# Integration of -omics data and networks for biomedical research with VANTED

**Christian Klukas**[1] **and Falk Schreiber**[1,2]

[1]Leibniz Institute of Plant Genetics and Crop Plant Research, Corrensstr. 3, 06466 Gatersleben, Germany

[2] Martin Luther University Halle-Wittenberg, 06099 Halle, Germany

klukas@ipk-gatersleben.de, schreibe@ipk-gatersleben.de

**Abstract**

Increasingly, research focus in the fields of biology and medicine moves from the investigation of single phenomena to the analysis of complex cause and effect relationships. The clarification of complicated relationships requires the consideration of different domains, such as gene expression, protein, and metabolite data. Furthermore, it is often sensible not to analyze the collected data in isolation, but to consider the context of relevant biological networks. In this paper newly developed functionalities of the VANTED system are presented. They allow users from medicine and biology to interactively structure extensive experimental data, to filter, to evaluate, and to visualize the data and the analysis results in the context of biological networks and classification hierarchies.

## 1 Introduction

The methodology of biochemical research has strongly changed during the last years. Nowadays large amounts of experimental data is produced by massive-parallel analysis technologies, for instance by automated enzyme-assays, metabolite- and transcript-profiling. Using the right supporting software, the resulting data base provides a comprehensive view on the biochemistry of an organism. The clarification of complicated connections in organisms generally requires the consideration of different domains. To handle this problem, instead of analysing the data in isolation, it is worth to consider the context of relevant biological networks. Available software systems for this task (see [5]) are tuned besides a few exceptions to single data domains and/or are firmly coupled to certain databases. In this paper newly developed functionalities of the VANTED system [2] are presented. They allow users from medicine and biology to interactively work with extensive experiment data, to filter, to statistically evaluate, and to visualize the analysis results directly in context of relevant biological networks and classification hierarchies.

## 2 Methods

For the analysis of experimental data integrated views of the measured values and relevant background information should be generated. This approach corresponds with the idea of system biology – instead of considering single parts, the analysis covers the overall system with

all interactions to better understand biological phenomena. In order to fulfill the goal of creating a software system which supports users in such an integrated analysis, three aspects are of importance for the design of the VANTED software: 1) data models for experiment data and biological networks, 2) the process of data mapping, by means of connecting experiment data and networks, 3) the analysis and visualization of the network-integrated data sets. These three points are described in the following.

**1) Data models for experiment data and biological networks** By investigation of common experiment designs the following crucial experimental factors have been identified: information about time series, replicates, environmental conditions, treatments and genetic lines. A data model which is able to handle experiment data, partitioned by the mentioned experiment factors, has been developed and is shown in Figure 1. To simplify the design and implementation, the model does not store information about the experiment procedures, but instead focuses on information required for experiment data mapping, visualization and analysis.

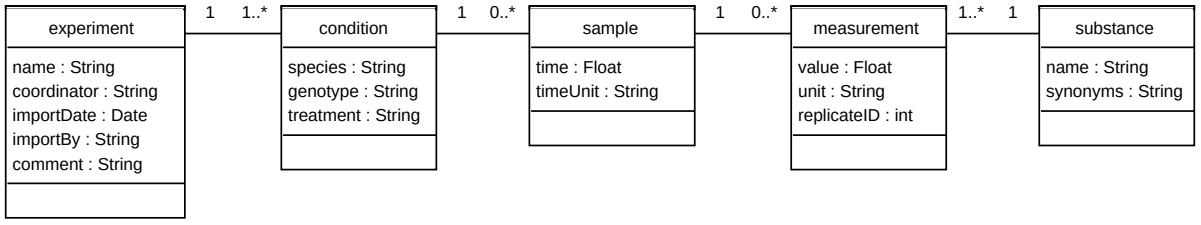


**Figure 1: Data model for experiment data sets.**

In contrast to some other systems VANTED supports dynamic networks. Networks can be loaded into the system from databases (e. g. KEGG) or from files (e. g. GML, SBML, Pajek .net format). In addition, it is possible to construct or edit networks manually with integrated editor functions, thus networks can be easily extended if more substances were measured. The following graph data model is used as a flexible basis for different kinds of biological networks as well as for classification hierarchies:

The mapping-graph MG is defined based on a graph $G = (V, E)$ with a set of nodes $V$ and a set of directed or undirected edges $E$, and additionally, a set of labels $L$, sets of node- and edge-types $T_V$ and $T_E$, a set of experiment data $M$, the label function $l : V, E \mapsto L$, the node- and edge-type functions $t_V : V \mapsto T_V$, $t_E : E \mapsto T_E$, the data mapping function $z : V, E \mapsto M$. This *mapping-graph* $MG$ is written as $MG = (V, E, l, t_V, t_E, z)$.

Depending on the biological network under investigation, different sets of node- and edge-types are used. Two examples:

(1) For a protein-protein interaction mapping-graph $MG_{PPI}$, the following node- and edge-types are used: $T_V = \{Protein\}$ and $T_E = \{Interaction\}$. Each edge of the modelled graph $MG_{PPI}$ stands for an undirected interaction between two proteins, represented by the end points of the particular edge.

(2) A KEGG-mapping-graph $MG_{KEGG}$ uses the following node- and edge-types: $T_V = \{$ *Orthologue, Enzyme, Gene, Gene-Group, Metabolite, Pathway-Link*$\}$ and $T_E = \{$*ECrel, PPrel,*

*GErel, PCrel, Link*}. A description of the KGML format, which stores information about a particular KEGG pathway, the listed node- and edge-types as well as the transformation into a dynamic graph model has been published previously [4].

**2) Data mapping** For the integration of measurement data into relevant networks a data mapping is carried out. Given a mapping-graph $MG$ and a set of experimental data sets $ED$ (according to the definition shown in Figure 1), from $ED$ derived subsets of the experiment data are connected to elements of $MG$, using the following algorithm. The result of this algorithm corresponds to the data function $z$ of $MG$.

**Algorithm 1** Data mapping

**Require:** $MG$ – mapping-graph
**Require:** $ED$ – experiment data, objects of type *experiment*
1: $M \Leftarrow$ generate for each substance in $ED$ a separate data set
2: **for all** graphelements $ge \in MG$ **do**
3: $Z \Leftarrow \emptyset$
4: $A \Leftarrow label(ge) \cup synonyms(ge)$
5: **for all** $m \in M$ **do**
6: $B \Leftarrow id(m) \cup synonyms(m)$
7: **if** $|A \cap B| > 0$ **then**
8: $Z \Leftarrow m \cup Z$
9: **end if**
10: **end for**
11: $z(ge) = Z$
12: **end for**

In the beginning of the data mapping algorithm (line 1), the given exerimental data sets are partitioned into multiple data sets. For each substance in the data sets $ED$ new data sets are constructed, containing only the measurement data and corresponding experiment info, which is related to a particular substance. Two sets $A$, $B$ are initialized (lines 4 and 6). $A$ contains the substance identifier ($id(m)$) and corresponding synonyms, set $B$ contains the graphelement label and additionally defined or derived synonyms. Information about synonyms is taken automatically from the integrated databases (Expasy Enzyme [1], KEGG Compound and KEGG BRITE [3]) or can be provided by the user. For each node and edge in $MG$ (line 2 and 4), it is tested, whether the intersection of set $A$ and $B$ contains at least one element (line 7). Is this the case, the data set $m$ (containing experiment data of one substance) is included in the set of data to be mapped ($Z$, line 8). After checking all substance data sets $m \in M$ the mapped data for a particular graph element is defined (line 11).

Optionally (not shown in Algorithm 1), data sets which could not be mapped on the basis of substance names and synonyms are mapped to newly generated network nodes. In this manner new substances can be easily integrated into an existing network.

**3) Histogram functions for classification hierarchies and network-integrated data** The basis of the histogram functions are classification hierarchies modelled as graphs (e. g. Gene Ontology or KEGG BRITE) consisting of classification nodes $CN$ and leaf-nodes $LN$