The Knowledge-Based Software Assistant Program

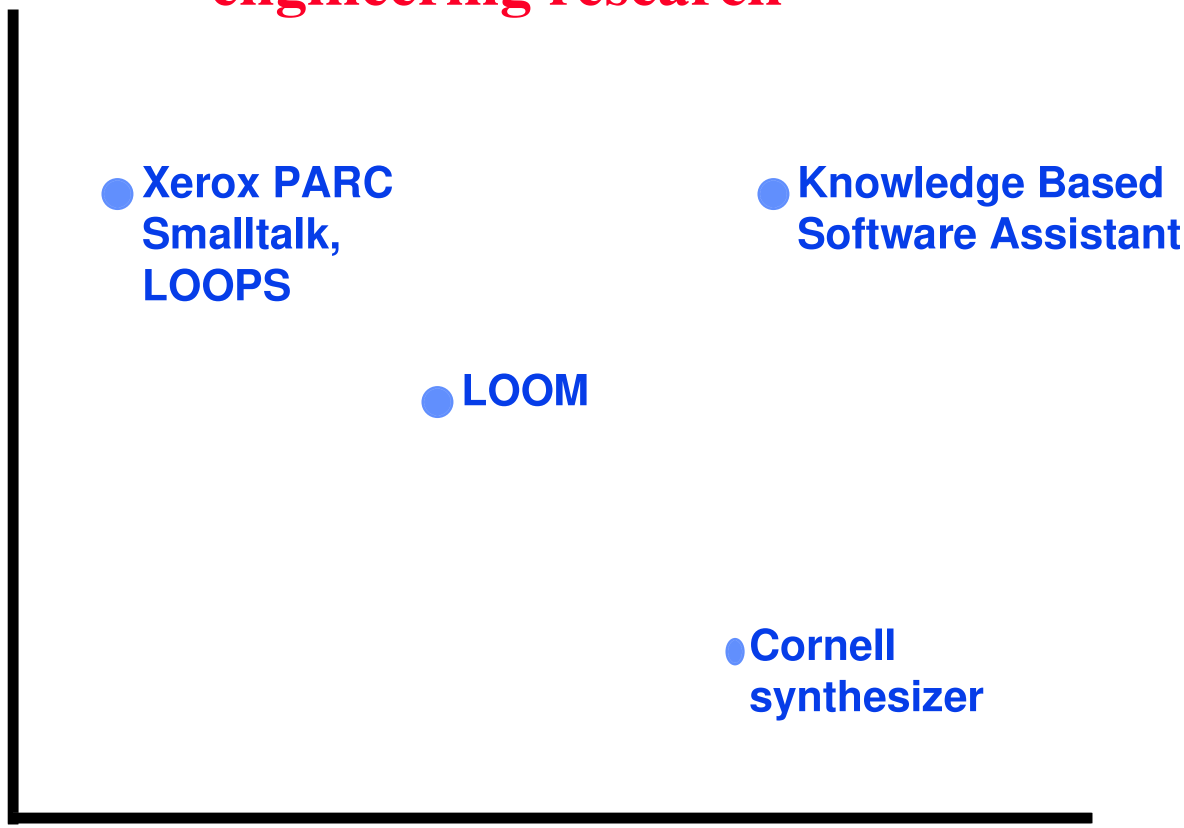
Michael DeBellis
Andersen Consulting
1661 Page Mill road
Palo Alto, CA 94304

michael.debellis@ac.com
(415) 842-2960

KBSA integrates knowledge & software engineering research

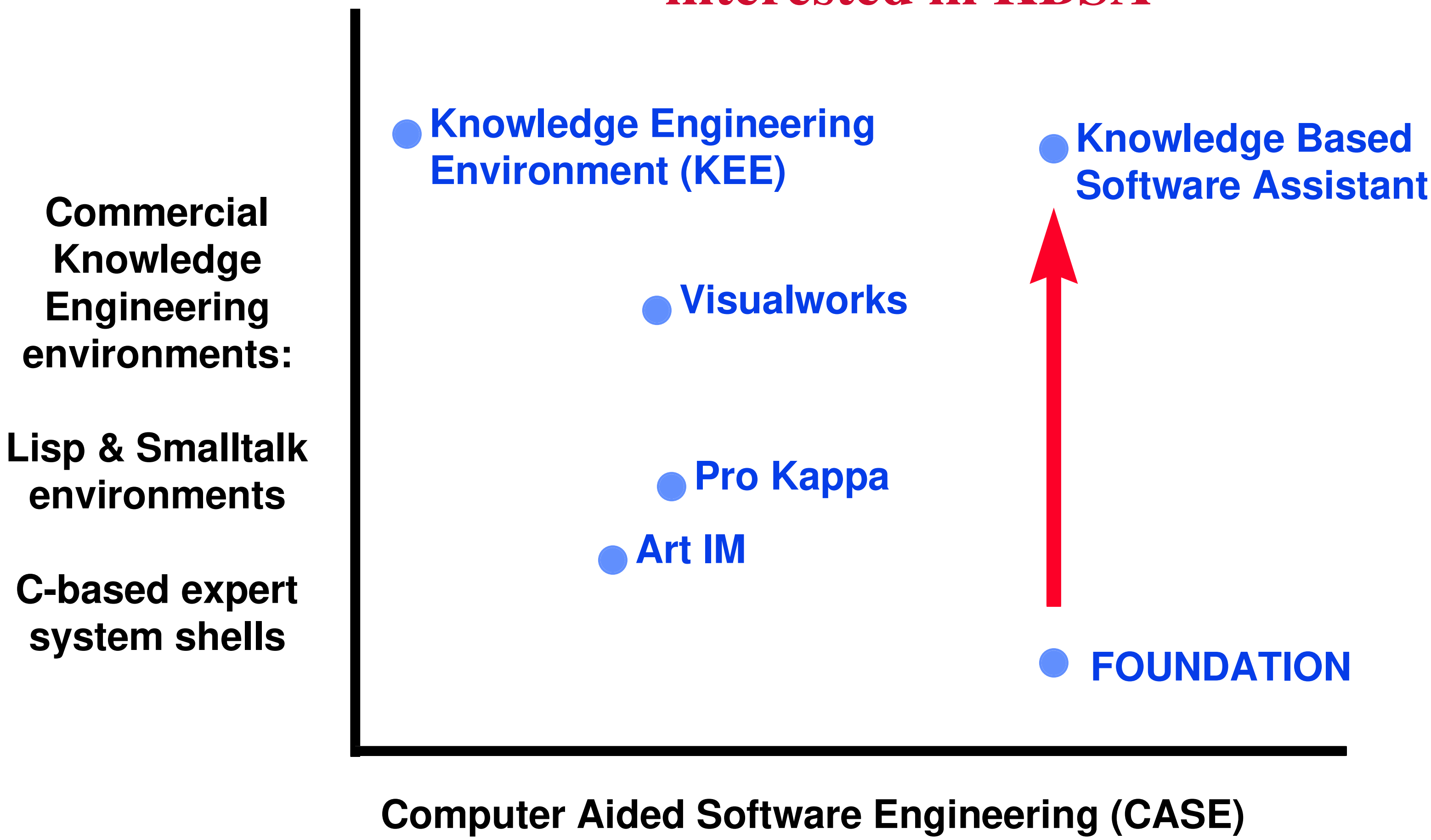
Knowledge Engineering:

Frame, rule, & object based development environments



Software Engineering : Formal methods, code generation, visual programming, ...

Why Andersen Consulting is interested in KBSA



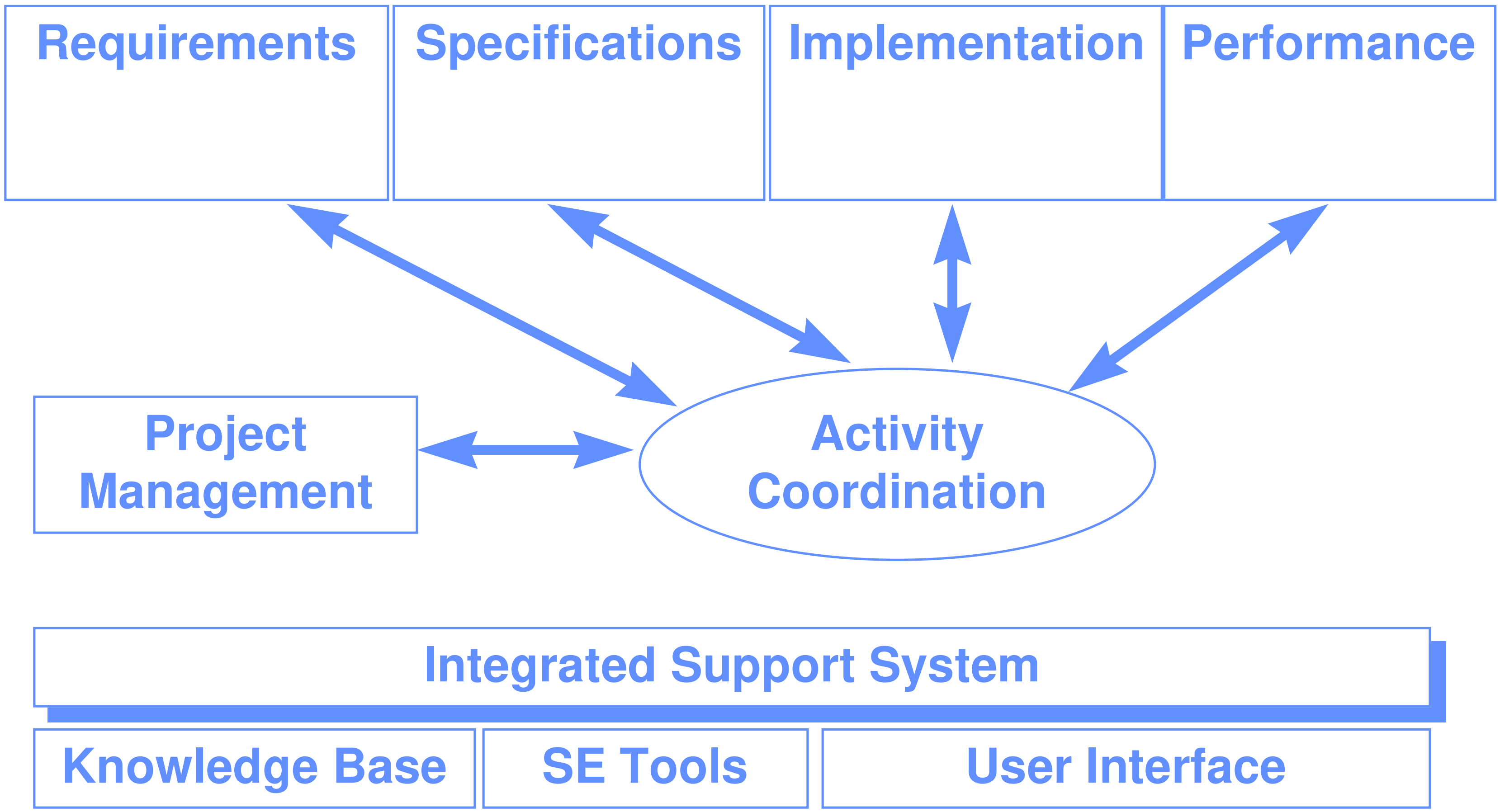
Initial Idea: Can we automate software development via AI?

- Success of rule-based systems applied to diagnosis in avionics domain
- Initial attempts
    - translation of natural language to code
    - automatic programming

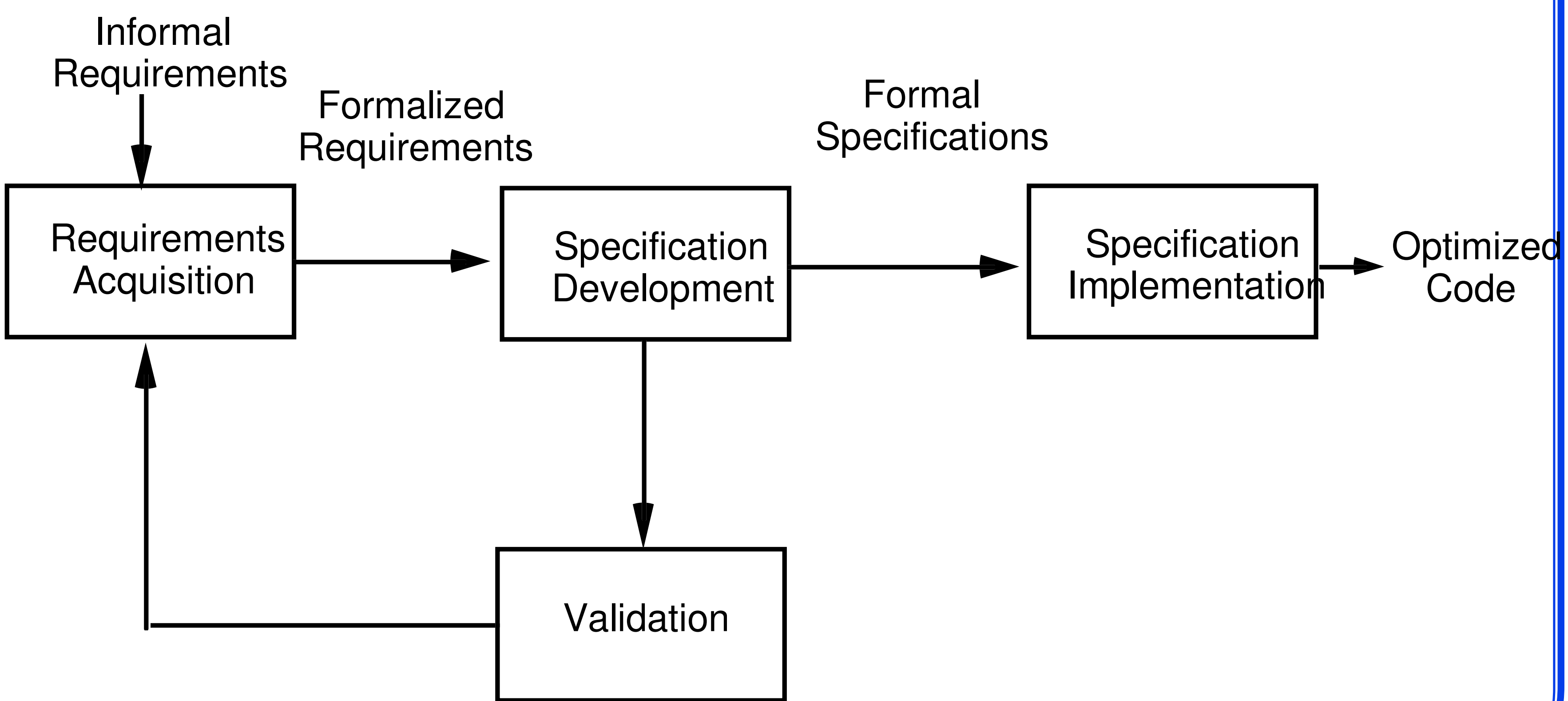
Initial Result: Can we automate software development via AI?
No!!!

- Design problems vs. deterministic problems
- Need for common sense reasoning
- Result: move from automation to "intelligent assistance"
    - seminal KBSA paper by Balzer, Cheatham, Green, Luckham, & Rich for Rome Laboratory

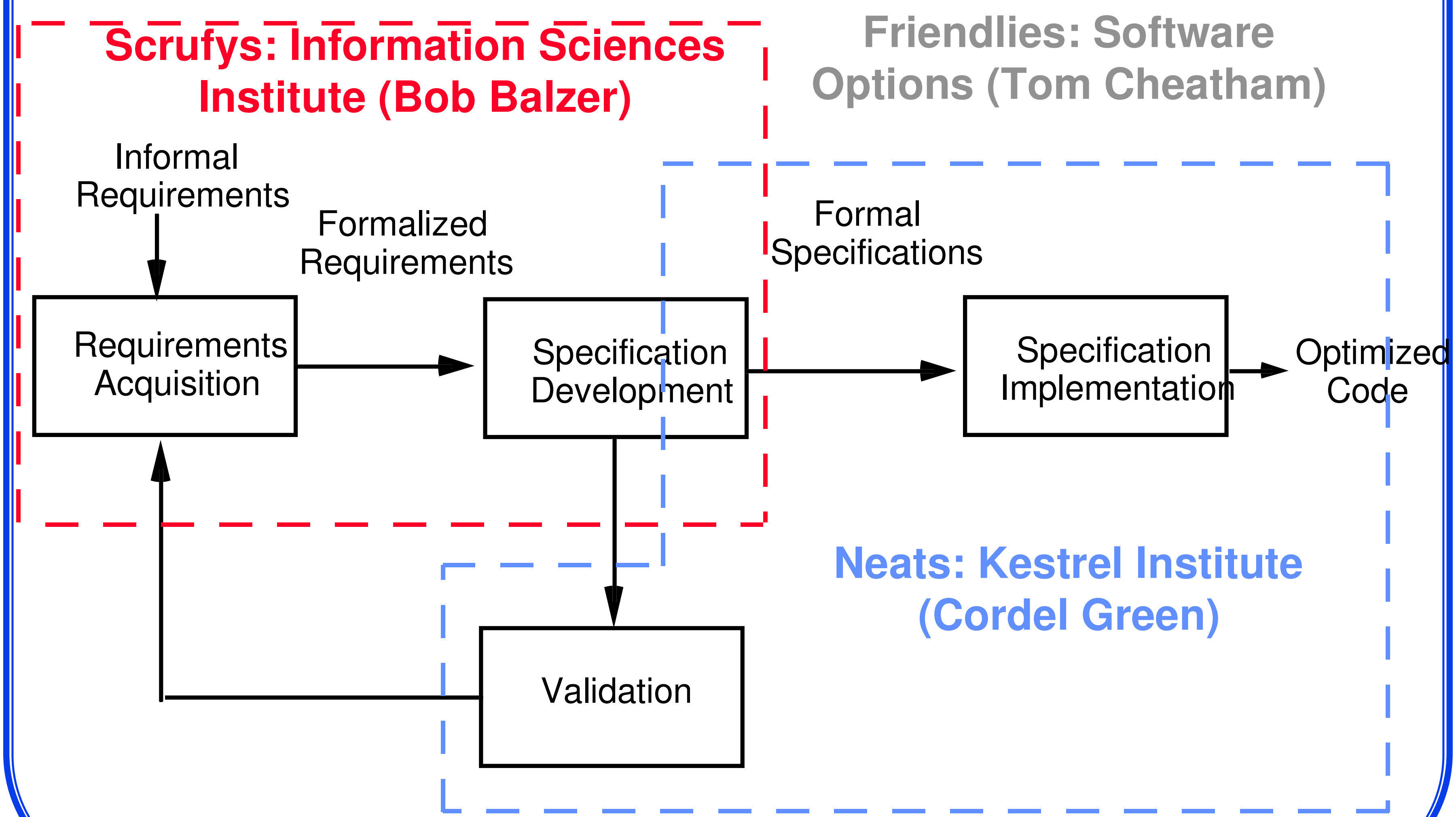
KBSA Architecture



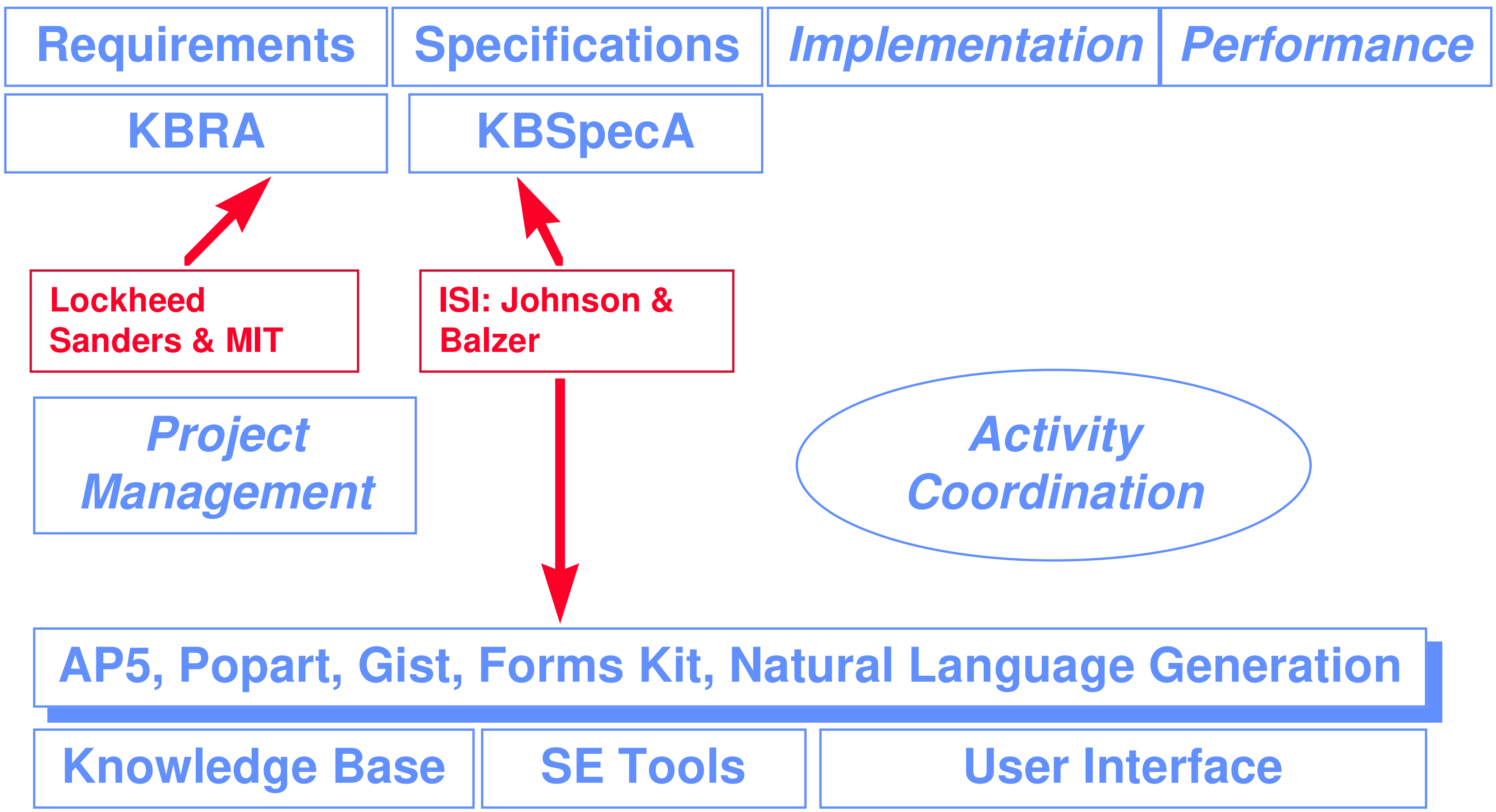
Simplified KBSA Process Model



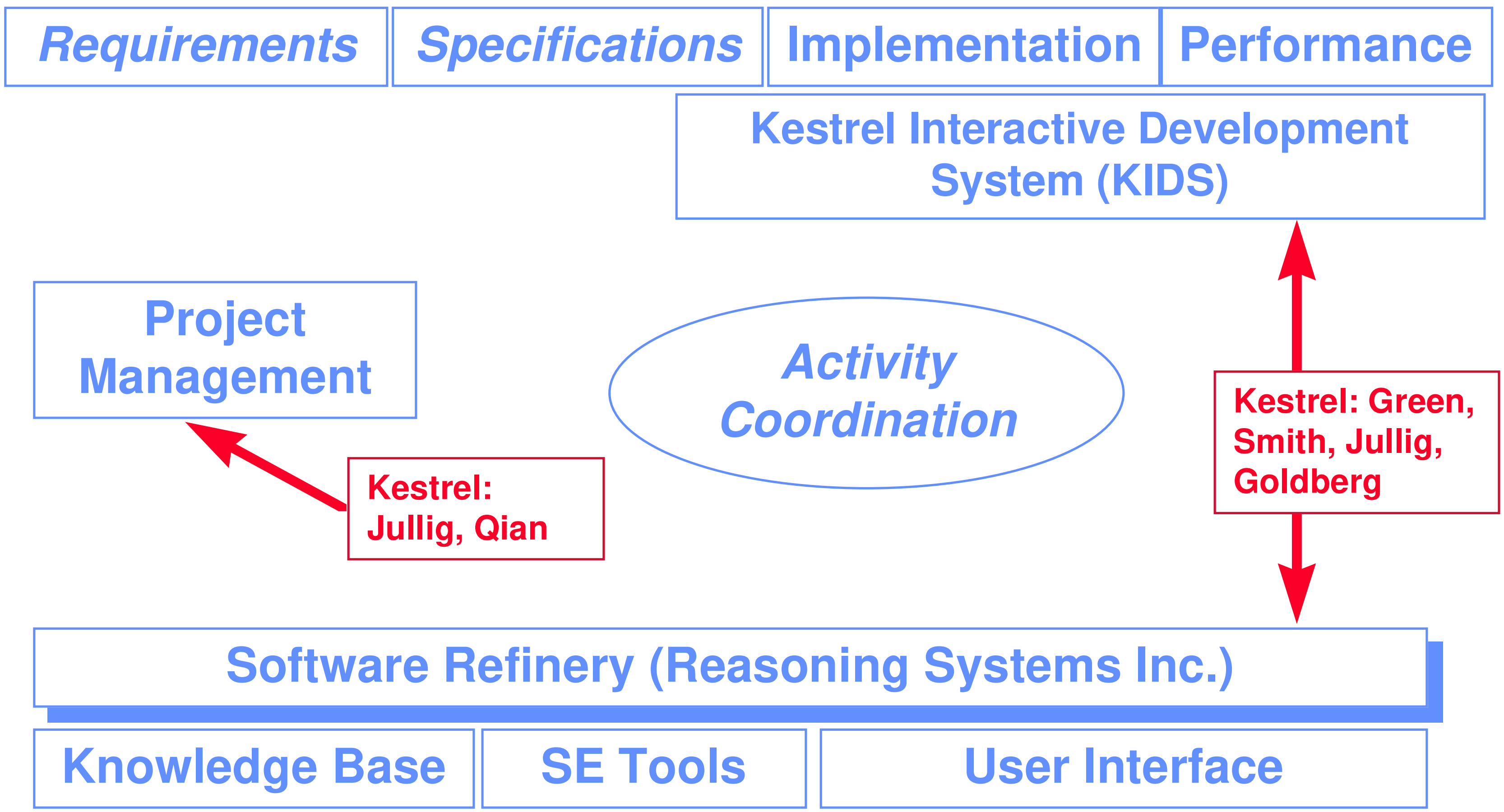
Scruffys, Neats & Friendlies



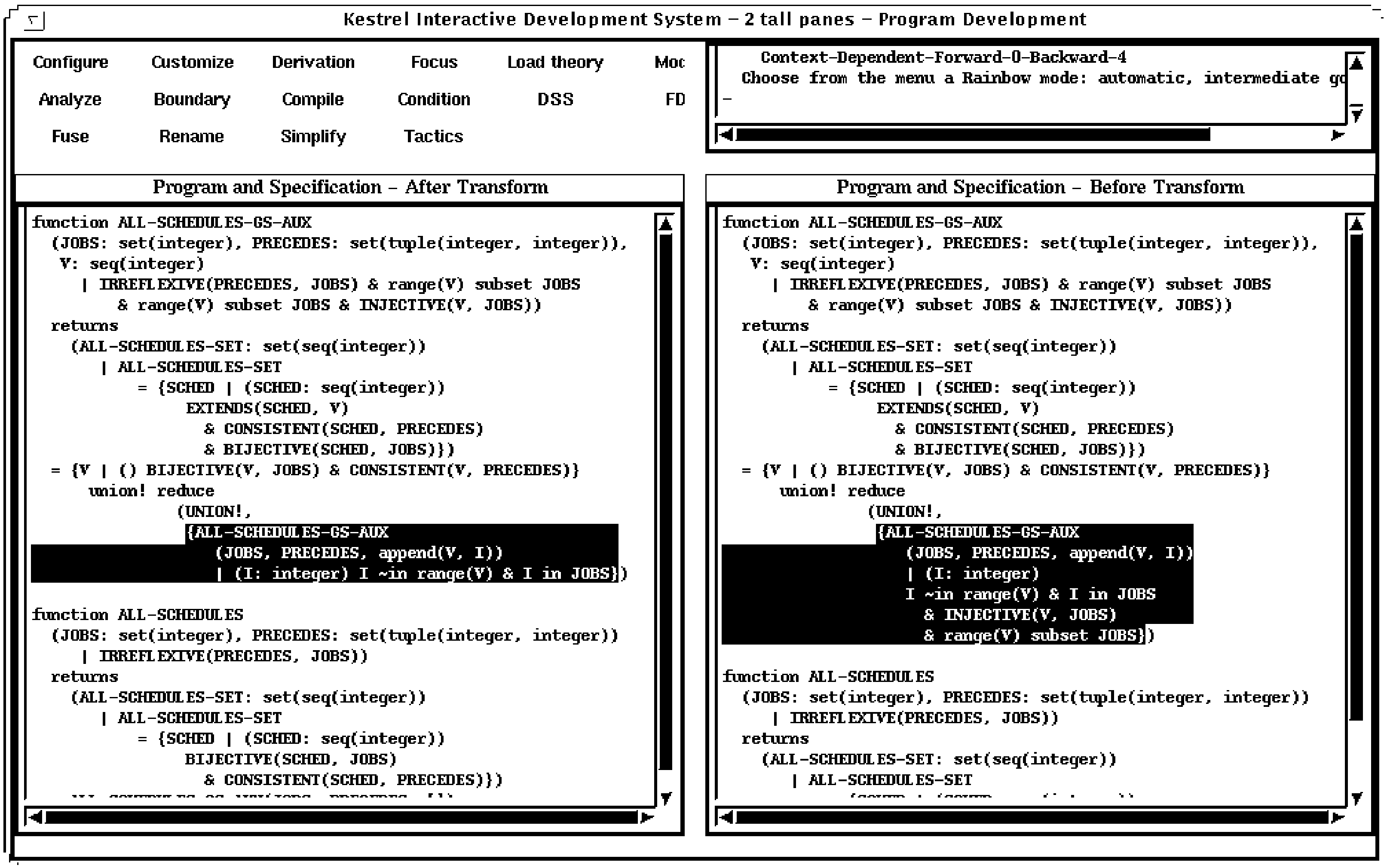
Scruffys: Information Sciences Institute



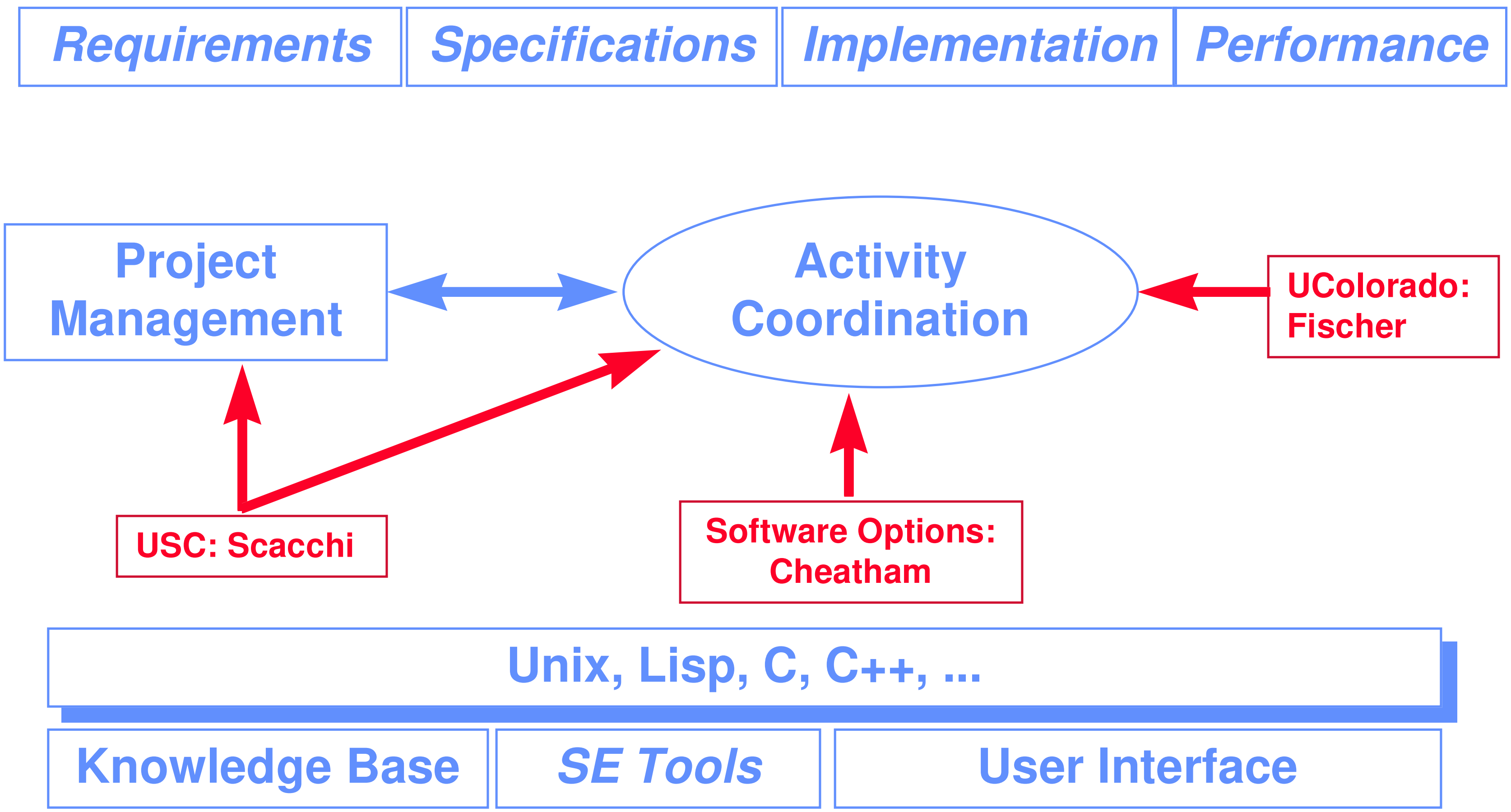
Neats: Kestrel Institute



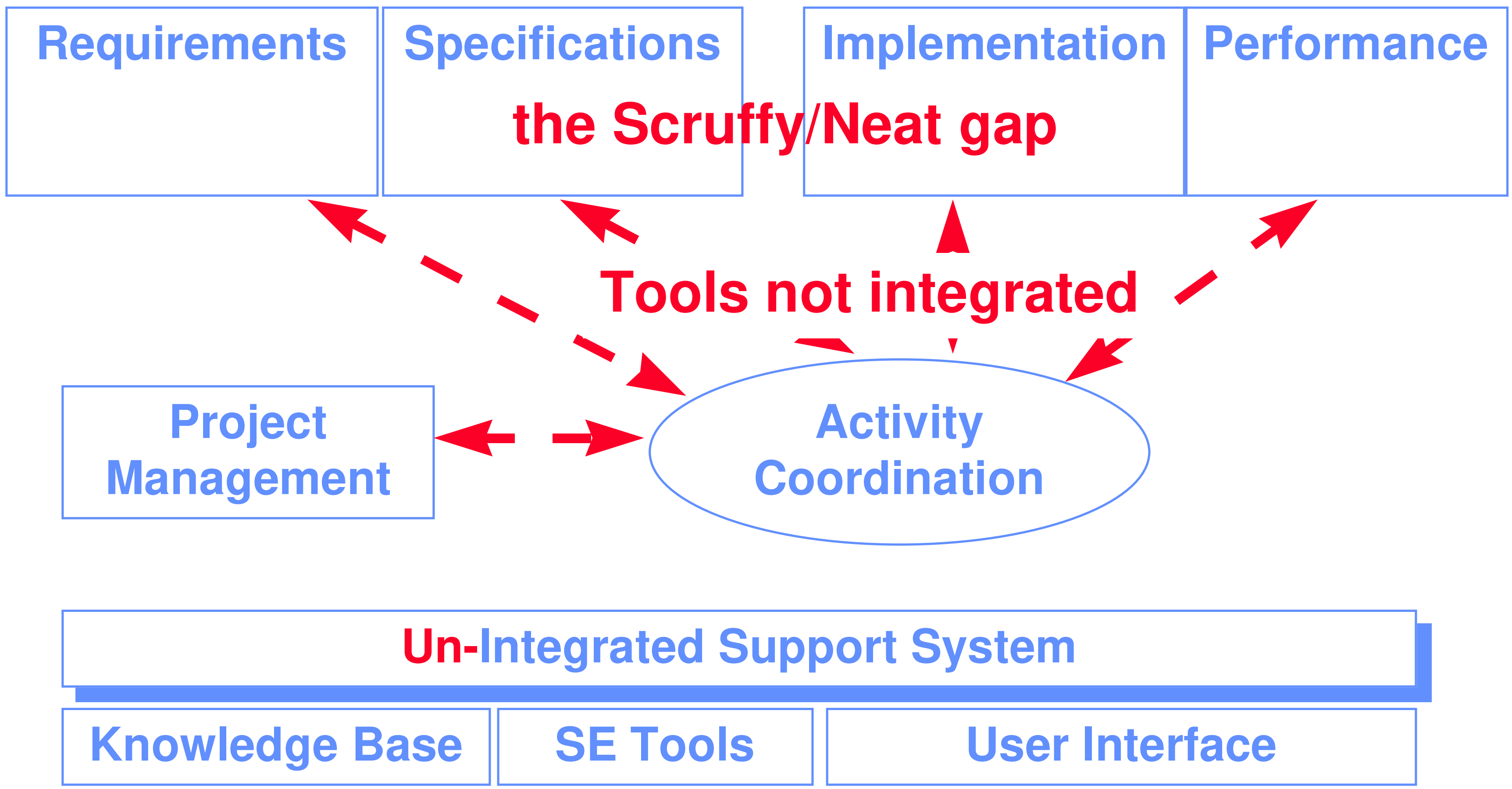
Kestrel Interactive Development System



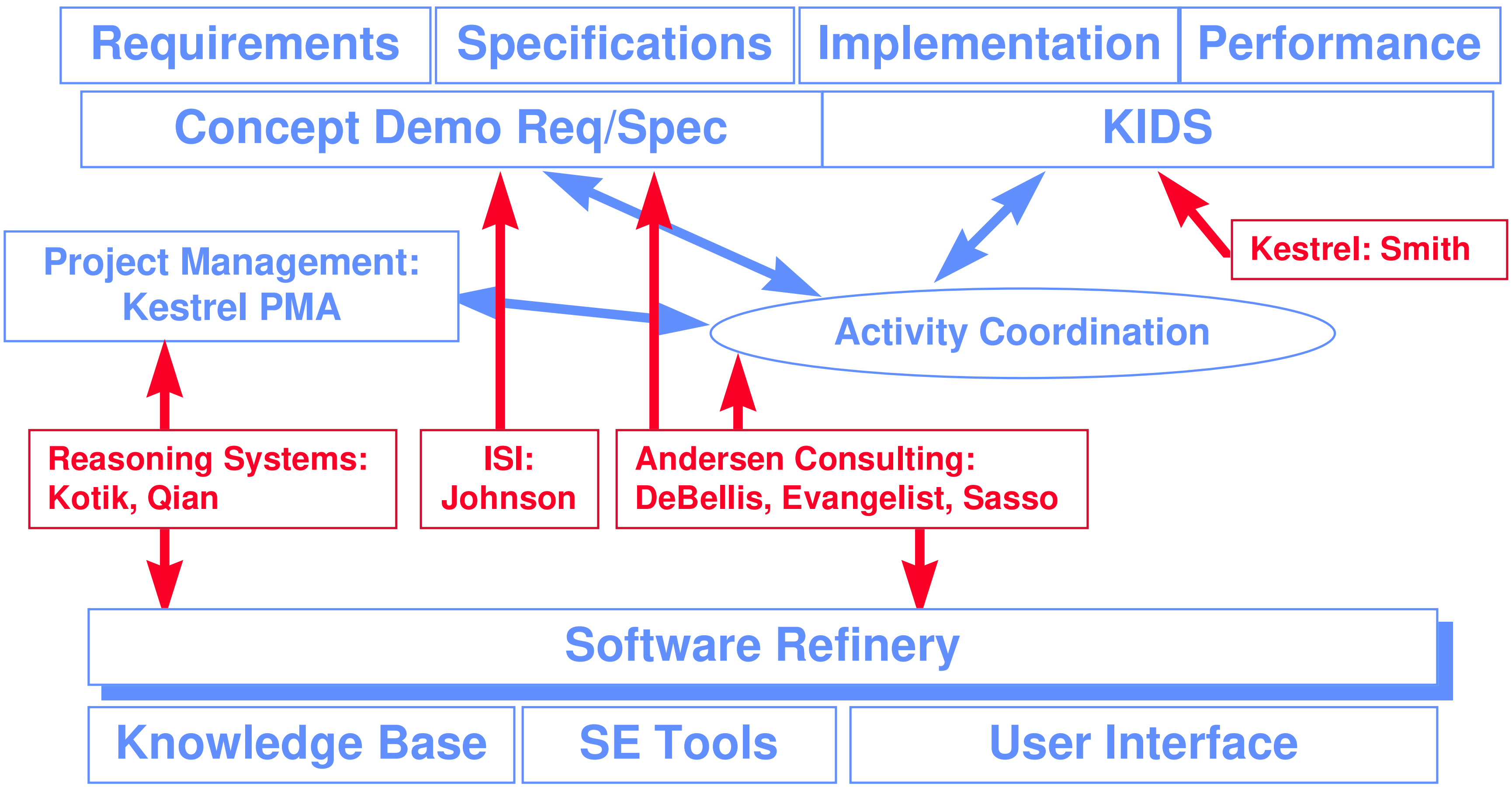
Friendlies: USC, Software Options, UColorado



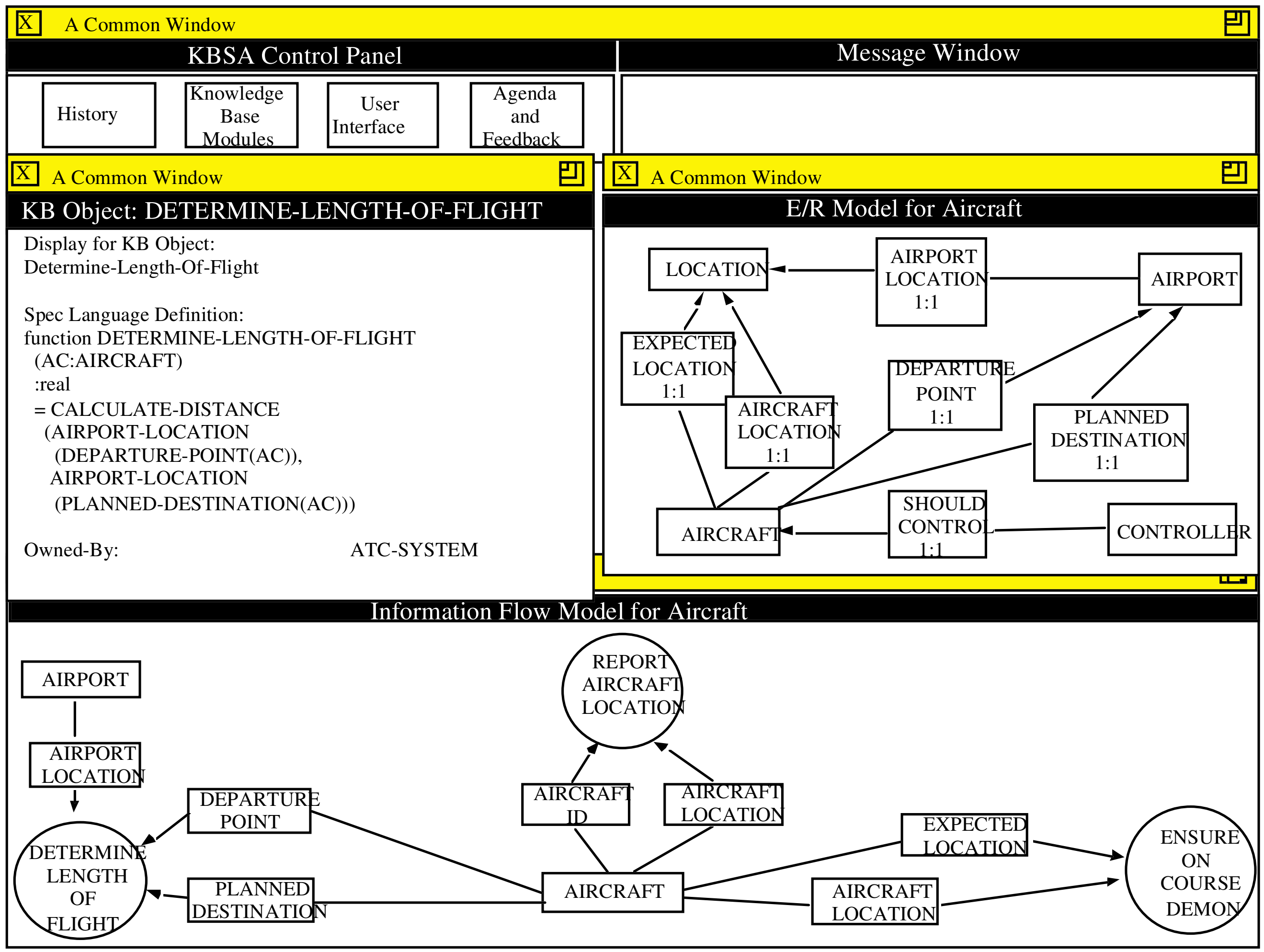
Problems from Phase 1 Development



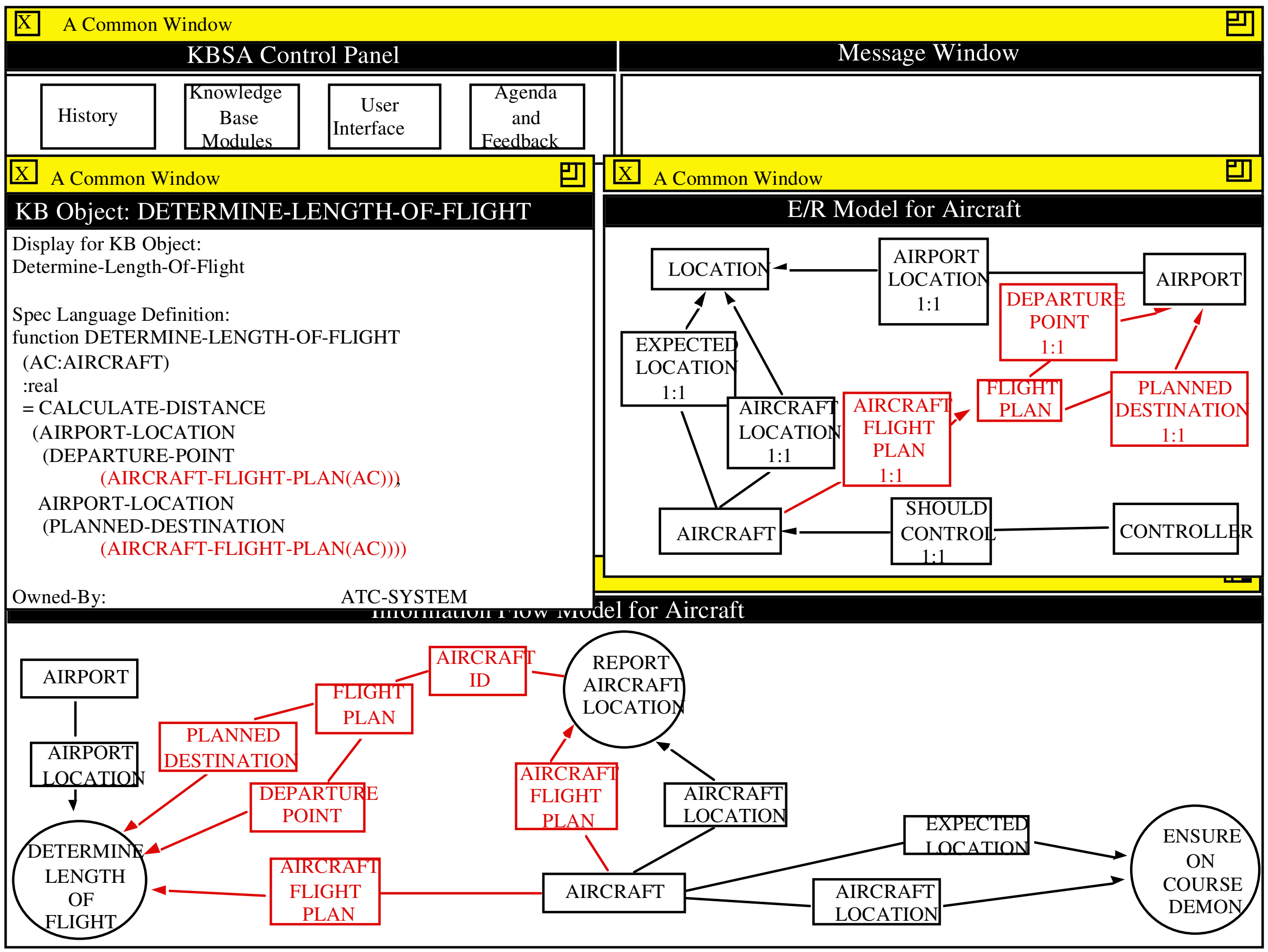
Phase 2 (integration): KBSA Concept Demo



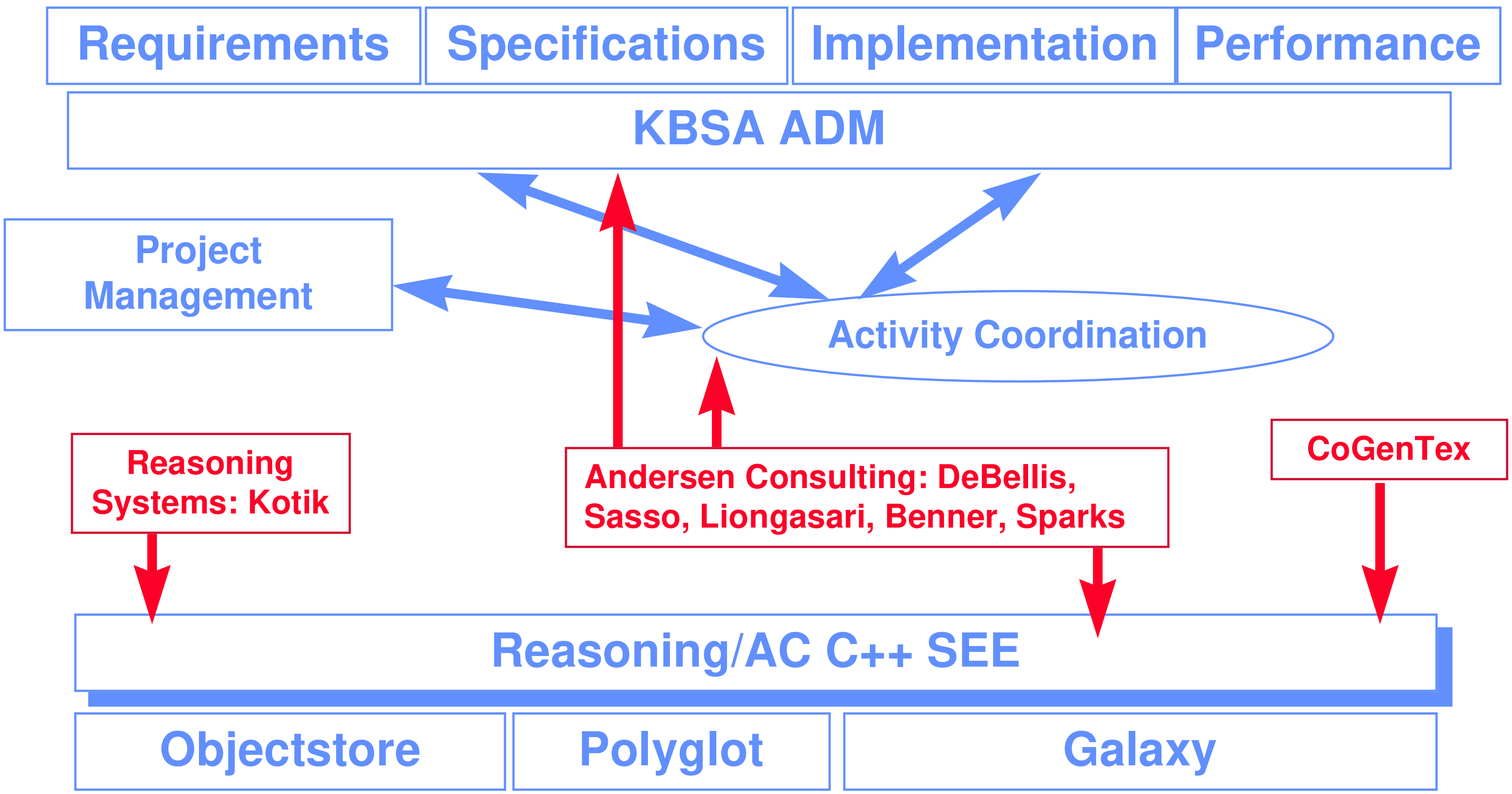
KBSA Concept Demo



KBSA Concept Demo



Phase 3 (advanced development): KBSA Advanced Development Model (ADM)



Some Personal Opinions: Where did we go wrong?

- Didn’t market well
- Ignored the end user
- Programmers should be coordinated not eliminated
  - Spec languages as high-level programming languages
  - Formal environments that require Ph.D. programmers

Some Personal Opinions: Where did we go right? software engineering

- Integrated graphic views of underlying spec language
- Specification evolution transformations
- Provably correct specifications and transformations
- Powerful tool for algorithm development
- Software re-engineering tools

Some Personal Opinions: Where did we go right? knowledge engineering

- Intelligent assistance is a better paradigm for expert systems
  - Text generation
  - Process modeling for individual and group guidance and coordination

KBSA impact on AC: Object-Oriented Architectures and Tools Project

- Use Object Request Brokers (CORBA & OLE) to integrate tools and applications
- Advanced Development based on KBSA and DSSA research
- $24M Andersen-led team with Raytheon, Expersoft, and CoGenTex; funded by Technology Reinvestment Program

Bibliography

DeBellis, Michael, and Christine Haapala, User-centric Software Engineering (USE), IEEE Expert, February, 1995, Volume 10, No. 1

Fischer, Gerhard, "Domain-Oriented Design Environments", Proceedings of the 7th Annual Knowledge-Based Software Engineering Conference, September 1992, McLean, Virginia, Rome Laboratory.

Green, Cordel, Luckham, D., Balzer, R., Cheatham, T. and Rich, C., "Report on a Knowledge-Based Software Assistant," RADC TR 83-195, Contract No. F30602-81-C-0206, Kestrel Institute, Palo Alto, California, June 1983.

Johnson, Lewis, Harris, D., The KBSA Requirements/ Specification Facet: Aries, Proceedings of the 6th Annual Knowledge-Based Software Engineering Conference, Syracuse, New York, September, 1991.

Smith, Doug, "KIDS: A Semiautomatic Program Development System," IEEE Transactions of Software Engineering, Vol 16, No. 9., September 1990