FF"}},
    {"data": {"id": "e5","source": "2","target": "1","direction": "RR"}},
    {"data": {"id": "e6","source": "3","target": "4","direction": "FF"}},
    {"data": {"id": "e7","source": "4","target": "3","direction": "RR"}}
]
}}
]
}
}
```

Likes to minimal json it's possible to have properties "length" in data nodes and "coverage" in data nodes and edges. Whereas the property "k", is calculated automaticaly with this json structure.

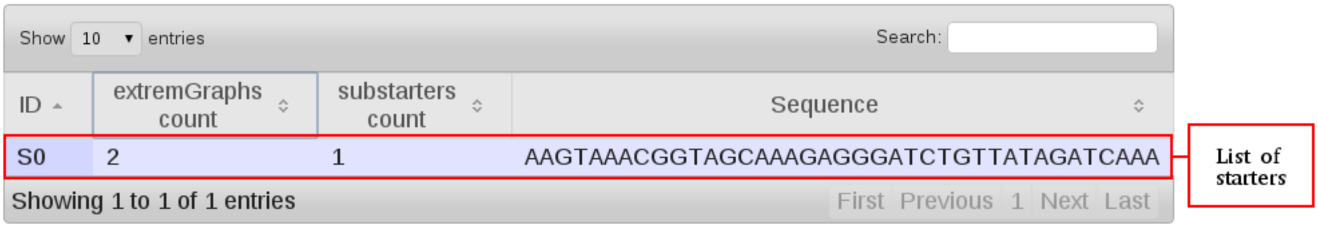
b. Session File

It's possible to load a session file (.sjson). Session files contain positions of nodes and vizmapper properties defined previousely for nodes and edges. Loaded a session file, display automatically the graph viewer.

1.2. Data tables of graph loaded

a. Starters

When mapsembler json output is loaded, a table of starters has been displayed.

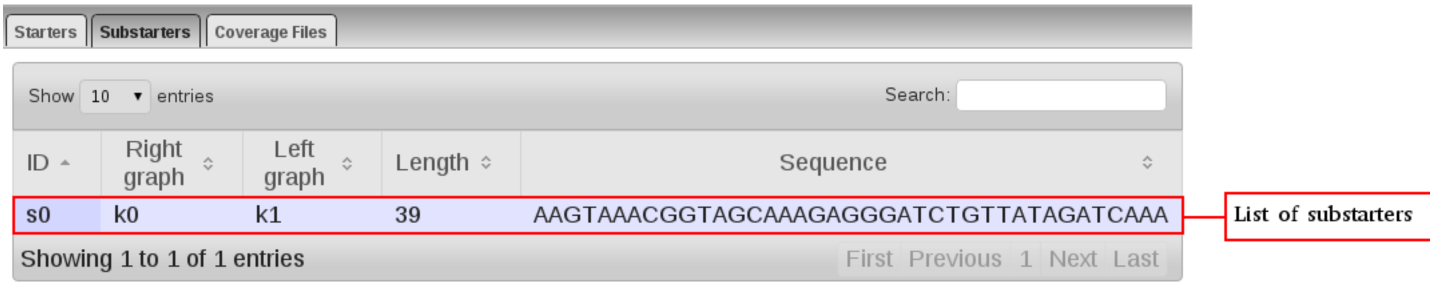


ID	extremGraphs count	substarters count	Sequence
S0	2	1	AAGTAAACGGTAGCAAAGAGGATCTGTTATAGATCAAA

Showing 1 to 1 of 1 entries

Click on one of them selecting and displaying substarters table in a new internal tab.

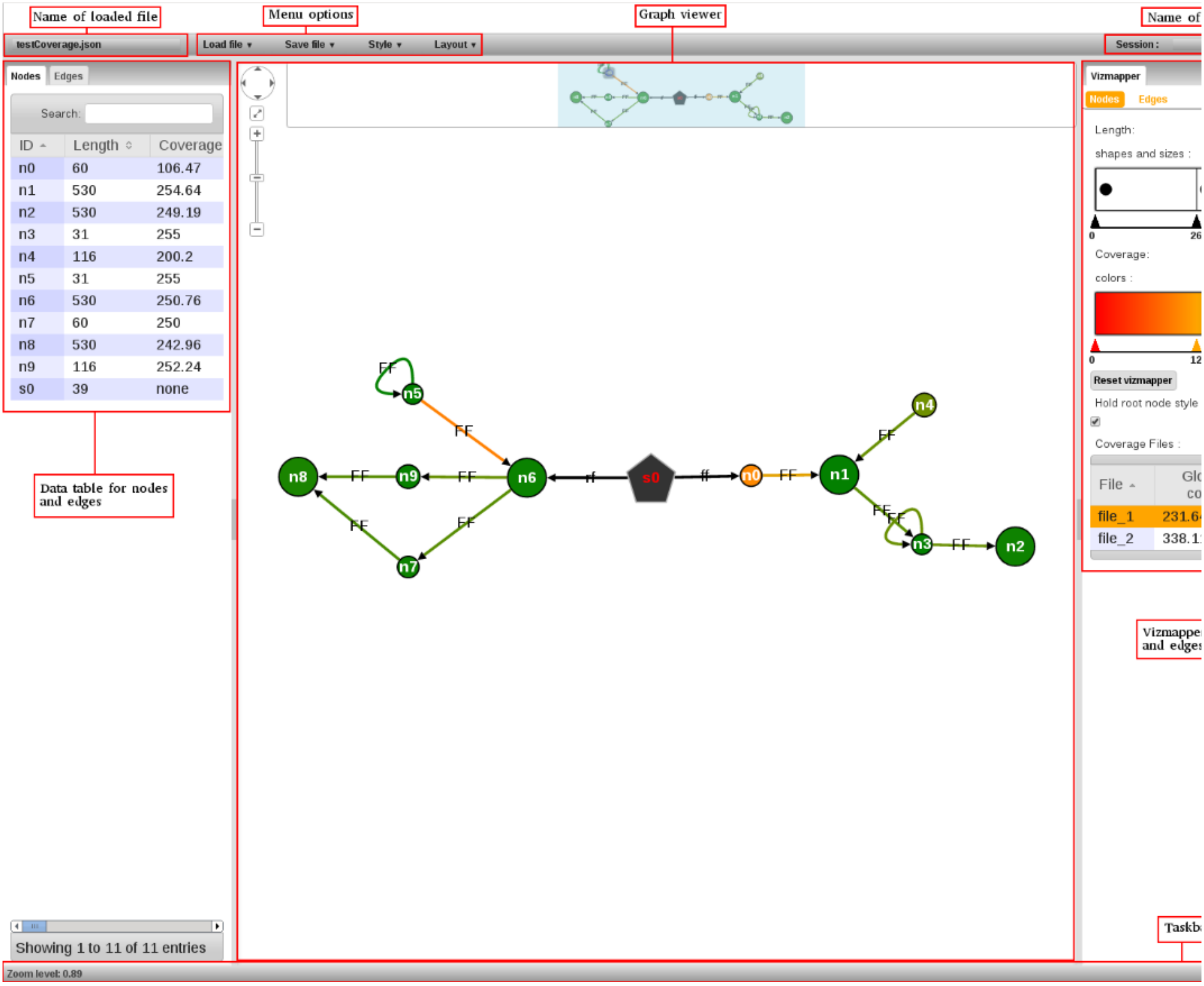
b. Sub-starters



ID	Right graph	Left graph	Length	Sequence
s0	k0	k1	39	AAGTAAACGGTAGCAAAGAGGATCTGTTATAGATCAAA

Showing 1 to 1 of 1 entries

Click on one of them selecting and displaying the graph viewer in a new browser tab.

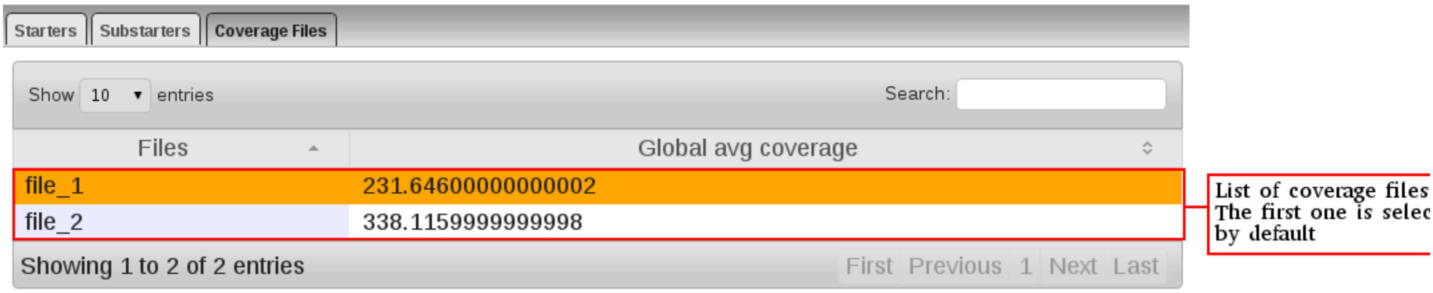


ID	Length	Coverage
n0	60	106.47
n1	530	254.64
n2	530	249.19
n3	31	255
n4	116	200.2
n5	31	255
n6	530	250.76
n7	60	250
n8	530	242.96
n9	116	252.24
s0	39	none

Showing 1 to 11 of 11 entries

c. Coverage Files

Set the coverage file, need to be done before selected a substarter. By default, first coverage files is considered.


Starters
Substarters
Coverage Files
Show 10 ▾ entries
Search:
First Previous 1 Next Last
List of coverage files The first one is selec by default

Files	Global avg coverage
file_1	231.64600000000002
file_2	338.1159999999998

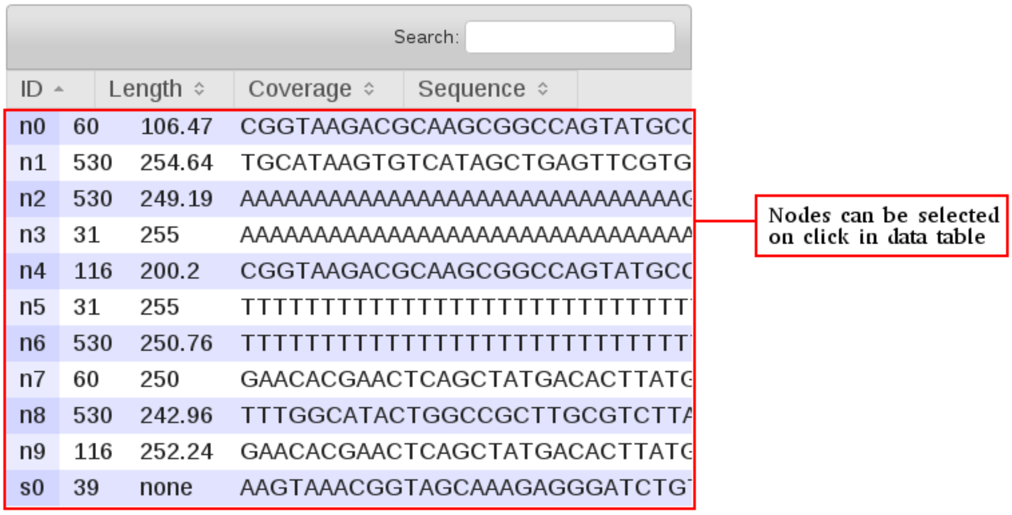
Showing 1 to 2 of 2 entries

2. Graph viewer

2.1. Data tables

In the graph viewer, part left contains nodes data table and edges data table in two tabs. In this area, data elements are displayed:

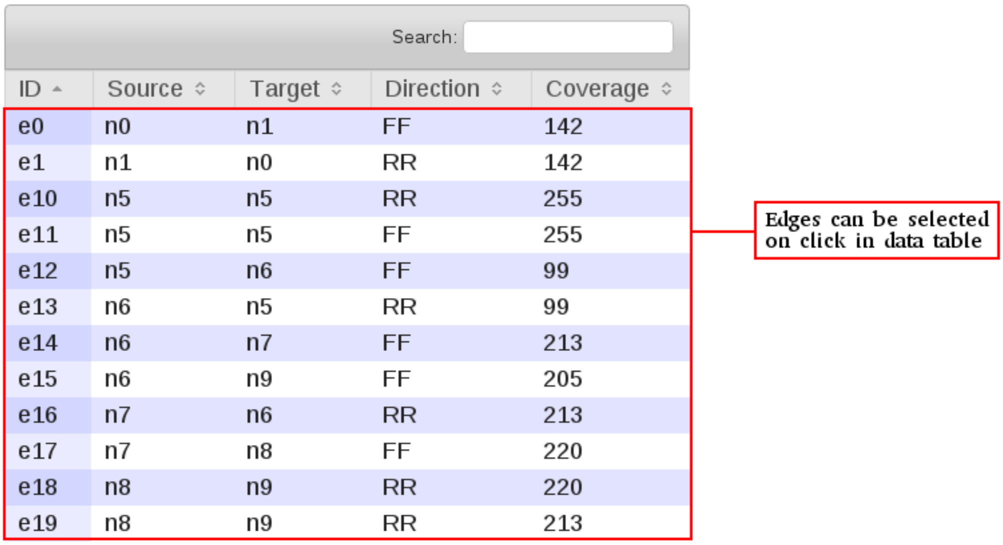
- For nodes : ID, length, sequence and average coverage(if it's present in json).
- For edges: ID, source, target and average coverage (if it's present in json).



Search:

ID	Length	Coverage	Sequence
n0	60	106.47	CGGTAAGACGCAAGCGGCCAGTATGCC
n1	530	254.64	TGCATAAGTGTCATAGCTGAGTTCGTG
n2	530	249.19	AAAAAAAAAAAAAAAAAAAAAAAAAAAG
n3	31	255	AAAAAAAAAAAAAAAAAAAAAAAAAAAA
n4	116	200.2	CGGTAAGACGCAAGCGGCCAGTATGCC
n5	31	255	TTTTTTTTTTTTTTTTTTTTTTTTTTTT
n6	530	250.76	TTTTTTTTTTTTTTTTTTTTTTTTTTTT
n7	60	250	GAACACGAACTCAGCTATGACACTTATG
n8	530	242.96	TTTGGCATACTGGCCGCTTGCGTCTTA
n9	116	252.24	GAACACGAACTCAGCTATGACACTTATG
s0	39	none	AAGTAAACGGTAGCAAAGAGGGATCTG

Nodes can be selected on click in data table



Search:

ID	Source	Target	Direction	Coverage
e0	n0	n1	FF	142
e1	n1	n0	RR	142
e10	n5	n5	RR	255
e11	n5	n5	FF	255
e12	n5	n6	FF	99
e13	n6	n5	RR	99
e14	n6	n7	FF	213
e15	n6	n9	FF	205
e16	n7	n6	RR	213
e17	n7	n8	FF	220
e18	n8	n9	RR	220
e19	n8	n9	RR	213

Edges can be selected on click in data table

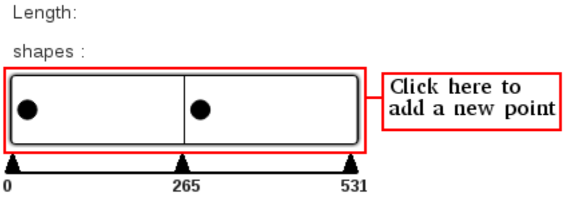
Select/unselect one of the elements in table select/unselect this element in the graph. Multi selection is possible with hot key "Shift+Left Click". The search, is a functionality very useful, allow to find an elements by his ID or find nodes with specific motif.

2.2. Vizmapper

The vizmapper allow to define style in function of the properties length and average coverage. For nodes, it's possible to define shape and size in function of length. If average coverage is present, edges tab has been appeared and it's possible to define color in fonction of average coverage for nodes and edges. Each cursor define a point in the distribution of nodes length or nodes/edges average coverage. By default, the style of the root is holded.

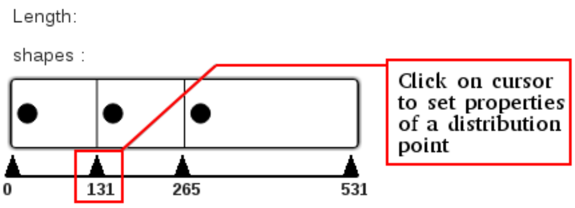
a. Length property

Points in distribution of nodes length define points in distribution of shapes (discrete) and in distribution of size (gradient). By default, shape is round for all nodes except the root and the distribution of sizes contains three points with the size :20px (minimum length), 30px (medium length), 40px(maximum length). For example, with the default values of size, nodes have sequence length between 0 and medium values have a size values between 20px and 30px. Size values grow up gradually with values of sequences length.



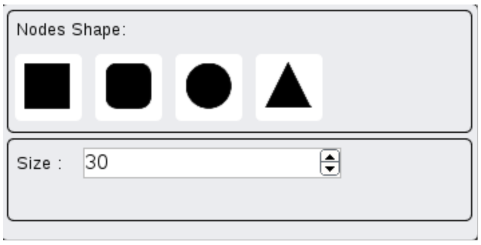
Click here to add a new point

Click on preview of distribution add a new point (and a new cursor) in the distribution.



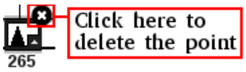
Click on cursor to set properties of a distribution point

Click on a cursor display a selector to allow to set the properties (shape , size) of the distribution point.



Size : 30

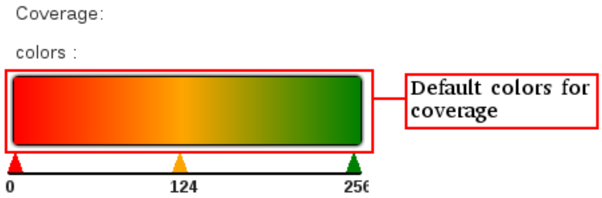
This action display a frame with a cross arround the cursor too. Click on the cross delete this distribution point (and the cursor).



Click here to delete the point

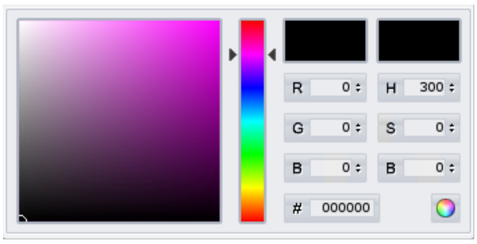
b. Coverage property

If coverage is present, points in distribution of average coverage define points in distribution of colors (gradient). By default, the distribution of colors contains three points : red (poor coverage), orange (medium coverage), green (good coverage).



Default colors for coverage

Similary to [lenght property](#), it's possible to add, remove and set style (here the color). For example, with the default values of color, nodes have average coverage between 0 and medium values have a color values between red and orange with a gradient transition between these.

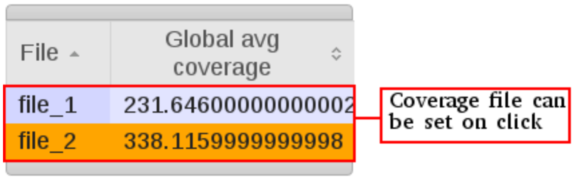


R 0
G 0
B 0
H 300
S 0
B 0
# 000000

c. Reset and table of coverage files

The reset button reset the vizmapper to default state for nodes or for edges (not the two in same time).

The table of coverage files, allow to set the coverage files consedered for the graph. The orange one is the selected one. Click on an other one have set data table (nodes and edges), and vizmapper has been updated with new values of average coverage.



File	Global avg coverage
file_1	231.64600000000002
file_2	338.1159999999998

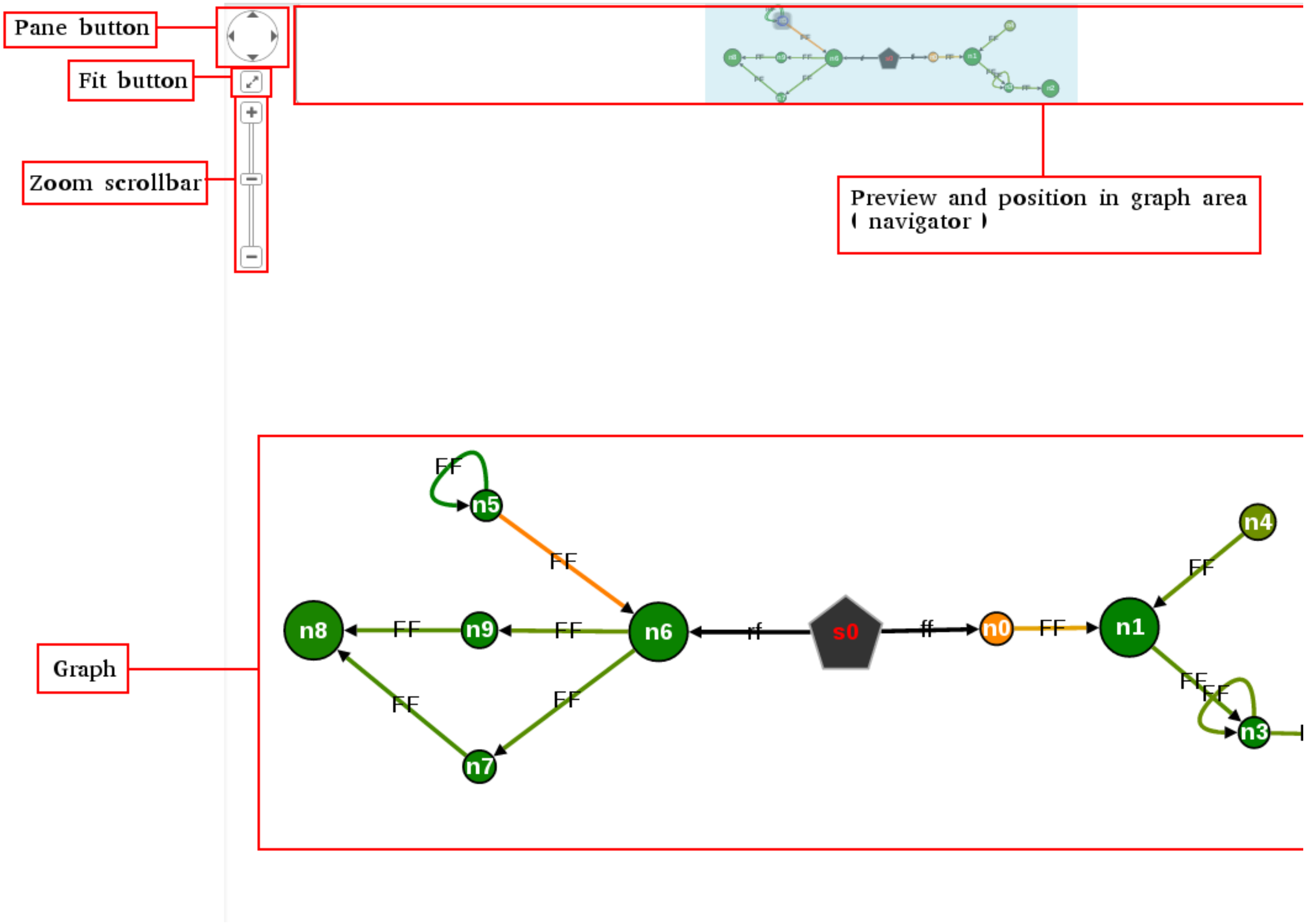
Coverage file can be set on click

2.3. Graph panel

The graph viewer, allow to:

- Select one element on click
- Select several elements with the hotkey "Shift+Left Click" or "Left Click pushed + Mouse move" (selection box)
- Move nodes with drag and drop
- Zoom/Dezoom with mousewheel, click on zoom scrollbar
- Navigate with "long Left Click pushed (1 sec) + Mouse move" or with navigation button

- Preview the global graph in navigator frame
- Fit the graph with fit button



a. Load file

Load file button allow to return to start page to set starter and substarter, and allow to load a previous session file.

b. Save file

Save file button allow to save current session in a file. The session needs to be save in same directory as the json loaded else session has not been able to restore after.

c. Style

The style button allow to show/hide labels of graph elements (nodes and edges).

d. Layout

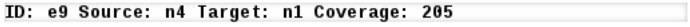
The layout button allow to recalculate the current layout (reset), and set type of layout. Nine types of layout are available:

- Null : all nodes in the same positions
- Preset : fit nodes
- Random : random nodes positions
- Grid : nodes into a grid
- Arbor : force directed layout
- Circle : nodes on a circle
- Cose : group by nodes similarities
- BreadthFirst : layout with breadthfirst algorythm
- Hierarchical : hierarchical layout (in dev)

2.4. Data viewer

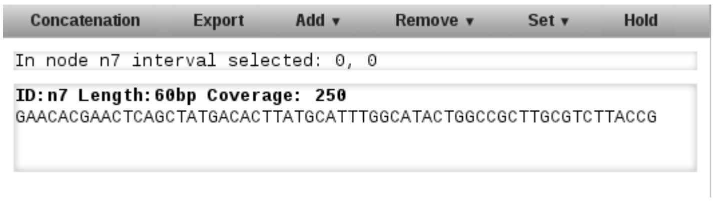
Select nodes displaying a bottom panel that contains properties of the elements selected. The panel give:

For edges : ID, source, target, average coverage.



```
ID: e9 Source: n4 Target: n1 Coverage: 205
```

For nodes : ID, length of sequence, average coverage and sequence. The panel have a menu and display the interval of selection for the current node.



```
In node n7 interval selected: 0, 0
```

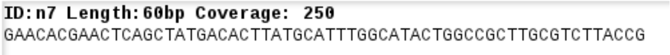
```
ID:n7 Length:60bp Coverage: 250
GAACACGAACTCAGCTATGACACTTATGCATTTGGCATACTGGCCGCTTGCGTCTTACCG
```

The panel is close, after elements are unselected. The hold button allow to keep open the panel after elements are unselected.

a. Sequence format

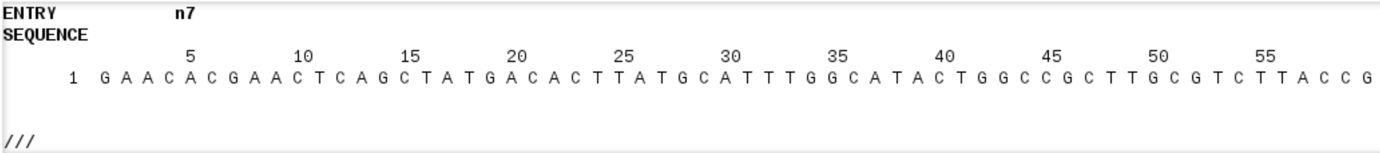
It's pssible to set format of sequence(s) displayed (Set button). Four formats are available:

FASTA



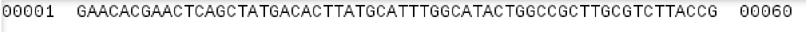
```
ID:n7 Length:60bp Coverage: 250
GAACACGAACTCAGCTATGACACTTATGCATTTGGCATACTGGCCGCTTGCGTCTTACCG
```

CODATA



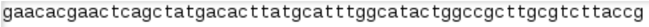
```
ENTRY           n7
SEQUENCE
                5         10        15        20        25        30        35        40        45        50        55
      1  G A A C A C G A A C T C A G C T A T G A C A C T T A T G C A T T T G G C A T A C T G G C C G C T T G C G T C T T A C C G

///
```

PRIDE



```
00001   GAACACGAACTCAGCTATGACACTTATGCATTTGGCATACTGGCCGCTTGCGTCTTACCG   00060
```

RAW

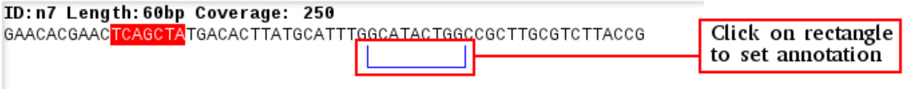


```
gaacacgaactcagctatgacacttatgcatttggcatactggccgcttgcgtcttaccg
```

b. Annotation and highlight

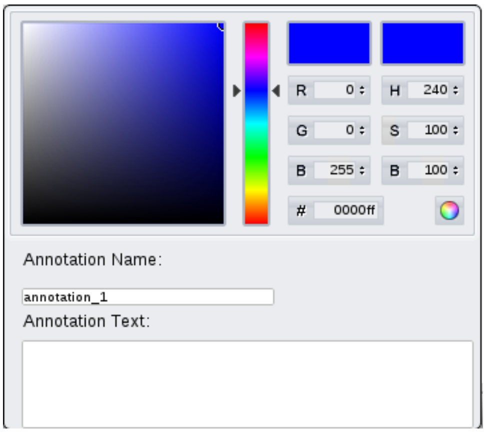
It's possible to spot part(s) of sequence(s) with highlight or annotation (Add button):

- To add highlight select part of sequence and click on "Highlight". The selected part of sequence become highlight with red color.
- To add annotation select part of sequence and click on "Annotation". Under the part of selected sequence a rectangle is displayed.



```
ID:n7 Length:60bp Coverage: 250
GAACACGAACTCAGCTATGACACTTATGCATTTGGCATACTGGCCGCTTGCGTCTTACCG
```

Click on annotation rectangle displaying a color selector with text area to define name of the annotation and a commentary.



R 0
G 0
B 255
H 240
S 100
B 100
0000ff

Annotation Name:
annotation_1
Annotation Text:

After the selector is hidden, name and commentary are shown on mouse over annotation rectangle.

The highlight and annotation are persistant, so if nodes are unselected these have not been loose. For now, it's only possible to remove all highlight or/and all annotation displayed (Remove button).

c. Export

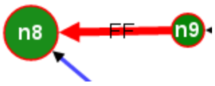
The export function allow to generate and download a text file, contains all sequences displayed in bottom panel, in selected format.

d. Concaténation

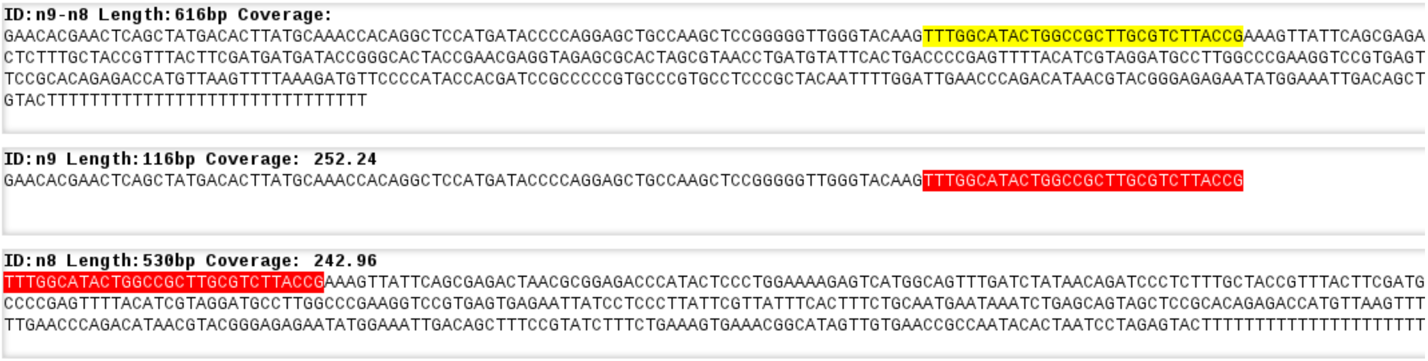
The concatenation function allow to concatenate sequence of tow nodes or more(available only for Mapsembler json output or minimal json with k in data nodes). This function involve some constraints to allow or disallow the concatenation of sequences. It's important to know that if tow nodes or link with an edge, there are n characteres similar (overlap) in the two sequences:

- Edge direction is FF (forward-forward): The last k-1 characters of the squences of nodes source is identical to the first k-1 characters of the sequence of the target nodes.
- Edge direction is RR (reverse-reverse): The last k-1 characters of the reverse complement of the squences of nodes source is identical to the first k-1 characters of the reverse complement of the sequence of the target nodes.
- Edge direction is RF (reverse-forward): The last k-1 characters of the reverse complement of the squences of nodes source is identical to the first k-1 characters of the sequence of the target nodes.
- Edge direction is FR (forward-reverse): The last k-1 characters of the squences of nodes source is identical to the first k-1 characters of the reverse complement of the sequence of the target nodes.

To concatenat two nodes the order of click is very important because only just one direction is shown on the graph but in some cases an other direction exist. (This can be checked in edges table).



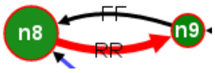
So for example if click on n9 nodes first, and after click on n8. The direction followed is FF.



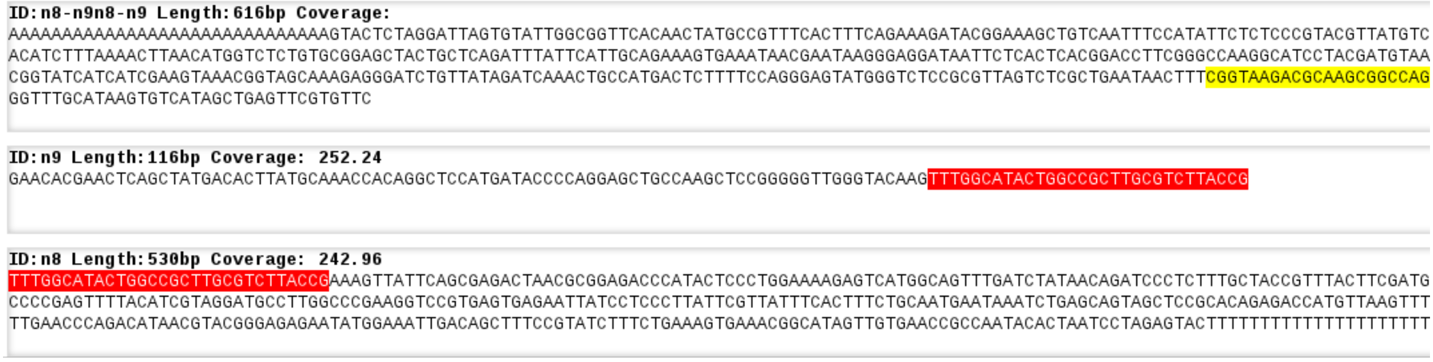
```
ID:n9-n8 Length:616bp Coverage:
GAACACGAACTCAGCTATGACACTTATGCAAACCACAGGCTCCATGATACCCCAGGAGCTGCCAAGCTCCGGGGGTTGGGTACAAGTTTGGCATACTGGCCGCTTGCGTCTTACCGAAAGTTATTCAGCGAGA
CTCTTTGCTACCGTTTACTTCGATGATGATACCGGGCACTACCGAACGAGGTAGAGCGCACTAGCGTAACCTGATGTATTCACTGACCCCGAGTTTTACATCGTAGGATGCCTTGGCCCGAAGGTCCGTGAGT
TCCGCACAGAGACCATGTTAAGTTTTAAAGATGTTCCCCATACCACGATCCGCCCCCGTGCCCGTGCCTCCCGCTACAATTTTGGATTGAACCCAGACATAACGTACGGGAGAGAATATGGAAATTGACAGCT
GTACTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
```

```
ID:n9 Length:116bp Coverage: 252.24
GAACACGAACTCAGCTATGACACTTATGCAAACCACAGGCTCCATGATACCCCAGGAGCTGCCAAGCTCCGGGGGTTGGGTACAAGTTTGGCATACTGGCCGCTTGCGTCTTACCG
```

```
ID:n8 Length:530bp Coverage: 242.96
TTTGGCATACTGGCCGCTTGCGTCTTACCGAAAGTTATTCAGCGAGACTAACGCGGAGACCCATACTCCCTGGAAAAGAGTCATGGCAGTTTGATCTATAACAGATCCCTCTTTGCTACCGTTTACTTCGATG
CCCCGAGTTTTACATCGTAGGATGCCTTGGCCCGAAGGTCCGTGAGTGAGAATTATCCTCCCTTATTCGTTATTTCACTTTCTGCAATGAATAAATCTGAGCAGTAGCTCCGCACAGAGACCATGTTAAGTTT
TTGAACCCAGACATAACGTACGGGAGAGAATATGGAAATTGACAGCTTTCCGTATCTTTCTGAAAGTGAAACGGCATAGTTGTGAACCGCCAATACACTAATCCTAGAGTACTTTTTTTTTTTTTTTTTTTTT
```



But if click on n8 before n9, the direction follow is RR.
So the result of concatenation will be different.



```
ID:n8-n9n8-n9 Length:616bp Coverage:
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGTACTCTAGGATTAGTGTATTGGCGGTTCACAACTATGCCGTTTCACTTTCAGAAAGATACGGAAAGCTGTCAATTTCCATATTCTCTCCCGTACGTTATGTC
ACATCTTTAAAACTTAACATGGTCTCTGTGCGGAGCTACTGCTCAGATTTATTCATTGCAGAAAGTGAAATAACGAATAAGGGAGGATAATTCTCACTCACGGACCTTCGGGCCAAGGCATCCTACGATGTAA
CGGTATCATCATCGAAGTAAACGGTAGCAAAGAGGGATCTGTTATAGATCAAACTGCCATGACTCTTTTCCAGGGAGTATGGGTCTCCGCGTTAGTCTCGCTGAATAACTTTCGGTAAGACGCAAGCGGCCAG
GGTTTGCATAAGTGTCATAGCTGAGTTCGTGTTC
```

```
ID:n9 Length:116bp Coverage: 252.24
GAACACGAACTCAGCTATGACACTTATGCAAACCACAGGCTCCATGATACCCCAGGAGCTGCCAAGCTCCGGGGGTTGGGTACAAGTTTGGCATACTGGCCGCTTGCGTCTTACCG
```

```
ID:n8 Length:530bp Coverage: 242.96
TTTGGCATACTGGCCGCTTGCGTCTTACCGAAAGTTATTCAGCGAGACTAACGCGGAGACCCATACTCCCTGGAAAAGAGTCATGGCAGTTTGATCTATAACAGATCCCTCTTTGCTACCGTTTACTTCGATG
CCCCGAGTTTTACATCGTAGGATGCCTTGGCCCGAAGGTCCGTGAGTGAGAATTATCCTCCCTTATTCGTTATTTCACTTTCTGCAATGAATAAATCTGAGCAGTAGCTCCGCACAGAGACCATGTTAAGTTT
TTGAACCCAGACATAACGTACGGGAGAGAATATGGAAATTGACAGCTTTCCGTATCTTTCTGAAAGTGAAACGGCATAGTTGTGAACCGCCAATACACTAATCCTAGAGTACTTTTTTTTTTTTTTTTTTTTT
```

Be carefule on unselected nodes, concatenation disapear and all annotation or highlight are loose (for concatenated sequence). If two (or more) nodes can't be concatenated, for example tow nodes not linked, an error message has been displayed in the corner right of the application.