
Subject: Re: can you give an example on how to use refvalue()...

Posted by [thomas](#) on Thu, 02 Jul 2015 19:56:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dear Axel,

the refvalue(ref-ID-column, category-column, value-column, ref-ID) syntax would be:
refvalue("B","A","C","xxxx")

Your second question is more difficult to answer: DataWarrior out-of-the-box doesn't have a simple solution.

But with a little programming it should be possible to even generate such a tree of hierarchical increasing scaffolds contained in a dataset and to attach all compounds as leaves.

Currently, DataWarrior does support graph data, which includes tree structures. Various functions generate such data, e.g. the evolutionary library and most prominent the similarity analysis. Last week I used an unpublished and still experimental extension to layout a graph from a few thousand diseases connected over their similarities, which were estimated by commonly mentioned genes in the literature. But DataWarrior does not allow from the user interface to define a column to contain references to other rows.

If you want to construct a dwar file from tabular data in a way that DataWarrior recognized graph relationships, then you have to do it manually. First you need a column with unique row identifiers.

Second you need another column that contains references of the respective row to one or more other rows.

This is done by putting unique IDs from the first row into the second one, one ID per reference, delimited

by "; ". Third you need to define the second row as 'referencing column' and state whether references are

bi-directional or only one way. This is done by adding column properties to the file; Example:

```
<columnName="child">
```

```
<columnProperty="refColumn ID">
```

```
<columnProperty="refType topdown">
```

I have created a small dwar file from your example that illustrates the options. The file contains three different tree views: one with static coordinates of all structures and two dynamic one, which always use the selected molecule as root, both use a kind of forcefield approach for the layout (circular and standard vertical tree)

Kind regards,

Thomas

File Attachments

1) [fragmentHierarchy.dwar](#), downloaded 981 times
