

Towards Readable Layouts for Modeling Data Warehouses* (PREPRINT)

Jesús Pardillo
University of Alicante – DLSI/Lucentia, Spain
jesuspv@ua.es

Florian Mansmann
University of Konstanz, Germany
Florian.Mansmann@uni-konstanz.de

ABSTRACT

Data warehouses are large-scale databases that are usually managed by means of diagram-based conceptual models. However, the complexity of those models often imposes significant design challenges. In particular, this article studies their different underlying graph layouts. The working hypothesis is that graph layouts influence diagram readability, with the latter being significant for facilitating the design process. We define the main viewpoints involved in conceptual modeling. For each one, surveyed as well as alternative layouts were evaluated against a set of aesthetics and efficiency measures. As a result, more readable graph layouts than those found in the literature were identified.

Categories and Subject Descriptors

D.1.7 [Software]: Programming Techniques—*Visual Programming*; H.2.3 [Information Systems]: Database Management—*Languages, Data description languages (DDL)*

General Terms

Design, Languages, Theory

Keywords

Conceptual modeling, Data warehouses, Multidimensional modeling, OLAP, Visualization

1. INTRODUCTION

Data warehouses are databases that store historical data for decision-making purposes [7], with support for OLAP

*Supported by J. Pardillo’s FPU grant AP2006-00332 and the MESOLAP (TIN2010-14860) project from the Spanish Ministry of Science and Innovation. Special thanks to Svetlana Mansmann and Jose-Norberto Mazón for reviewing the draft of this article, and to DOLAP’s anonymous reviewers, who gave us the opportunity to discuss our findings with the DOLAP community.

(*on-line analytical processing*) queries, trend analysis, or data clustering, among others. In a corporate environment, a data warehouse¹ may be very hard to manage, whereas conceptual models provide high-level abstractions and proper documentation for facilitating the design, understanding, and management of such databases [13].

However, the readability of (diagram-based) conceptual models may be aggravated through disadvantageous design decisions with respect to their concrete syntax (*i.e.*, diagrams), resulting in an increased data warehouse design overhead. Although “the bandwidth of information presentation is potentially higher in the visual domain than for media reaching any of the other senses” [17], further research is necessary in order to understand the principles “that will help the field cross the chasm to wider success” [11]. Data warehousing is thereby not an exception [10], which strongly relies on both data and metadata visualization.

Some authors have already attempted to manage the complexity of the conceptual models of data warehouses by defining different abstraction levels. An archetypal example is due to [8], where UML packages modularize the visualization of the otherwise flattened conceptual model, but also to [1], which provides viewpoints for similar abstraction levels. Moreover, others works, such as [16], study some metrics on the structural complexity of data-warehouse models to predict their understandability.

The solution presented in this article aims to handle the complexity of data warehouses by means of OLAP viewpoints (*i.e.*, keeping current metamodels unaltered) that are defined (§2.1) to optimize diagram readability. OLAP diagrams are then characterized as graph layouts (§2.2), for which several aesthetics and efficiency measures can be associated (§2.3), thus supporting the evaluation of the OLAP-diagram readability. Then, both the graph layouts presented in the literature and the proposed alternatives are evaluated (§3.1). This study builds the foundation to understand the most readable graph layouts for each of the defined OLAP viewpoints (§3.2). The major conclusions of this study highlight the prevailing readability bottle-necks in the conceptual modeling of data warehouses and the benefits of introducing new layouts. Implementation of the latter would enable practitioners to visually manage data warehouses of much higher complexity than currently supported.

1.1 Preliminary Notions

Conceptual models may be characterized as *formal lan-*

¹Henceforth, the term ‘data warehouse’ refers to this kind of OLAP database.

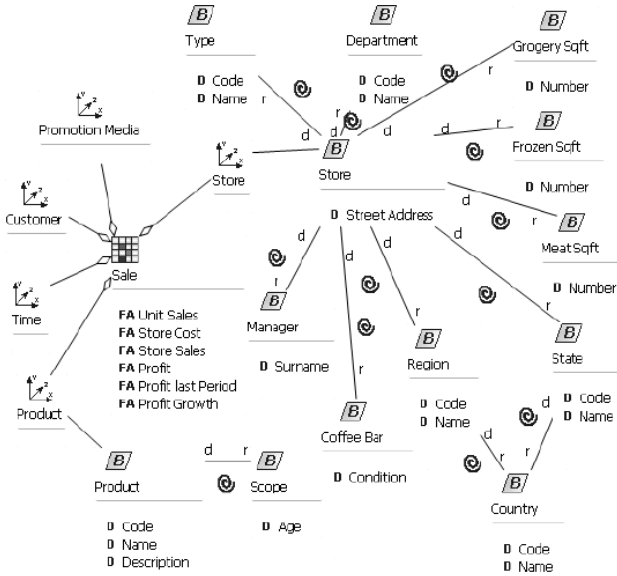



Figure 1: An excerpt of an archetypal conceptual data warehouse model for OLAP

guages, which are built on only syntactic elements, *i.e.*, pure abstract formulae, without attached interpretation (*semantics*) or representation (*notation*) [6]. However, visual-language syntaxes may be further decomposed into: the *abstract syntax* [15], which defines the modeling elements and their relationships without references to graphical primitives, and the *concrete syntax* or notation [2], which defines such graphics and their connection to the abstractions to be perceived. Moreover, diagram-based notations may be formally characterized as *graphs*. Then, the concrete syntax may be additionally decomposed into: the graphical primitives that render diagram nodes and links (the *primary* notation) and the graph layout (the *secondary* notation [5]), on which this study is focused.

1.2 Motivating Example

Since there are several conceptual data warehouse models that have been proposed in the literature [13], we have selected one of them [9] as an archetypal example in order to illustrate the aforementioned readability problems. Such an archetype was obtained from the analysis and comparison of the abstract syntaxes that characterize the state of the art of conceptual modeling. The research proposals studied were extracted from the survey in [14]. After a filtering process based on the presented diagram suitability, the qualifying works are as follows (identified by the first author's name): Cabibbo's, Golfarelli's, Hüsemann's, Bonifati's, Phipps', Prat's, Abelló's, and Luján-Mora's.

Fig. 1 shows an OLAP diagram, which is based on the UML profile of [9]. Briefly, the data warehouse scenario modeled there stores *sales* as a *fact* (identified by the icon ) to be analyzed along several *dimensions* ($\langle \mathcal{D} \rangle$), *e.g.*, *product*, *customer*, or *store*. For such analysis, fact *measures* (**FA**) such as *unit sales* or *profit* are aggregated along some *levels* ($\langle \mathcal{B} \rangle$) such as those of the *roll-up* (\odot) path *store*, *state*, and *country* (the association end 'd' refers to the *drill-down* path). Each level can be displayed by some *attribute* (**D**), *e.g.*, *manager's surname*. As shown in Fig. 1,

the conceptual model gets rather complex even when only two dimensions (of a total of five in this case) are presented.

Since such conceptual models are the core design artifacts in many data-warehousing methods, their readability should be assessed and optimized to minimize the design efforts.

2. ANALYSIS FRAMEWORK

The example of Fig. 1 also points out the view-based *separation of concerns* that can be achieved by means of viewpoints, which this study is based on. Indeed, for the study of diagram readability, the main viewpoints for conceptual data warehouse modeling are defined (§2.1), together with the characterization of the graph layouts involved (§2.2) and the readability measurement framework to apply (§2.3).

2.1 OLAP Viewpoint Taxonomy

OLAP diagrams are decomposed into *viewpoints* [4]. Each viewpoint renders OLAP modeling elements (*e.g.*, facts, dimensions, or any of their relationships) that are managed at once by data warehouse designers. The viewpoints are intuitively defined according to our experience with data modeling and visualization, covering all the studied OLAP metadata. However, the set of viewpoints was not conceived to be minimal, *i.e.*, some metadata may appear from multiple viewpoints. This is due to the fact that the same metadata could be involved in various design activities advocating a different viewpoint. The viewpoints were also inspired by the task taxonomy for visual information seeking [17], in particular, the *relate* task ("View relationships among items").

Viewpoints were also classified by complexity: *primitive* (views on atomic modeling elements), *relational* (on relationships between modeling elements), and *hierarchical* (on aggregation paths). Primitive viewpoints are listed next:

Attribution of a Level (denoted as $\mathcal{A}\vartheta(l)$) is the collection of attributes of some level, *e.g.* (see Fig. 1),

$$\mathcal{A}\vartheta(\text{Product}) = \{\text{Code}, \text{Name}, \text{Description}\}.$$

Granularity of a Fact (denoted as $\mathcal{L}\vartheta(f)$) is the collection of the *defining* levels of some fact, *e.g.*,

$$\mathcal{L}\vartheta(\text{Sale}) = \{\text{Customer}, \text{Product}, \dots, \text{Time}\}.$$

Measurement of a Fact (denoted as $\mathcal{M}\vartheta(f)$) is the collection of measures of some fact, *e.g.*,

$$\mathcal{M}\vartheta(\text{Sale}) = \{\text{Unit Sales}, \dots, \text{Profit Growth}\}.$$

Relational viewpoints are as follows:

Attribution of a Dimension (denoted as $\mathcal{L}^{\times \mathcal{A}}\vartheta(d)$) is the relation between levels and their attributes for a particular dimension, *e.g.*,

$$\mathcal{L}^{\times \mathcal{A}}\vartheta(\text{Product}) \supset \{\langle \text{Product}, \text{Code} \rangle, \langle \text{Scope}, \text{Age} \rangle\}.$$

Dimensionality (denoted as $\mathcal{F}^{\times \mathcal{D}}\vartheta$) is the relation between facts and the dimensions associated. Each dimension is related by means of the defining level, *e.g.*,

$$\mathcal{C}^{\times \mathcal{D}}\vartheta \supset \{\langle \text{Sale}, \text{Customer} \rangle, \langle \text{Sale}, \text{Product} \rangle\}.$$

The unique hierarchical viewpoint is defined as follows:

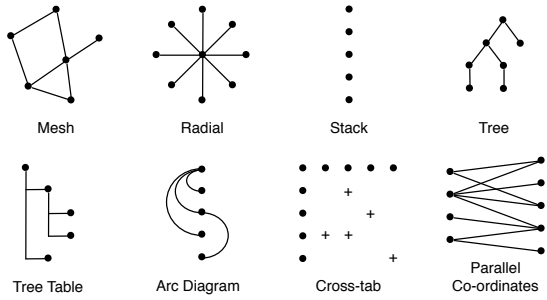


Figure 2: Graph layouts (surveyed and proposed) that characterize OLAP diagrams

Aggregation Hierarchies of a Dimension ($L \times L^n \vartheta(d)$) is the collection of the aggregation hierarchies (relations between levels) of some dimension, *e.g.*,

$$L \times L^n \vartheta(\text{Store}) \supset \{ \langle \text{Store}, \text{Manager} \rangle, \langle \text{Region}, \text{Country} \rangle \}.$$

In addition, other viewpoints, complementary to the previous ones, could be considered. For instance, [17] argues for viewpoints that support visual-seeking tasks, such as *zoom*, *filter*, or *overview*, among others. However, we leave them out of the scope since we consider them to be more related to interaction rather than to information visualization.

2.2 OLAP Graph Layout Taxonomy

Conceptual models, decomposed as aforementioned, can be (abstractly) characterized as graph layouts. Fig. 2 shows the main graph layouts both identified in the literature and proposed herein. Alternative layouts from the graph visualization literature have been selected in order to compare their readability with those provided by the original research proposals. Layouts are described by means of their vertex & edge properties in such a manner as to enable their analysis in the context of the conceptual data warehouse modeling.

The layouts identified in the literature are listed next:

Mesh arranges vertices for convenience. Edges are drawn as lines between vertices.

Radial layout arranges vertices around a pivot. Edges are drawn as straight lines between normal vertices and this pivot.

Stack arranges equidistance vertices along a certain axis. Edges are omitted since all vertices are related to a unique pivot that is omitted in the viewpoint.

Tree arranges vertices along some direction according to some transitive relationship. Edges are depicted as straight lines between each pair or related vertices.

Tree-table arranges vertices along an axis and by some increment on another axis orthogonal to the former one, according to some transitive relationship. Edges are drawn as polylines with a 90° angle.

The alternative layouts for comparison purposes are the following ones:

Arc Diagram arranges equidistance vertices along an axis. Edges are drawn as arcs between the vertices in such a way that their slope is preserved.

Cross-tab arranges equidistance vertices along two orthogonal axes. Edges are drawn as points in such space.

Paragraph (omitted in Fig. 2 for convenience) arranges vertices as words in a paragraph. Edges are omitted since all vertices are related to a unique pivot that is omitted in the viewpoint.

Parallel Coordinates arrange vertices along parallel axes. Edges are drawn as straight lines between vertices from one axis to the other.

2.3 Readability Measures

The readability associated with the graph layouts is evaluated according to the measures presented below. First, we select the following four aesthetics criteria (expressed as boolean values) related to the well-known Gestalt laws of perception [19], which are empirically validated principles about how humans perceive visual stimuli.

Continuity (abbreviated as C in Table 1). “Humans tend to assign objects to an entity that is defined by smooth lines or curves”. It qualifies OLAP layouts whose modeling elements can be transitively related by following a continuous path with the eyes.

Orthogonality (O). Despite of not being represented by its own Gestalt law, it is a usual graph aesthetics criterion [12]. It qualifies OLAP layouts whose modeling elements are located according to the facts of an imaginary grid.

Proximity (P). “Humans tend to group nearby objects”. It qualifies OLAP layouts whose modeling elements of the same kind are located near and are thus visually identified as a unique group.

Symmetry (S). “Humans tend to perceive objects as symmetrical shapes that form around their centre”. It qualifies OLAP layouts whose modeling elements are arranged symmetrically, whether such symmetry be radial or mirror, etc.

We also select the following convenient indicators, which are motivated by the related work [3, 12, 18]:

Density (D) is an indicator of the number of modeling elements by diagram space unit. It is evaluated separately in both axes, the horizontal and the vertical one. It is measured by a 4-value ordinal scale interpreted as ‘very low’, ‘low’, ‘medium’, and ‘high’.

Edge Overlap (EO) is an indicator of the number of edge crossings. It is measured by a 4-value ordinal scale interpreted as ‘null’, ‘low’, ‘medium’, and ‘high’.

Fast Traversal (FT) is a boolean indicator of the velocity to traverse all the modeling elements of the OLAP graph layout. It is evaluated along any readable path within the graph.

In addition, an aggregated indicator of readability is calculated (see Table 1) by summarizing the previously introduced properties and indicators as follows. Boolean measures are coded in $\{0, 1\}$ as usual, density (in both x, y) is coded in $\{-2, -1, 0, +1\}$ (-2 for ‘very low’, $+1$ for ‘high’), and edge overlap is coded in $\{-3, -2, -1, 0\}$ (-3 for ‘high’, 0 for ‘null’). In addition, fast traversal is codified as 0.5, if it only evaluates some component of the graph layout.

Table 1: Readability evaluation of the graph-layout in OLAP diagrams

Layout	C	O	P	S	D	EO	FT	Total
Mesh					...xy	...		-1
Radial			✓	✓	...xy		✓	3
Stack		✓	✓		...xy		✓	5
Star			✓		xy		✓ _o	-0.5
Tree			✓		...xy	.	✓ _x	0.5
Tree-table		✓	✓		...xy	.	✓ _x	1.5
Arc Diag.	✓				...x'y	.	✓	2
Cross-tab		✓	✓		...xy		✓	5
Parallel Co.			✓		...xy	...	✓	2
Paragraph	✓				...xy		✓	4

3. FINDINGS

This section reports our findings. For each viewpoint defined, the layouts identified and those defined for comparison purposes were evaluated against the readability measures defined above (§3.1). Moreover, this evaluation is summarized and related to the corresponding OLAP viewpoints in order to identify the most readable graph layouts (§3.2).

3.1 Readability Evaluation

Table 1 shows the readability evaluation based on the graph layout characterization by viewpoint². In order to produce it, some assumptions had to be made for each layout (listed in Appendix A).

As shown in Table 1, density was estimated separately for the horizontal and the vertical axis. For instance, arc diagram layout has a medium horizontal (x) density and a high vertical (y) density. The estimation of fast traversal was done in some cases by focusing on specific components. For instance, fast traversal of the star layout was evaluated in its radial components (✓_o). Moreover, the column ‘Total’ presents the aggregated readability value.

Some additional remarks have to be made here, what concerns the indicators based on the Gestalt laws. For the star layout, proximity and symmetry evaluate the component radials. For the paragraph layout, continuity does not evaluate line breaks. Edge overlap is not applicable to radial, stack, and paragraph layouts since no edges are needed. It is not null for tree and tree-table, since the relations may be arranged as lattices (is the case of $L \times L^n \vartheta(d)$). Concerning fast traversal, it is high for radial circumferences and low for radial centers of stars. It is high along the same horizontal axis and low through axes of tree and tree-tables.

3.2 Readability by Viewpoint

In order to identify the most readable graph layouts by viewpoint, the former were ranked by viewpoint class (§2.1). For this task, the applicability of graph layouts to each viewpoint was also taken into account. Both the relationship between graph layouts and viewpoints and that between the graph layout readability values are shown in Table 2.

In Table 2, the most readable graph layouts by viewpoint class are shown with shading. Indeed, only tree layouts (one by viewpoint class) are necessary to cover all viewpoints according to our measurement framework. In particular, only $L \times L^n \vartheta(d)$ stacks were already presented in the surveyed research proposals (but not as a separated viewpoint). Inter-

²Evaluation of composite layouts of $L \times A \vartheta(d)$ can be derived from their components.

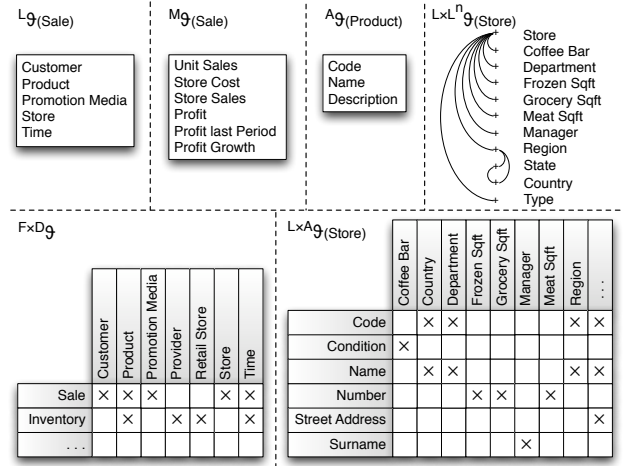


Figure 3: Viewpoints with optimal (graph-based) diagram readability of the motivating example

estingly, cross-tab layout is an outlier (ranked with 5, followed by layouts ranked with 2) for relational layouts that were overlooked by the surveyed research proposals. For primitive viewpoints, arc diagrams were ranked over the surveyed layouts, but with less difference than the former class (2, followed by 1.5 for tree table).

3.3 Conclusion

The practical result of this study is the proposal of better alternative layouts, founded in advanced visualization techniques, than the presented in current proposals for the conceptual modeling of data warehouses. The proposed readability measurement framework points out two alternative layouts, namely, arc diagrams (applicable to OLAP aggregation hierarchies) and cross-tabs (applicable to level-attribute & fact-dimension relationships), as better candidates than the ones proposed in the literature.

Resuming our motivating example, Fig. 3 shows (with a convenient notation) some sample diagram counterparts of the whole diagram from Fig. 1, which represent the viewpoints for the most readable graph layouts identified herein. In this way, the modeling tools that implement OLAP diagramming by means of these viewpoints and graph layouts can empower the management of conceptual models and thus reduce the cost of a data warehousing project. Indeed, one of our ongoing works consists in building a prototype tool that supports our findings and offers designers usable conceptual models in production environments.

Conceptual modeling for data warehouses has implicitly assumed the benefits of viewpoints. However, they were not formally treated up to now. We hope that this study will inspire further research along this line, such as the proposal of cognitive theories and measurement frameworks or the application of similar materials and methods like ours.

4. REFERENCES

- [1] A. Abelló, J. Samos, and F. Saltor. YAM²: a multidimensional conceptual model extending UML. *Inf. Syst.*, 31(6):541–567, 2006.

Table 2: Summary of layouts by viewpoint

Source	Layout	Eval.	$A_{\vartheta}(l)$	$L_{\vartheta}(f)$	$M_{\vartheta}(f)$	$L \times A_{\vartheta}(d)$	$F \times D_{\vartheta}$	$L \times L^n_{\vartheta}(d)$
Surveyed	Mesh	-1		✓	✓	✓		✓
	Radial	3	✓	✓	✓	✓	✓	
	Stack	5	✓	✓	✓	✓		
	Star	-0.5					✓	
	Tree	0.5				✓		✓
	Tree table	1.5	✓	✓			✓	✓
Altern.	Arc Diagram	2				✓	✓	✓
	Cross-tab	5				✓	✓	
	Parallel Coord.	2				✓	✓	
	Paragraph	4	✓	✓	✓			

- [2] T. Baar. Correctly defined concrete syntax. *Software and Systems Modeling*, 7(4):383–398, 2008.
- [3] M. Eiglsperger, M. Kaufmann, and M. Siebenhaller. A Topology-Shape-Metrics Approach for the Automatic Layout of UML Class Diagram. In *SOFTVIS*, pages 189–198, 2003.
- [4] A. Finkelstein and I. Sommerville. The viewpoints FAQ. *BCS/IEEE Software Eng. J.*, 11(1):2–4, 1996.
- [5] T. R. G. Green and M. Petre. Usability Analysis of Visual Programming Environments: A ‘Cognitive Dimensions’ Framework. *J. Vis. Lang. Comput.*, 7(2):131–174, 1996.
- [6] D. Harel and B. Rumpe. Meaningful Modeling: What’s the Semantics of “Semantics”? *IEEE Computer*, 37(10):64–72, 2004.
- [7] R. Kimball and M. Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. Wiley, 2002.
- [8] S. Luján-Mora, J. Trujillo, and I.-Y. Song. Multidimensional Modeling with UML Package Diagrams. In *ER*, pages 199–213, 2002.
- [9] S. Luján-Mora, J. Trujillo, and I.-Y. Song. A UML profile for multidimensional modeling in data warehouses. *Data Knowl. Eng.*, 59(3):725–769, 2006.
- [10] S. Mansmann, F. Mansmann, M. Scholl, and D. Keim. Hierarchy-driven Exploration of Multidimensional Data Cubes. In *BTW*, pages 96–111, 2007.
- [11] C. Plaisant. The Challenge of Information Visualization Evaluation. In *AVI*, pages 109–116, 2004.
- [12] H. Purchase, J. Allder, and D. Carrington. Graph layout aesthetics in UML diagrams: user preferences. *J. Graph Algorithm. Appl.*, 6(3):255–279, 2002.
- [13] S. Rizzi, A. Abelló, J. Lechtenböcker, and J. Trujillo. Research in data warehouse modeling and design: dead or alive? In *DOLAP*, pages 3–10, 2006.
- [14] O. Romero and A. Abelló. A Survey of Multidimensional Modeling Methodologies. *Int. J. Data Warehousing Min.*, 5(2):1–23, 2009.
- [15] E. Seidewitz. What Models Mean. *IEEE Software*, 20(5):26–32, 2003.
- [16] M. A. Serrano, C. Calero, H. A. Sahraoui, and M. Piattini. Empirical studies to assess the understandability of data warehouse schemas using structural metrics. *Software Qual. J.*, 16(1):79–106, 2008.
- [17] B. Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *VL*, pages 336–343, 1996.
- [18] E. R. Tufte. The Visual Display of Quantitative Information. *American Journal of Physics*, 53:1117, 1985.
- [19] M. Wertheimer. Laws of Organization in Perceptual Forms. *A Source Book of Gestalt Psychology*, pages 71–88, 1938.

APPENDIX

A. GRAPH LAYOUT ASSUMPTIONS

The assumptions made for evaluating the surveyed layouts are listed next:

Mesh. Vertices are uniformly distributed.

Radial. Vertices are uniformly distributed along the radial circumference and are alphabetically ordered.

Stack. Vertices are alphabetically ordered and vertically aligned.

Star. Given a component radial, vertices are uniformly distributed along the radial circumference and alphabetically ordered.

Tree, Tree-table. Vertices at the same distance from the root are alphabetically ordered, horizontally aligned, and spaced proportionally to their leaf cardinal. The root vertex is on the upper left-hand side corner.

In what follows, the assumptions of the alternative layouts are presented:

Arc Diagram. Vertices are uniformly distributed, vertically aligned, and ordered alphabetically & by modeling element (concrete notation discriminates them somehow). Directed relationships (*i.e.*, roll-ups) are read downwards. The total arc diameter is the lowest possible (given the previous constraint), and arc height is constrained.

Cross-tab. Each axis contain vertices of the same kind. Crosses model the corresponding relation. Axes are alphabetically ordered.

Paragraph. Vertices are alphabetically ordered. For viewpoints with a given parameter, *e.g.*, $A_{\vartheta}(l)$, it is inferred from the context.

Parallel Coordinates. Each axis contain vertices of the same kind. Axes are alphabetically ordered and vertically aligned.