The DNA Framework of Visualization

Yuri Engelhardt[1]([✉]) [iD] and Clive Richards[2] [iD]

[1] University of Twente, Enschede, The Netherlands
`yuri.engelhardt@utwente.nl`
[2] Birmingham City University, Birmingham, UK
`clive.j.richards@me.com`

Abstract. A comprehensive framework is presented for analyzing and specifying an extensive range of visualizations in terms of their fundamental 'DNA' building blocks of visual encoding and (de)composition.

1 Introduction

In their "Tour through the Visualization Zoo", Heer et al. (2010) say that "all visualizations share a common 'DNA' – a set of mappings between data properties and visual attributes" (p. 60). We use this metaphorical idea of the 'DNA of visualization' in a similar vein, taking it to the extent of identifying a comprehensive set of individual DNA building blocks of visualizations and the rules for combining them. This allows for the construction of a broad range of different types of visualizations.

Numerous authors have written about analysing visualizations and various visualization grammars have been developed (e.g. *Vega-Lite* www.vega.github.io). We have reviewed this work and we have identified gaps in what is covered (see Engelhardt and Richards 2018). The framework we present here fills these gaps. It:

1. provides a comprehensive system for exploring and checking design possibilities for visualization.
2. offers a system of tree diagrams for representing (de)composition and visual encoding in visualizations (constructed from their 'DNA').
3. presents a way of describing visualizations with rigorously systematic natural language sentences, which specify (de)composition and visual encoding.
4. covers a very broad design space of visualization, not only including visual representations that involve numerical information, but also visualizations such as family trees, Venn diagrams, flow charts, texts using indenting, technical drawings and scientific illustrations.

The above characteristics of the framework enable the analysis and comparison of visualization types, and potentially provide a design method for exploring visualization options. Like academic work in linguistics, the work presented here is primarily not prescriptive but descriptive, in the sense that it enables the understanding and modelling of (graphic) language.

Y. Engelhardt and C. Richards—both authors contributed equally to the work

2 The Building Blocks and How They Relate to Each Other

The process diagram in Fig. 1 shows all our DNA building blocks and their possible relationships for expressing information visually. For ease of use we have given each DNA building block a three letter abbreviation. The DNA building blocks fall into several main groups: *types of information* to be represented, *visual encodings* to represent them, *visual components* that make up the visualization, and any *directions* or *layout principles* that may be involved. *Visual encodings* can be used for *arranging*, *varying* or *linking* visual components. *Arranging* visual components into meaningful configurations is how visualizations are *constructed*.

Our *visual encodings* include the use of Bertin's 'visual variables', some Gestalt principles of perception (e.g. grouping by proximity) and other fundamental ways of expressing information visually. A visual component can be involved in several different visual encodings, simultaneously representing different types of information.

We refer to a 'well-formed' combination of DNA building blocks as a *visualization pattern*. Many common *visualization patterns* have been given a name (e.g. 'pie chart') and are generally referred to as 'chart types', while novel or rare patterns often do not have a name (yet). A visualization pattern can be transformed into another pattern by adding, replacing or removing one or more DNA building blocks. A large number of patterns has been analyzed using this system. Some examples can be found in Fig. 2. Many more analyses are on our accompanying website: www.VisDNA.com.

3 Discussion and Conclusions

The framework offers a potential research tool for exploring various kinds of commonalities, family resemblances and differences between visualization patterns within collections of graphic representations. The DNA building blocks and the precisely defined methods by which they can be combined (see www.VisDNA.com) offer the potential for machine readable specifications. This may serve as a basis for a system providing computer generated visualization advice, which could be linked to a rendering engine in order to produce actual visualizations and variants of them.

Because of its flexible building block structure, additional DNA elements may be added to the framework to accommodate any new constructions that one may want to describe and that cannot be fully analysed using the current scheme. An example may be the addition of DNA building blocks for interactivity in visualizations.
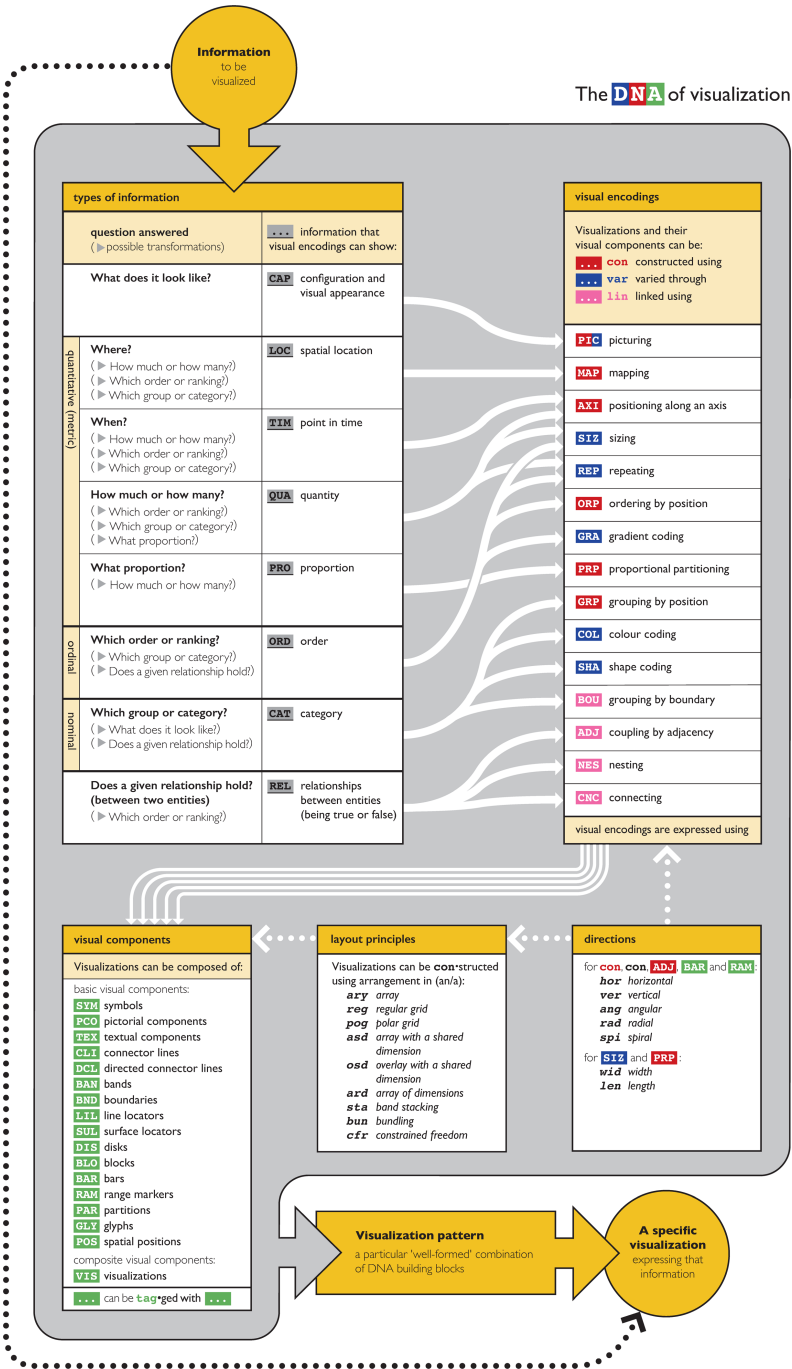
Fig. 1. Process diagram showing our DNA building blocks and their possible relationships.

Pattern & specimen	Information*	DNA	Specification by natural language	DNA tree diagram

1 Technical illustration

CAP ▶ / ... ▶

VIS — con — PIC — PCO — tag — TEX

A technical illustration is constructed using **pictur**ing of pictorial components that are tagged with textual components.

2 Comic strip

TIM ▶ ORD ▶ / CAP ▶

VIS — con — hor — ORP — VIS — con — PIC — PCO

A comic strip is constructed using **ordering by** *horizontal* **position** of visualizations that are constructed using **pictur**ing of pictorial components.

3 Hans Rosling bubble chart

QUA ▶ / QUA ▶ / QUA ▶ / CAT ▶

VIS — con — hor — AXI — ver — AXI — SYM — var — SIZ — COL

A Hans Rosling bubble chart is constructed using **positioning along a** *horizontal* **axis** and **positioning along a** *vertical* **axis** of symbols that vary through **sizing** and **colour coding**

4 Pie chart map

LOC ▶ / QUA ▶ / PRO ▶ / QUA ▶ / CAT ▶

VIS — con — MAP — DIS — var — SIZ — con — ang — PRP — PAR — var — SIZ — COL

A pie chart map is constructed using **mapping** of disks that vary through **sizing** and that are constructed using **proportional** *angular* **partitioning** into partitions that vary through **sizing** and **colour coding**.

5 Encoding tree diagram that describes itself

ORD ▶ / CAT ▶ / CAT ▶ / REL ▶ / REL ▶

VIS — con — ver — ORP — TEX — var — COL — lin — ADJ — CNC — CLI — DCL

This encoding tree diagram that describes itself is constructed using **ordering by** *vertical* **position** of textual components that vary through **colour coding** and that are linked using **coupling by** *horizontal* **adjacency** and **connecting** by connector lines and directed connector lines.

---

* **TIM** *Point in time* **LOC** *Location* **QUA** *Quantity* **PRO** *Proportion* **ORD** *Order* **CAT** *Category* **CAP** *Configuration and appearance* **REL** *Relationships*

... Placeholder for any *type of information*   Transformations between these types of information are shown as grey arrows.
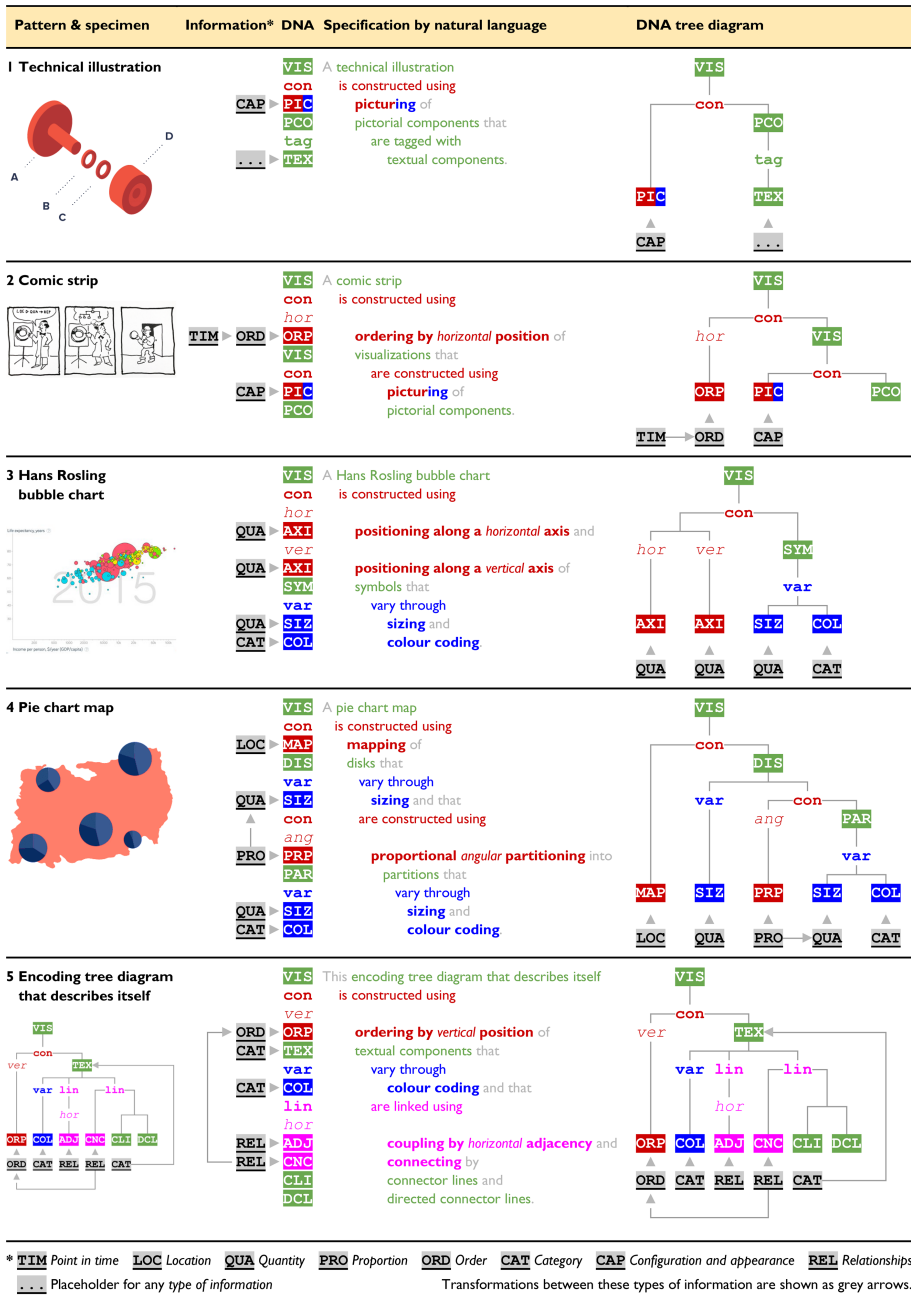

Fig. 2. Example DNA analyses. See www.VisDNA.com for analyses of many more types of visualizations. *Images at 1 and 4 courtesy of the DataVizProject by Ferdio, Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License:* www.datavizproject.com

References

Engelhardt, Y., Richards, C.: A framework for analyzing and designing diagrams and graphics. In: Chapman, P., Stapleton, G., Moktefi, A., Perez-Kriz, S., Bellucci, F. (eds.) Diagrams 2018. LNCS (LNAI), vol. 10871, pp. 201–209. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91376-6_20

Heer, J., Bostock, M., Ogievetsky, V.: A tour through the visualization zoo. Commun. ACM **53**(6), 59–67 (2010). https://doi.org/10.1145/1743546.1743567