Soar Lite:
Designed for Speed
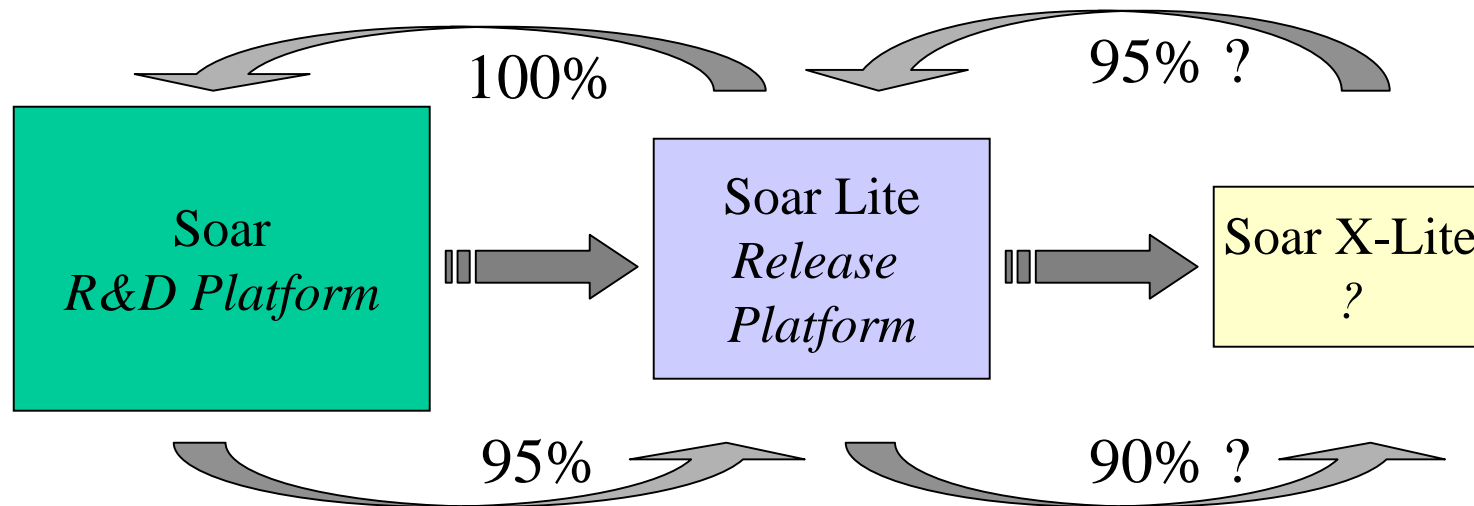

Karen Coulter

Scott Wallace

Motivation

- Meet performance requirements of some interesting domains
  - Video games
  - Mobile robots
- Want a embeddable, low-profile version
- Soar/CLIPS comparison revealed two things
  - simple modifications can greatly change performance
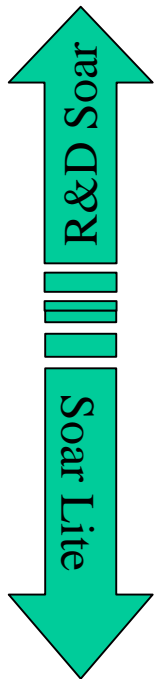 - some architectural capabilities appear very costly

Design Goals

- Increase overall speed
- Maintain current production semantics
- Keep a core set of the architectures capabilities
- Create an API

100%   95% ?

Soar R&D Platform → **Soar Lite Release Platform** → **Soar X-Lite ?**

95%   90% ?

On the Agenda

R&D Soar / **Soar Lite** (vertical arrow label)

- Look for memory leaks with Purify

- Profile with V-Tune (Intel Machines)

- Clean up execution cycle

- Examine efficiency of subgoaling mechanism

- Remove "extraneous" functionality

  – Majority of callbacks

  – Learning support (must be performed prior to release)

  – Debugging utilities