

Soar 9.3

Jonathan Voigt
voigtjr@gmail.com
University of Michigan
Soar Workshop 30
May 2010

Soar Versions

2007

2008

2009

2010

8.6.3

8.6.4 beta (RL)

9.0 (RL)

9.3.0 (RL/EpMem/SMem)

9.1 (EpMem)

9.2 (SMem)

New in 9.3.0

- Performance, correctness, stability throughout
 - Especially with respect to newer learning systems
- Improved, easier end-user experience
 - Clickable launchers on all operating systems, terminal no longer necessary for basic use
- GraphViz integration
 - Memory visualization tools for debugging the new memory systems

Long-Term Identifiers (LTI)

- An identifier stored in SMem is converted to an LTI
- If SMem is enabled: LTIs may exist in any symbolic memory system
 - Future retrievals are guaranteed to return the associated letter-number pair
 - Soar productions can be written to match against specific LTIs
 - When printed or used in productions, LTIs are prefixed with the @ symbol (e.g. @S5, @C7)

```
sp {match*lti
    (state <s> ^foo @F20)
    (...conditions...)
-->
    (...actions...)
}
```

On-Demand Memory Allocation

- `allocate` command increases pool sizes on-demand (in roughly 32kB blocks) instead of “just-in-time”
- Relieves need to throw out first run when doing timed experiments

```
allocate
```

```
Memory pool statistics:
```

Pool Name	Item Size	Itm/Blk	Blocks	Total Bytes
-----	-----	-----	-----	-----
wma_decay	200	163	0	0
chunk conditio	72	454	0	0
io wme	40	818	1	32720

Operator Selection Probability

- Operator selection probability included in `preferences` command output
- Works with all different kinds of preferences (numeric indifference, RL, etc.)

```
Preferences for S1 ^operator:
```

```
...
```

```
selection probabilities:
```

```
  O3 (fill) + :I (50.0%)
```

```
  O2 (fill) + :I (50.0%)
```

SML Output Link Change Tracking

- SML needs to track output-link changes to support many API functions
- Change tracking is convenient but expensive
 - Tracks everything, unnecessary for most applications

API calls dependent on change tracking:

- Commands
- GetCommand
- GetParamValue
- GetNumberOutputLinkChanges
- GetOutputLinkChange
- IsOutputLinkChangeAdd
- AddOutputHandler
- IsJustAdded
- AreChildrenModified

Not dependent:

- GetOutputLink
- GetNumberChildren
- GetChild
- All WME/Element calls

SML Output Link Change Tracking

- Can be disabled in 9.3
 - SetTrackOutputLinkChanges(false)
- Clients can track what they need to manually
 - It is usually much less work than what SML does by default
 - Can be performed during same callback as other I/O
- Major performance boost when no I/O
 - SML was tracking changes regardless
- Memory leak in output plugged

Not dependent:
GetOutputLink
GetNumberChildren
GetChild
All WME/Element calls

Inter-Process Communication (before 9.3)

- Problems with multiple listeners
 - First SML instance binds to TCP/12121
 - Future SML instances fail to bind and shut down listener thread (can't connect to these)
- Flawed workaround: specify listener port at kernel creation
 - Requires knowledge of available ports

Inter-Process Communication

(9.3)

- Default behavior unchanged (compatibility)
- Create kernel with special port values:
 - *kSuppressListener (0)*: No listener thread
 - *kUseAnyPort (-1)*: Bind to any available port
- Retrieve bound port with *GetListenPort*

Local Inter-Process Communication

- Higher-performance local sockets/pipes have been used since around version 8.6.3
 - Faster communication on *localhost*
 - Great for debugger connections
- Before 9.3:
 - Local sockets/pipes named using (unrelated) IP port number
- After 9.3:
 - kUseAnyPort: Local sockets/pipes named using process ID number
 - Pass process ID number instead of port to use local socket/pipe (when connecting to *localhost* kernels)

New Directory Structure

- Clickable launchers in root folder (all platforms)
- *bin*
 - Executable binaries and scripts, DLLs
- *include*
 - C++ headers for Soar/SML
- *lib*
 - Static and shared libraries
- *share*
 - Soar resources & documentation
- *share/java*
 - Jars
- *share/soar*
 - Example and demo Soar code, Documentation, other resources

Easy Integration

- Add *include* directory to compiler include path
- *#include <sml_Client.h>*
- Add *lib* directory to compiler library path
- Link to *ClientSML, ConnectionSML, ElementXML*
 - Static linking also possible, requires a few more libs
- Make shared libraries available to dynamic loader
 - Install to well-known location or modify environment variables (system dependent)

Cleaner Development

- Build intermediates and targets separated from source
 - Allows a single source tree to be shared with multiple computers/architectures
- Install step creates distribution tree
 - bin, lib, share, etc. folders same as release
 - Easy to target /usr/local or \$HOME/sandbox
- Modular checkout: Only get what you need
 - Build & update faster
 - Most only need Core, Java
 - <http://code.google.com/p/soar/wiki/ProjectLayout#Components>

Nuggets

- Corrected several deep-rooted stability bugs
- Having clickable launchers seems like a small thing but has received huge positive feedback
- Soar is now a few very short steps away from being a package (apt/deb/rpm/etc.)
- Trivial release preparation
 - Generated by every build

Coal

- Many outstanding issues:
 - <http://code.google.com/p/soar/issues/list>
- Windows OS support losing ground to Mac, Linux
- Still largely single-threaded