Graph Transformations:
Where They Come From and Where They Should Go

Hans J. Schneider

Lehrstuhl für Programmiersprachen, Universität Erlangen-Nürnberg,
Martensstraße 3, D-91058 Erlangen
Electronic Mail: `schneider@informatik.uni-erlangen.de`

In 1971, Händler and Weizenbaum organized a NATO Advanced Study Institute on *Display Use for Man-Machine Dialog.* This was a research topic at that time and not yet an every-day technique. One of the expectations expressed in the preface of the conference proceedings was that teleprinter-like terminals are not a satisfactory medium to communicate with a computer. Therefore, the organizers had asked some authors for presenting formal language approaches to describe input/output using displays. There were some approaches that did not extend formal language theory: One approach used context-free expressions the operation symbols of which have been given a geometrical meaning. Another approach used attributes to identify geometrical juxtaposition. In my contribution, I presented a new formalism generalizing the string grammar approach. The poor displays of that time allowed only horizontal and vertical concatenation as well as enclosing subdiagrams. From this, I concluded that we had to replace strings by symbol arrangements, partially ordered with respect to some relations, and I proposed a formalism to transform partial orders by rewriting steps. Of course, these arrangements can be regarded as labelled graphs.

Today, we have many different formal models to describe graph transformations. They are not only much more general than we expected in the very beginning, but also there is a wide variety of application fields that may benefit from these formalisms. We are able to formally describe the syntax (and often also semantical aspects) of all kinds of structures. I mention only a few, but very different fields: term-graph rewriting as a model of computability, actor-graph grammars as a model of distributed systems, and compiler generators for visual applications. Succeeding in such different fields was a benefit we have got from the diversity of our models since each model is tailored to the specialities of a family of applications. But this diversity is a disadvantage as well: New-comers may be disturbed and annoyed about this pluralism.

My desideratum for the future is to identify a (relatively small) number of basic techniques and, especially, notations that allow to describe the different approaches in a common framework, clarifying the differences between them and emphasizing the advantages of the special models. On this common basis, which must not replace pluralism by fundamentalism, the diversity of research in graph transformation systems may flourish.