Constraint Handling Rules for SWI-Prolog

Tom Schrijvers, Bart Demoen
K.U.Leuven

Jan Wielemaker
University of Amsterdam, The Netherlands

1. SWI-Prolog

- open source Prolog system

1. SWI-Prolog

- open source Prolog system
- large user base (> 5,000 downloads/month) teaching, research, industry

1. SWI-Prolog

- open source Prolog system
- large user base (> 5,000 downloads/month) teaching, research, industry
- focus on usability
  - tools: editor, debugger, ...
  - libraries: networking, semantics web, ...

1. SWI-Prolog

- open source Prolog system
- large user base (> 5,000 downloads/month) teaching, research, industry
- focus on usability
  - tools: editor, debugger, ...
  - libraries: networking, semantics web, ...
- missing feature: CLP support

1. SWI-Prolog

- open source Prolog system
- large user base (> 5,000 downloads/month) teaching, research, industry
- focus on usability
  - tools: editor, debugger, ...
  - libraries: networking, semantics web, ...
- missing feature: CLP support
  $\Rightarrow$ Constraint Hanling Rules

2. K.U.Leuven CHR System

- state-of-the-art CHR system

2. K.U.Leuven CHR System

- state-of-the-art CHR system
- based on reference implementation (C. Holzbaur)

2. K.U.Leuven CHR System

- state-of-the-art CHR system
- based on reference implementation (C. Holzbaur)
- written in hProlog (B. Demoen)

2. K.U.Leuven CHR System

- state-of-the-art CHR system
- based on reference implementation (C. Holzbaur)
- written in hProlog (B. Demoen)
- ported to XSB for integration with tabling

3. Built-ins for CHR

- attributed variables
  for propagation (co-routining)

3. Built-ins for CHR

- attributed variables
  for propagation (co-routining)
- global variables
  for constraint stores
  - backtrackable
  - preserving variable identity

3. Built-ins for CHR

- attributed variables
  for propagation (co-routining)
- global variables
  for constraint stores
  - backtrackable
  - preserving variable identity
- `setarg/3`, `term_variables/2`

3. Built-ins for CHR

- attributed variables
  for propagation (co-routining)
- global variables
  for constraint stores
  - backtrackable
  - preserving variable identity
- `setarg/3`, `term_variables/2`
- (large terms as arrays in efficient constraint stores)

4. New Features

- integration with `term_expansion/2`

4. New Features

- integration with `term_expansion/2`
- integration in debugger
  hides underlying generated Prolog code

4. New Features

- integration with `term_expansion/2`
- integration in debugger
  hides underlying generated Prolog code
- hash tables as constraint stores

4. New Features

- integration with `term_expansion/2`
- integration in debugger
  hides underlying generated Prolog code
- hash tables as constraint stores
- mode declarations for CHR
  (and types coming)

5. Related Work

- co-routining
  - `dif/2`
  - `when/2`

5. Related Work

- co-routining
  - `dif/2`
  - `when/2`
- small FD library

5. Related Work

- co-routining
  - `dif/2`
  - `when/2`
- small FD library
- port of C. Holzbaur's clp($\Re$) (in progress)

6. Conclusion

- Port of CHR system easy

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required
- Allows other co-routining and constraint facilities

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required
- Allows other co-routining and constraint facilities

Future Work:

- More efficient CHR system

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required
- Allows other co-routining and constraint facilities

Future Work:

- More efficient CHR system
- More CLP support for SWI-Prolog

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required
- Allows other co-routining and constraint facilities

Future Work:

- More efficient CHR system
- More CLP support for SWI-Prolog
- More host languages for CHR

6. Conclusion

- Port of CHR system easy
- Small set of built-ins required
- Allows other co-routining and constraint facilities

Future Work:

- More efficient CHR system
- More CLP support for SWI-Prolog
- More host languages for CHR

2 more talks about CHR in SWI-Prolog!