CP-VIZ: An Open Source Visualization Platform for CP

Helmut Simonis and Paul Davern*

Cork Constraint Computation Centre
Department of Computer Science, University College Cork, Ireland
contact: h.simonis@4c.ucc.ie

Visualization is one of the best techniques for understanding the behaviour of constraint programs, allowing us to directly observe the impact of changes by visual inspection instead of using tedious debugging. So far, most constraint visualization tools have been closely linked to specific solvers, making it difficult to compare alternative solvers and to reuse development effort spent on other systems. Previous attempts [1] at generic tools were unsuccessful largely due to the complexity of the specification. The new, light-weight CP-VIZ system provides a simple XML based interface for solvers, and can be easily extended for new systems and constraints. In CP-VIZ, we try to visualize the search tree and the state of variables and (global) constraints in parallel views. The search tree shows choices, assignments and failures, modelled on the tree display in the OZ Explorer [3] and later in CHIP [4]. Constraints and variables are shown in a 2D layout defined by the user, individual global constraints are shown in custom visualizations similar to [5]. A new constraint can be added to the package by deriving a new class with a custom drawing method. The design of the visualization tool was driven by user-requirements, coming mainly from the development of an ECLiPSe ELearning course.

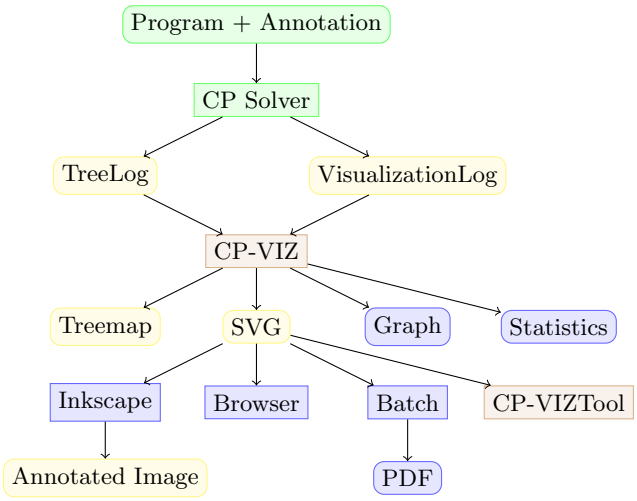
- We decided to concentrate on post-mortem analysis, which minimizes the requirements of interaction between the constraint solver and the visualization environment, but still provides most of the information required for analysis.
- The output of the visualization can be studied on-screen, but can also be provided as high-quality, vector based print output. Data feeds for other visualization tools are also provided.
- The tools are solver independent, written in a general purpose language (Java) and can be easily extended and specialized.
- We added invariant checking at each search node to the functionality, this allows a solver independent validation of the results, and can highlight missing propagation in individual constraints.
- The system is platform independent, and is provided as open source. The system specific XML generation requires minimal effort.

Figure 1 shows the basic architecture of the CP-VIZ system. The visualization is driven by annotations in the constraint program. When run in the solver, two

* This work was supported by Science Foundation Ireland (Grant Number 05/IN/I886). The support of Cisco Systems and of the Silicon Valley Community Foundation is gratefully acknowledged.

XML log files (one for the search tree, the other for the constraint and variable visualization) are produced. These files are then parsed in the main CP-VIZ application, producing graphical output as SVG, or as input for other tools (tree maps, graphs, statistics). The SVG output can be displayed interactively in the CP-VIZTool, or can be used in multiple ways to produce annotated or converted output for print or WEB media. Besides the initial interface to the

Fig. 1. Architecture



ECLiPSe constraint system, interfaces to SICStus Prolog, Choco and the JSR-331 reference implementations have been or are currently being developed.

References

1. Pierre Deransart. Main results of the OADymPPaC project. In Bart Demoen and Vladimir Lifschitz, editors, *ICLP*, volume 3132 of *Lecture Notes in Computer Science*, pages 456–457. Springer, 2004.
2. Pierre Deransart, Manuel V. Hermenegildo, and Jan Maluszynski, editors. *Analysis and Visualization Tools for Constraint Programming, Constraint Debugging (DiSCiPl project)*, volume 1870 of *Lecture Notes in Computer Science*. Springer, 2000.
3. Christian Schulte. Oz Explorer: A visual constraint programming tool. In *ICLP*, pages 286–300, 1997.
4. Helmut Simonis and Abderrahmane Aggoun. Search-tree visualisation. In Deransart et al. [2], pages 191–208.
5. Helmut Simonis, Abderrahmane Aggoun, Nicolas Beldiceanu, and Eric Bourreau. Complex constraint abstraction: Global constraint visualisation. In Deransart et al. [2], pages 299–317.