Practical Software Estimation
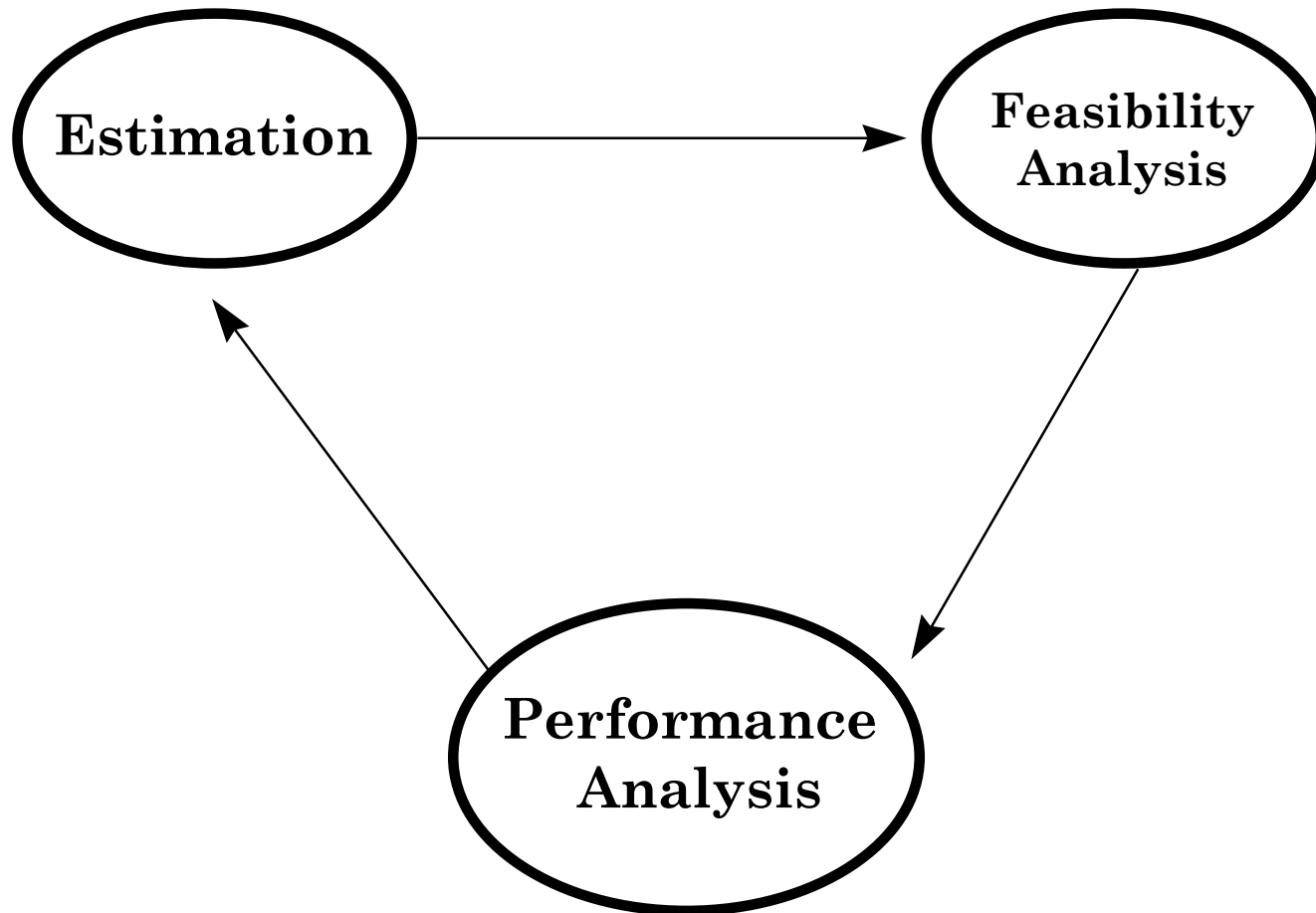
Betsy Bailey

Software Metrics Inc.

Doug Putnam

Quantitative Software Management
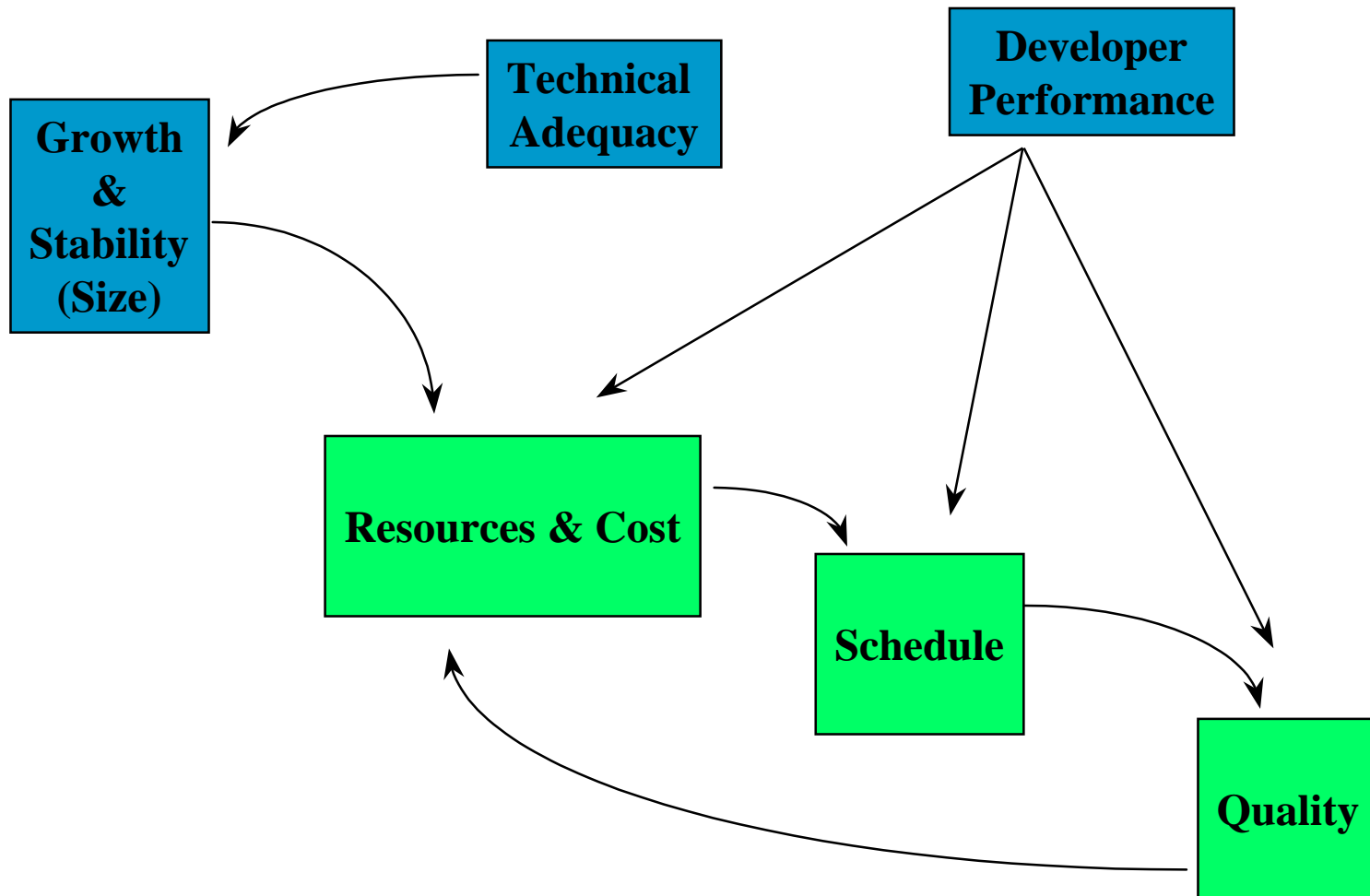
Three Primary Analysis Activities

Estimation → Feasibility Analysis → Performance Analysis → Estimation

Six Common Issues

- Growth and Stability
- Technical Adequacy
- Developer Performance
- Resources and Cost
- Schedule and Progress
- Quality

Issues Analysis "Roadmap"

Cost/Schedule Estimating:
Lessons Learned

- Size is the single most important input. Expect to spend a major portion of the time gathering/generating size estimates.

- Size estimates will be in the developer's terms (SLOC, words, components, objects).

- Understand what the numbers mean (e.g., if SLOC, what's included or excluded).

- Cross check size estimates with other information

Lessons Learned (continued)

- Estimates are essentially meaningless in the absence of historical data to ground them to reality.

- You can learn a lot about an organization from the data they provide.

- Understand the developer's process and how it does or doesn't map to your model.

- "Software maintenance" covers a huge range of activities that can differ drastically in their degree of fomality.

Lessons Learned (Continued)

- The job of the estimator is to make all assumptions and inputs visible so that the estimate can be replicated.

- Estimating is not a one-time activity.