

ARVis: Visualizing Relations between Answer Sets

Thomas Ambroz Günther Charwat Andreas Jusits Johannes Peter Wallner Stefan Woltran

{gcharwat, wallner, woltran}@dbai.tuwien.ac.at

Institute of Information Systems, Vienna University of Technology

Motivation

Improve answer set programming (ASP) versatility:

- ▶ Tools for user support needed
- ▶ In particular tools with GUI

Several existing tools that support answer set visualization:

- ▶ ASPViz
- ▶ IDPDraw
- ▶ Kara (part of SeaLion)
- ▶ ASPIDE

 However, no system for visualization of *relations* between answer sets

Visualization of relations can be used for

- ▶ Expressing preference criterion
- ▶ Representation of results that build upon graph structure
- ▶ Debugging

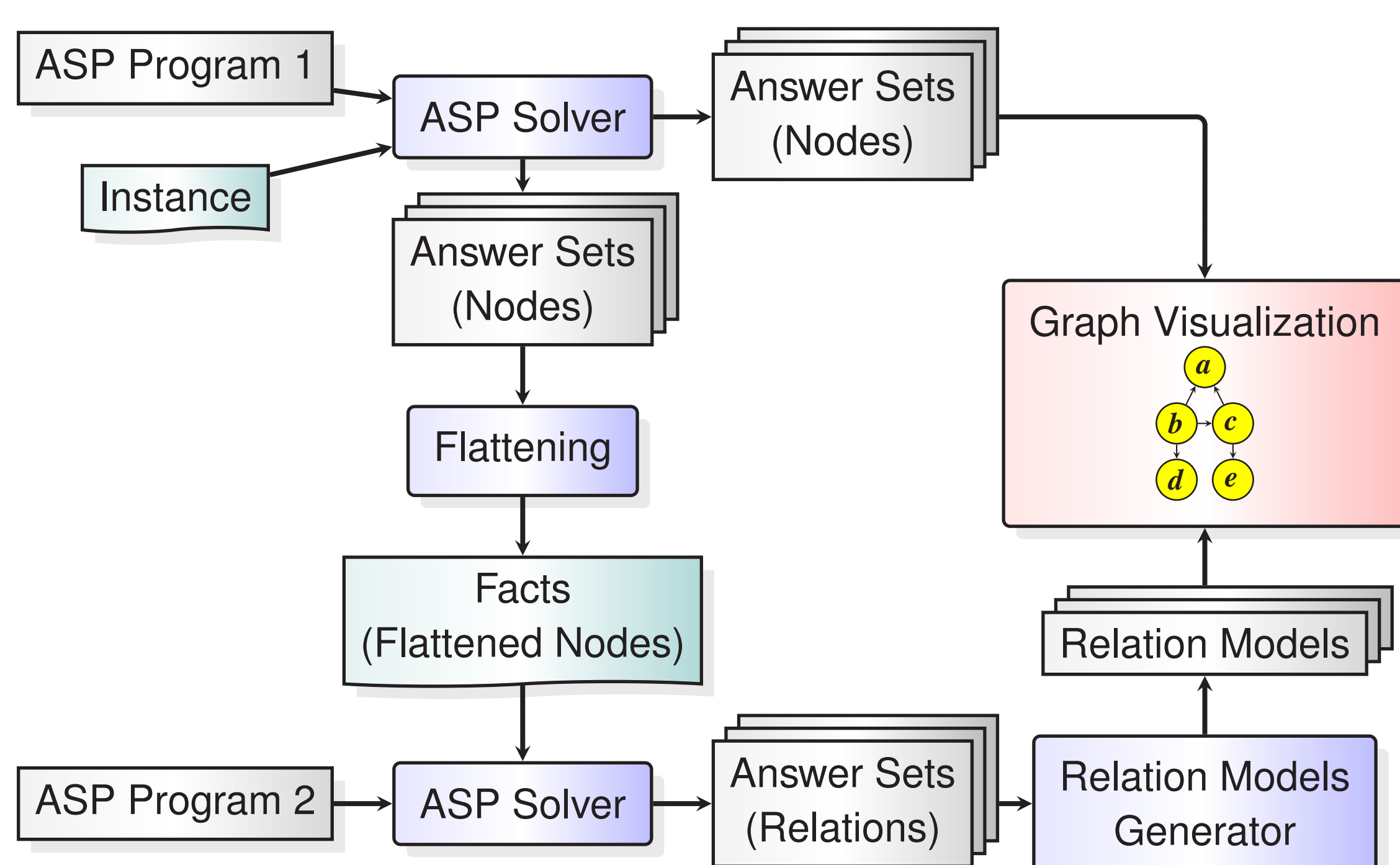
ARVis - “Answer Set Relationship Visualizer”


Key features:

- ▶ Problem and relations each expressed in an ASP encoding
- ▶ Results visualized as an interactive graph

System Concept

- ▶ Program 1, Program 2 and Instance are given as input



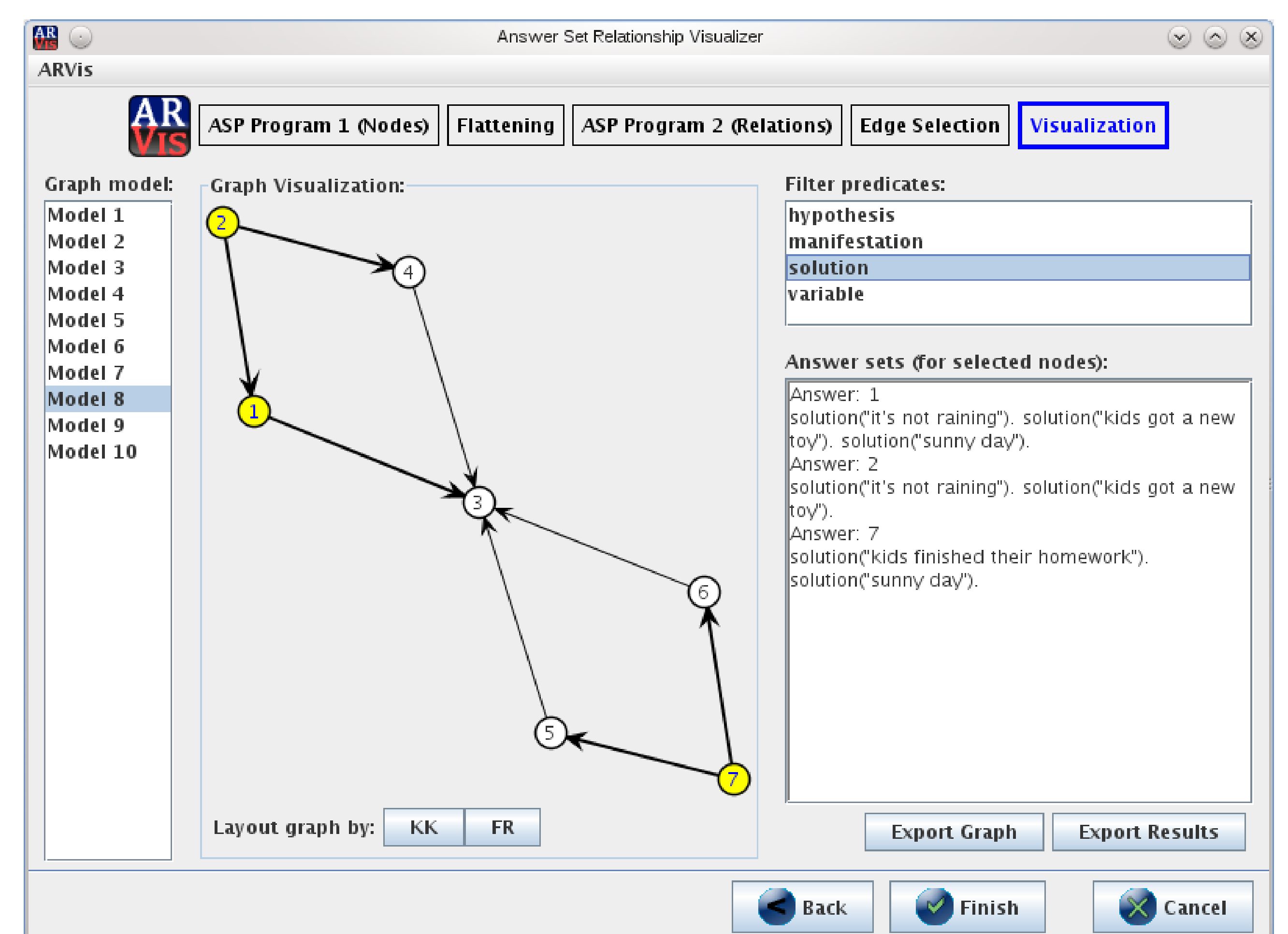
Case Study: Abduction

Abduction problem:

- ▶ Consists of background theory, manifestations and hypotheses
- ▶ Goal: Find subsets of hypotheses that explain manifestations
- ▶ Furthermore, relate these explanations w.r.t. their “quality” (e.g., cardinality, subset inclusion, ...)

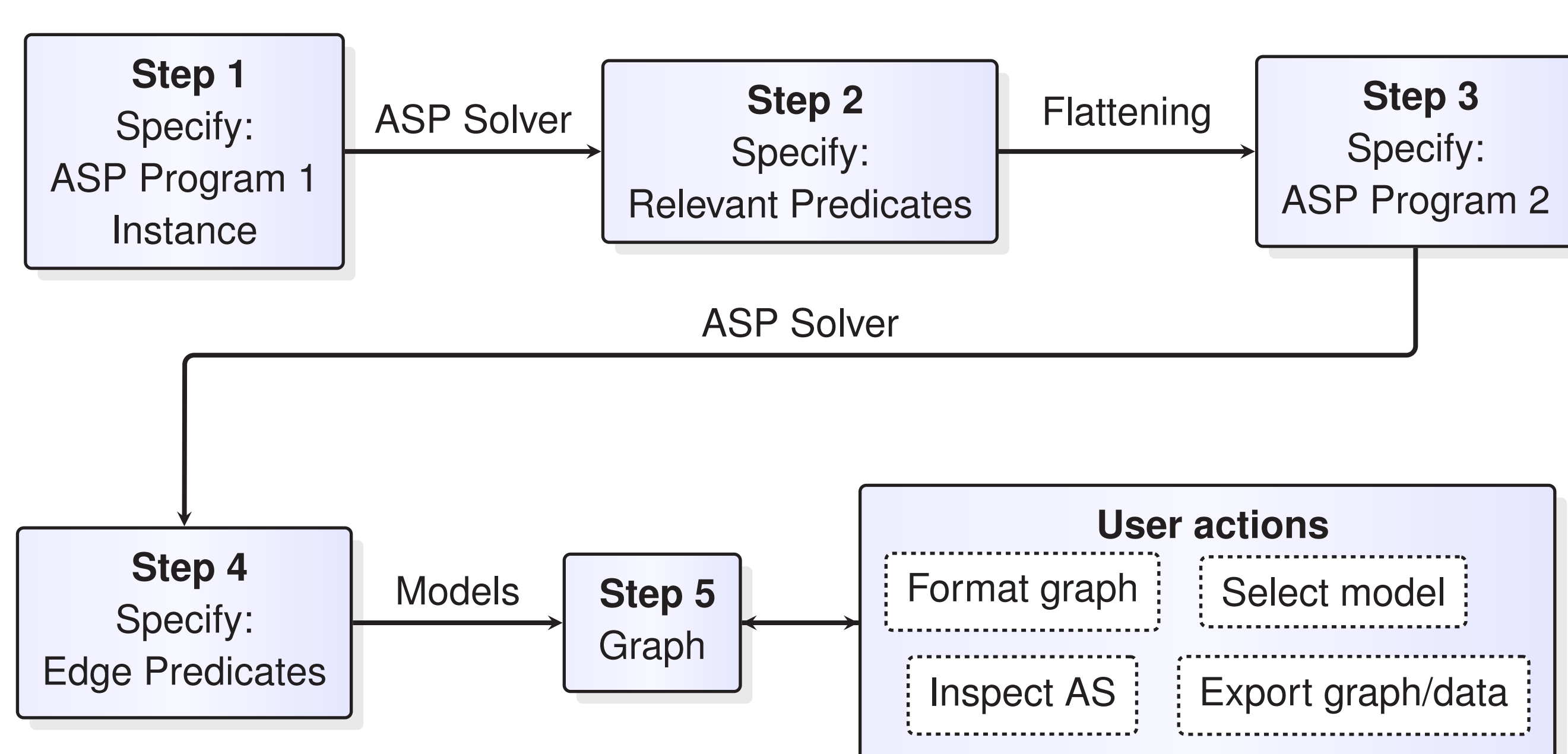
Implementation for ARVis:

- Program 1: Provides possible explanations for manifestations
- Program 2: Relates all explanations



Workflow

- ▶ Workflow, consisting of five steps, guides user through process



ARVis allows one to intuitively inspect obtained explanations:

- ▶ Nodes represent explanations
- ▶ Edges represent relation (here: subset inclusion)

Conclusion

System facts:

- ▶ Platform-independent – Implemented in Java, using Jung library
- ▶ Solver-independent – Support for Potassco family and DLV
- ▶ Domain-independent – Problems solely specified in ASP
- ▶ High performance – Handles 100s of nodes with 1000s of relations
- ▶ Versatile – Supports export and post-processing of graph and answer sets

Application areas:

- ▶ Any problem where preferences between solutions are relevant
- ▶ Problem domains with an underlying graph structure

Successfully applied:

- ▶ Abduction: Express preferences over explanations for observed behavior
- ▶ Argumentation: From a given knowledge base, obtain and visualize argumentation frameworks


<http://www.dbai.tuwien.ac.at/proj/arvis/>