

A New Soar Debugger In Java

Douglas Pearson

douglas.pearson@threepenny.net



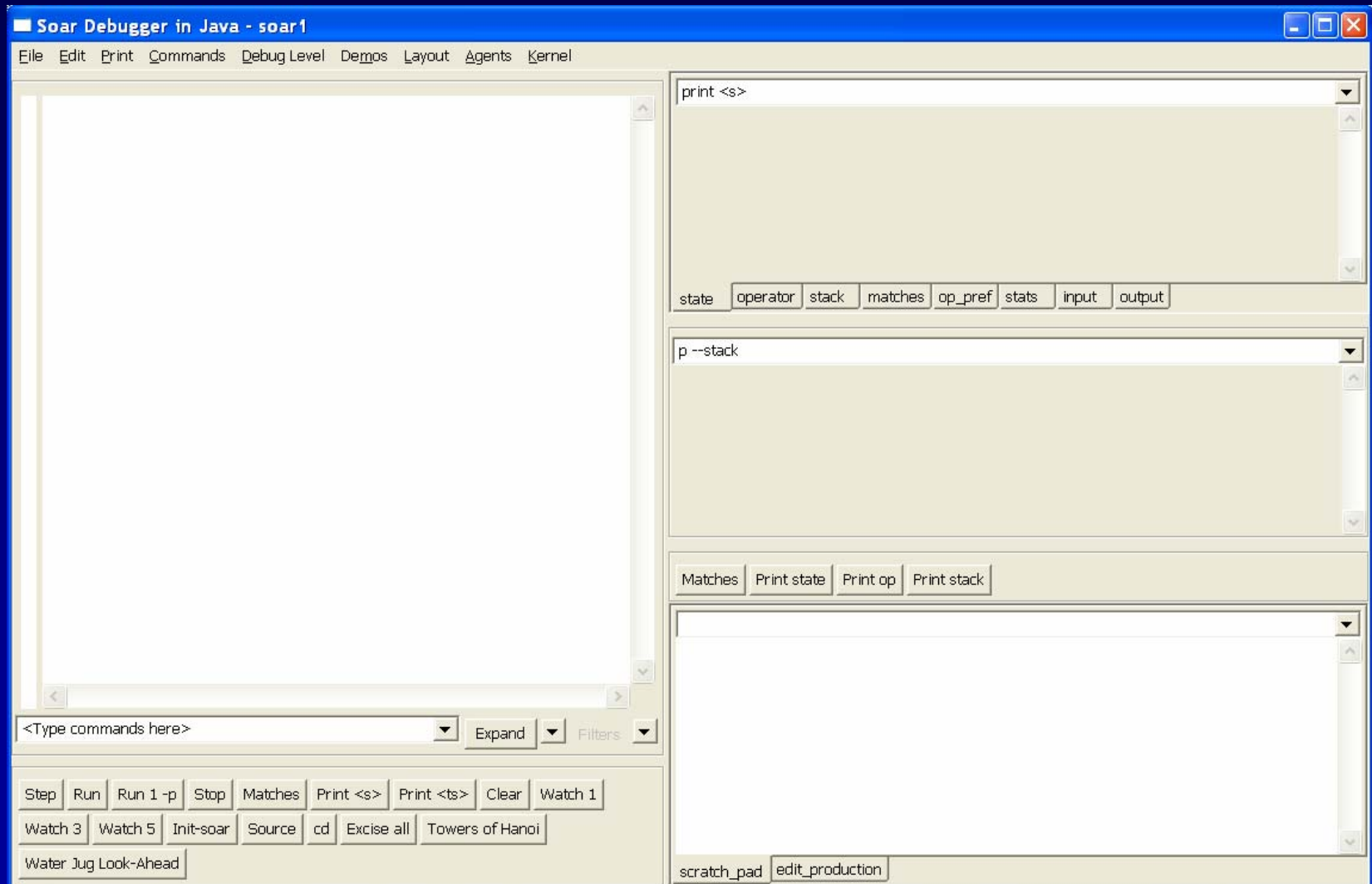
Why a New Debugger?

- Want to add a lot of new capabilities
- Extend TSI?
 - Based around string parsing – brittle
 - Kernel always embedded within debugger
 - gSKI inclusion required substantial rewrite anyway
 - Want plug-in and user-configurable architecture
 - Long-term Eclipse IDE goal
- So decision to create a new debugger in Java

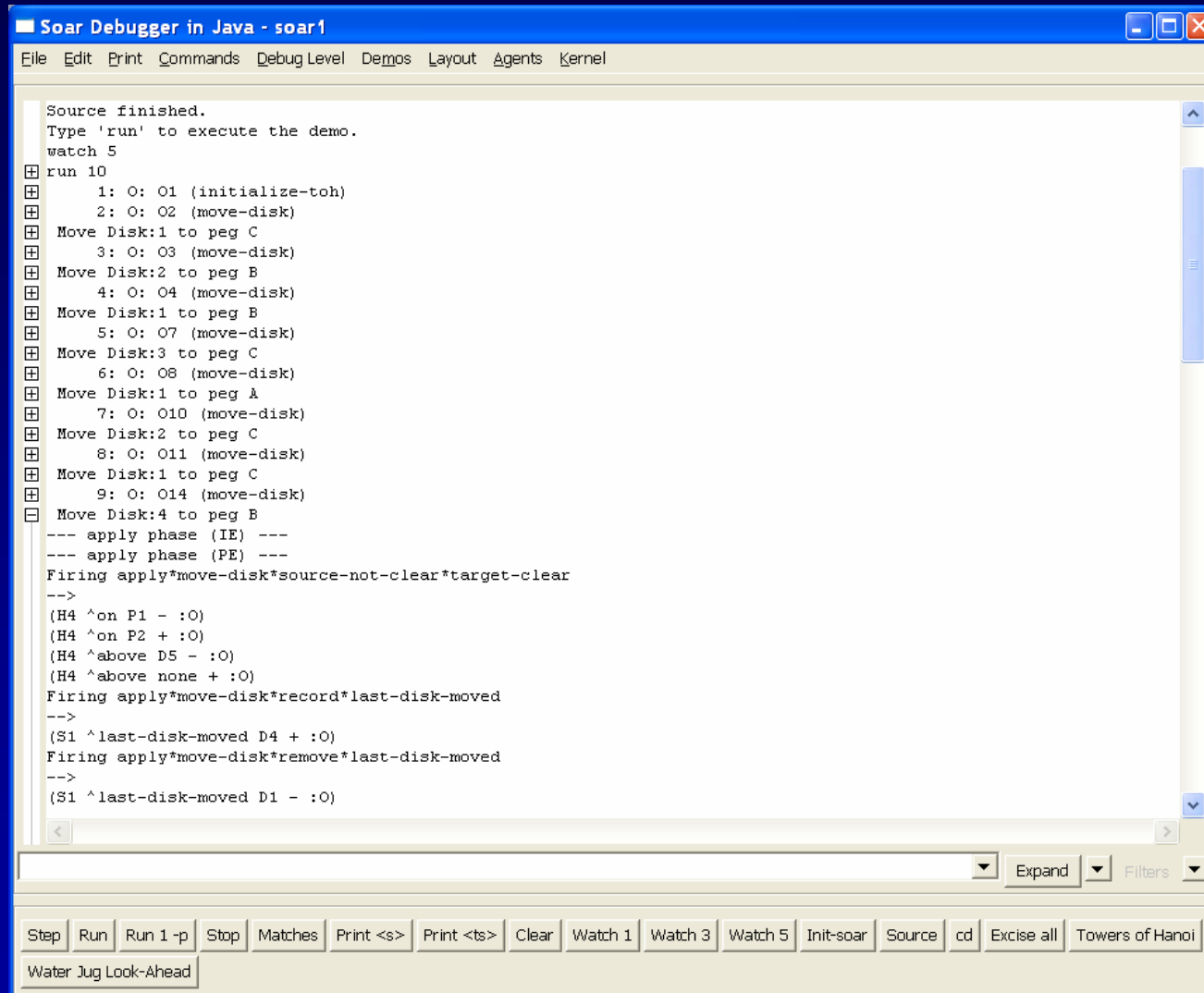
Some Initial Capabilities

- Structured (tree) trace
 - Watch more useful
- Filtered trace
 - Watch even more useful
- Automatic window updates at end of run
 - “watch windows” in other debuggers
- Visual Soar integration
 - Edit production / send file
- Dynamic connection to Soar kernels
 - Connect/disconnect to external processes

Quick Demo



Structured/Tree Trace



```
Soar Debugger in Java - soar1
File Edit Print Commands Debug Level Demos Layout Agents Kernel

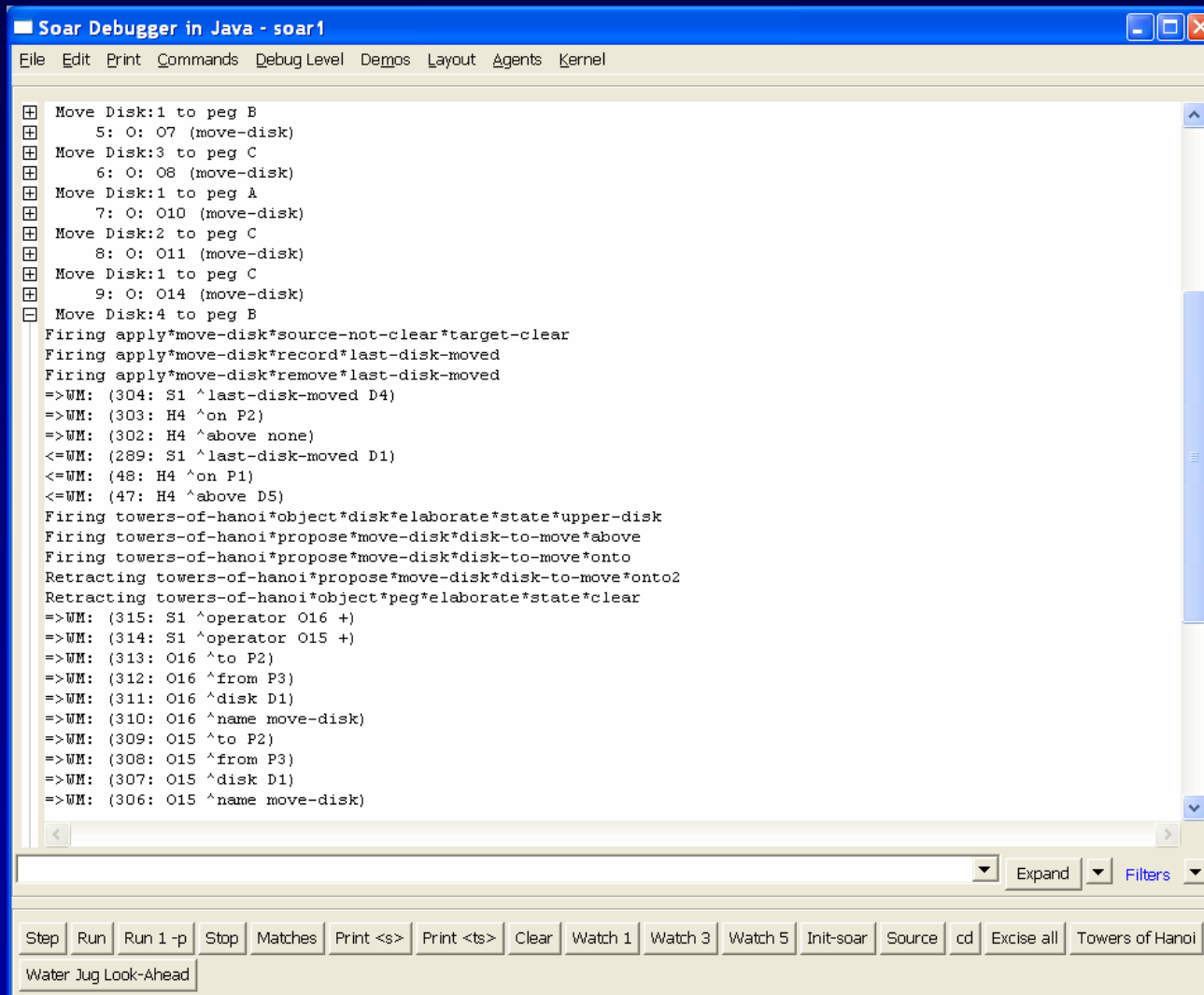
Source finished.
Type 'run' to execute the demo.
watch 5
run 10
  1: O: O1 (initialize-toh)
  2: O: O2 (move-disk)
  Move Disk:1 to peg C
  3: O: O3 (move-disk)
  Move Disk:2 to peg B
  4: O: O4 (move-disk)
  Move Disk:1 to peg B
  5: O: O7 (move-disk)
  Move Disk:3 to peg C
  6: O: O8 (move-disk)
  Move Disk:1 to peg A
  7: O: O10 (move-disk)
  Move Disk:2 to peg C
  8: O: O11 (move-disk)
  Move Disk:1 to peg C
  9: O: O14 (move-disk)
  Move Disk:4 to peg B
  --- apply phase (IE) ---
  --- apply phase (PE) ---
  Firing apply*move-disk*source-not-clear*target-clear
  -->
  (H4 ^on P1 - :O)
  (H4 ^on P2 + :O)
  (H4 ^above D5 - :O)
  (H4 ^above none + :O)
  Firing apply*move-disk*record*last-disk-moved
  -->
  (S1 ^last-disk-moved D4 + :O)
  Firing apply*move-disk*remove*last-disk-moved
  -->
  (S1 ^last-disk-moved D1 - :O)

Expand Filters
```

Step Run Run 1-p Stop Matches Print <s> Print <ts> Clear Watch 1 Watch 3 Watch 5 Init-soar Source cd Excise all Towers of Hanoi

Water Jug Look-Ahead

Filtering



The screenshot shows the Soar Debugger interface. The main window displays a list of events, including disk moves and firing rules. The filter bar at the bottom right contains the text "Expand" and "Filters".

```
File Edit Print Commands Debug Level Demos Layout Agents Kernel

+ Move Disk:1 to peg B
+   5: O: O7 (move-disk)
+ Move Disk:3 to peg C
+   6: O: O8 (move-disk)
+ Move Disk:1 to peg A
+   7: O: O10 (move-disk)
+ Move Disk:2 to peg C
+   8: O: O11 (move-disk)
+ Move Disk:1 to peg C
+   9: O: O14 (move-disk)
- Move Disk:4 to peg B
Firing apply*move-disk*source-not-clear*target-clear
Firing apply*move-disk*record*last-disk-moved
Firing apply*move-disk*remove*last-disk-moved
=>WM: (304: S1 ^last-disk-moved D4)
=>WM: (303: H4 ^on P2)
=>WM: (302: H4 ^above none)
<=WM: (289: S1 ^last-disk-moved D1)
<=WM: (48: H4 ^on P1)
<=WM: (47: H4 ^above D5)
Firing towers-of-hanoi*object*disk*elaborate*state*upper-disk
Firing towers-of-hanoi*propose*move-disk*disk-to-move*above
Firing towers-of-hanoi*propose*move-disk*disk-to-move*onto
Retracting towers-of-hanoi*propose*move-disk*disk-to-move*onto2
Retracting towers-of-hanoi*object*peg*elaborate*state*clear
=>WM: (315: S1 ^operator O16 +)
=>WM: (314: S1 ^operator O15 +)
=>WM: (313: O16 ^to P2)
=>WM: (312: O16 ^from P3)
=>WM: (311: O16 ^disk D1)
=>WM: (310: O16 ^name move-disk)
=>WM: (309: O15 ^to P2)
=>WM: (308: O15 ^from P3)
=>WM: (307: O15 ^disk D1)
=>WM: (306: O15 ^name move-disk)

Expand Filters
```

Step Run Run 1-p Stop Matches Print <s> Print <ts> Clear Watch 1 Watch 3 Watch 5 Init-soar Source cd Excise all Towers of Hanoi

Water Jug Look-Ahead

Visual Soar Integration

The image shows two overlapping windows from a software development environment. The left window, titled "Soar Debugger in Java - soar1", displays a list of commands and a context menu. The right window, titled "towers-of-hanoi", shows a code editor with a Soar script for the "move-disk" function.

Soar Debugger in Java - soar1

```
p
apply*move-disk*remove*last-disk1-peg
apply*move-disk*record*last-disk1-peg
apply*move-disk*remove*last-disk-moved
apply*move-disk*record*last-disk-moved
apply*move-disk*target-not-clear
apply*move-disk*source-not-clear*target-clear
apply*move-disk*source-clear*target-clear
towers-of-hanoi*propose*move-disk*disk-to-move*above2
towers-of-hanoi*propose*move-disk*disk-to-move*onto2
towers-of-hanoi*propose*move-disk*disk-to-move*onto2
print towers-of-hanoi*propose*move-disk*disk-to-move*onto2
edit towers-of-hanoi*propose*move-disk*disk-to-move*onto2
matches towers-of-hanoi*propose*move-disk*disk-to-move*onto2
matches towers-of-hanoi*propose*move-disk*disk-to-move*onto2 --wmes
excise towers-of-hanoi*propose*move-disk*disk-to-move*onto2
Window
towers-of-hanoi*object*disk*elaborate*state*larger
edit towers-of-hanoi*propose*move-disk*disk-to-move*onto2
edit towers-of-hanoi*monitor*operator-execution*move-disk
edit towers-of-hanoi*propose*move-disk*disk-to-move*onto2
```

towers-of-hanoi

```
File Edit Search Datamap View Soar Runtime Help
towers-of-hanoi
  _firstload
  all
  elaborations
  _readme
  initialize-toh
  move-disk

move-disk
File Edit Search Soar Insert Template Runtime
(<o> ^name move-disk
 ^disk <m-disk>
 ^from <source-peg>
 ^to <target-peg>)}

sp {towers-of-hanoi*propose*move-disk*disk-to-move*onto2
 "Target peg is clear."
 (state <s> ^name towers-of-hanoi
 ^upper-disk <m-disk>
 ^clear <target-peg>
 ^holds <h>
 ^last-disk-moved.name 1)
 (<h> ^disk <m-disk>
 ^on <source-peg>)
 (<m-disk> ^name <o> 1)
 -->
 (<s> ^operator <o>)
 (<o> ^name move-disk
 ^disk <m-disk>
 ^from <source-peg>
 ^to <target-peg>)}

sp {towers-of-hanoi*propose*move-disk*disk-to-move*above2
 "Upper disk on the target peg is larger."
 (state <s> ^name towers-of-hanoi)
```

Line: 43

Feedback

Design Principles

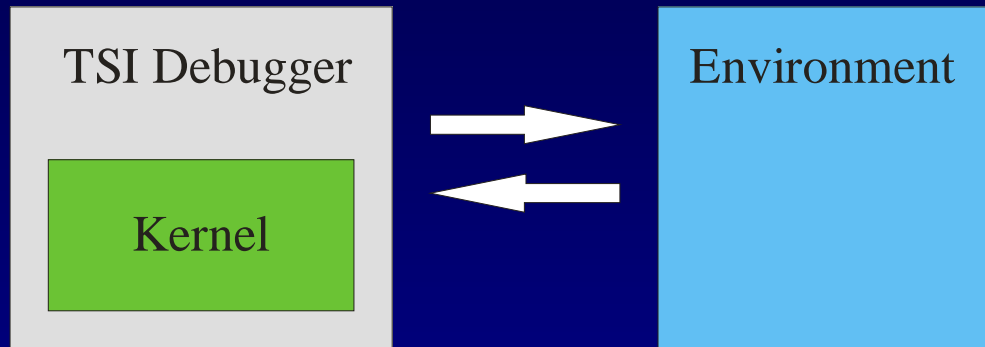
- XML based
 - Interface into Soar via XML (SML: Soar Markup Language)
No more string parsing, supports new capabilities
 - But maintain high performance
- Plug-in Architecture
 - Debugger made up of a series of modules
 - Future extensions and user additions
- User configurable
 - Select elements you find useful
 - Combine as you like
 - Easier decision making on whether to include features

Performance Comparison

Towers of Hanoi	TSI (8.5.2)	Java Debugger (Text)	Java Debugger (Tree)
Watch 1 (Run 100)	1.25 secs	0.73 secs	0.75 secs
Watch 5 (Run 100)	59.68 secs	2.14 secs	1.51 secs 0.58 secs (full filtering)

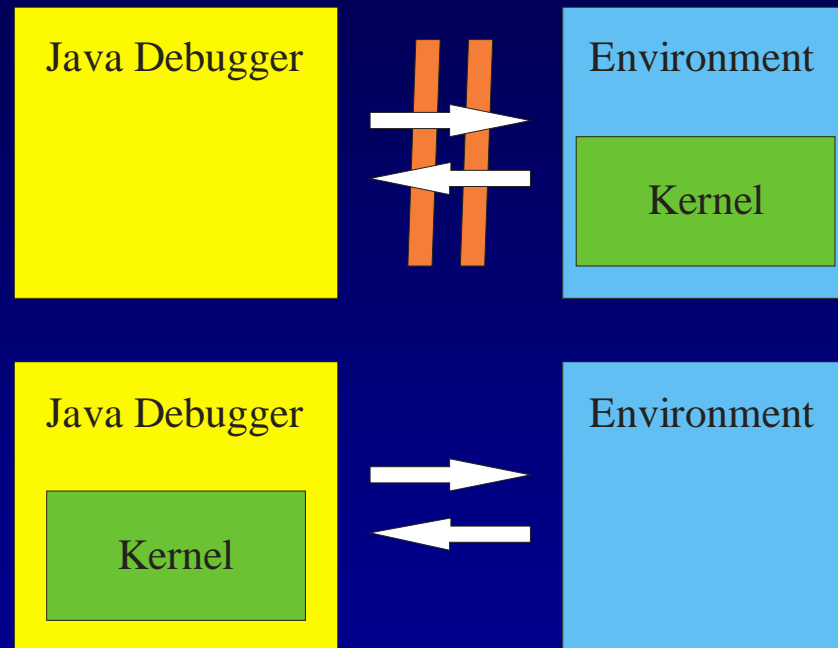
- Faster than 8.5.2 even when using XML
- Watch 5: 40 times faster than 8.5.2
- Watch 5 in 8.6. comparable to watch 1 in 8.5.2
 - In process and Towers of Hanoi

Remote Kernels & Dynamic Connections



- Debugger and kernel created and destroyed together
- No way to debug kernel embedded in environment

Remote Kernels & Dynamic Connections



- Connect and disconnect debugger as needed
- Embedded speed with debugging access
 - If remote connection and little tracing get embedded speed

Nuggets

- More power
 - Structured trace
 - Filtered trace
 - Integration with Visual Soar
 - Dynamic connection and disconnection
 - Debugging embedded kernels
- Higher performance
 - Order of magnitude improvements
 - Solid foundation
- More flexible
 - Customizable layouts
 - User plug-ins
 - XML based; no special access to kernel

Coal and the Future

- Early days
 - Mostly been building up the foundation
 - Lots and lots more we'd like to add
 - Lots of new code, so there will be some bugs lurking
 - “stop” looks really slow/unresponsive
- Documentation
 - “Intro to the Soar Debugger in Java.doc”
 - No documentation on the internals yet (beyond comments)
- Wishlist – please do speak up now
soar-sml-list@umich.edu or
winter.eecs.umich.edu/soarwiki/Debugger_wish_list