Phase B Software Metrics Workshop

Jairus Hihn

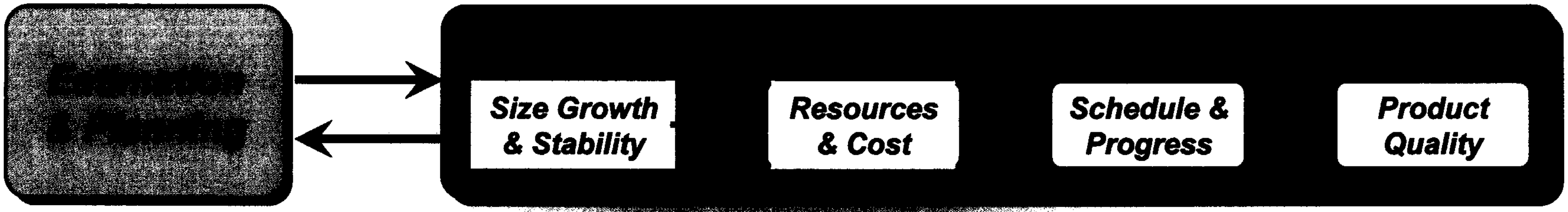
Software Quality Improvement Project

January 29. 2004

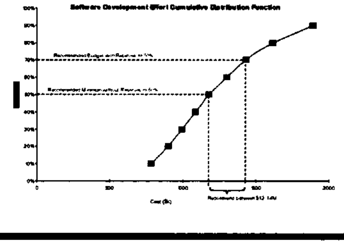
Measurement & Benchmarking

"Creating a Quantitative SW Management Culture"

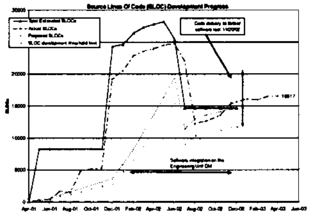
Helping Projects



- SW Cost Handbook
- Quantitative SW Management Class
- Estimation Support
- Estimation Tools
  - Flight SW Cost Model
  - Probabilistic COCOMO
  - Probabilistic Sizing Tool

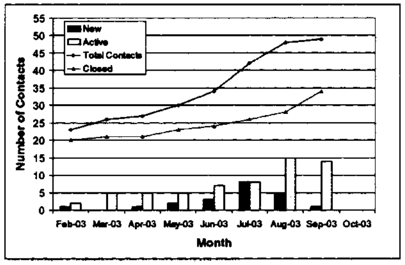


- Project SW Measures Guide
- Quantitative SW Management Class
- Measurement Support
- Software Repositories
  - Cost
  - Defects
  - Foundation Measures
- SW Engineering Models to support task planning



Organizational and Process Measures

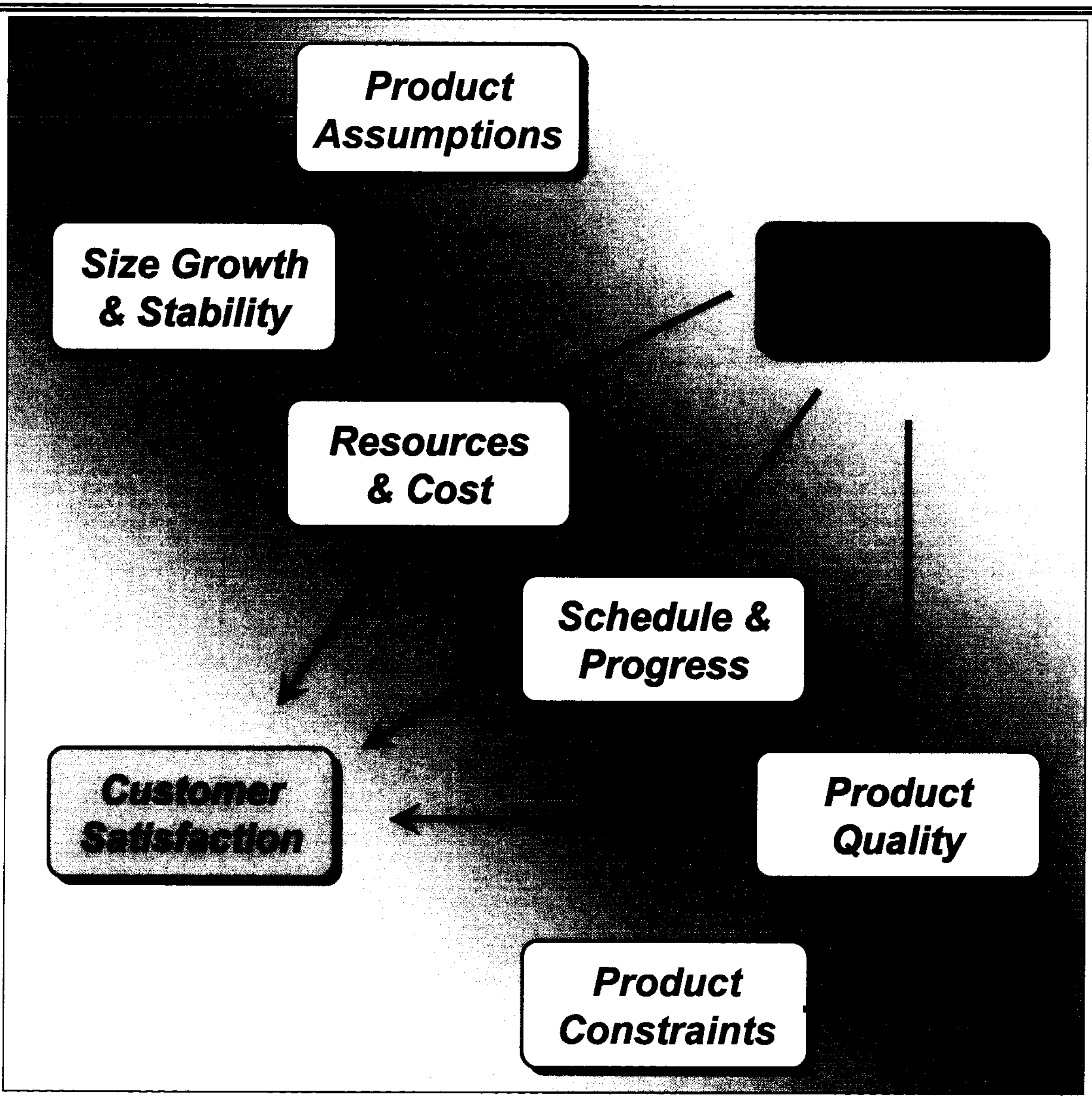
- JPL SW baselines and trends
- Measure Impact of SQI
- Benchmarking
  - Establish methods for conducting survey and experience gathering
  - Engage and collaborate with industry for best practices



SQI Approach to Software Measurement

There are many relationships among software measures hence you need more than one metric but too many just leads to confusion

In the Quantitative SW Mgmt Class we recommend at least 1 metric in each of 4 key areas



Product Assumptions

Size Growth & Stability

Resources & Cost

Schedule & Progress

Customer Satisfaction

Product Quality

Product Constraints

Basic Evaluation Criteria for ‘Good’ Metrics

- Metrics are a waste of time and money unless they
  1. have well defined goal/purpose
  2. are reviewed regularly and acted upon

- Metrics are more “accurate” when they are derived from
  3. Well defined completion criteria for products and intermediate products
  4. Disciplined development process

- Metrics will be maintained and not perceived as a burden when the
  5. Raw data used to construct the metrics are recorded as a natural part of work/process
  6. Artifacts and data are in electronic form

Lessons Learned and Current Project Experience

- We and our contractors have been more successful with Phase C/D metrics because code is 'naturally' consistent with items 1-6
  - –Code and unit test which are placed in a CM system
  - –Integration and test activities because you pass/fail the test
    - •There are still problems here because we do not use PFR system properly

- Phase B products are requirements, design, implementation plans. risk management plans, etc. and require some effort on the part of the project to manage and measure
  - –Phase B product components are more difficult to clearly identify
  - –Component quality and completion/acceptance criteria are more difficult to clearly identify

- Because of the nature of Phase B products completion and quality metrics must arise from
  - –peer reviews with check lists and defect/AI tracking
  - –MDS like development process which has requirements, implementation and verification packages (makes requirements more like code)
  - –Treat each requirement like a product component and then tracking requirements in a DOORS like tools (MRO)

Recommendations

- Except that there are too many metrics the proposed SIM metrics appear to meet the metrics evaluation criteria.

- Phase B metrics (in addition to standard cost and schedule reports)
  - –Earned Value like metric (very similar to our point counting/EV Light Methods)
    - •Total life Cycle Approach (makes requirements more like code)
    - •RTC SS Concept
  - –SIM Functionality Metric
    - •Combined with defect/AI tracking from peer reviews
  - –Requirements (similar to MRO)
    - •Traceability
      - –ICD's
    - •Volatility
  - –Risk & Lien Lists
    - •Liens as % of SW budget
    - •Liens as percent of reserves (use a 33-50% of available reserves rule)

Phase B Software Metrics Workshop

Jairus Hihn

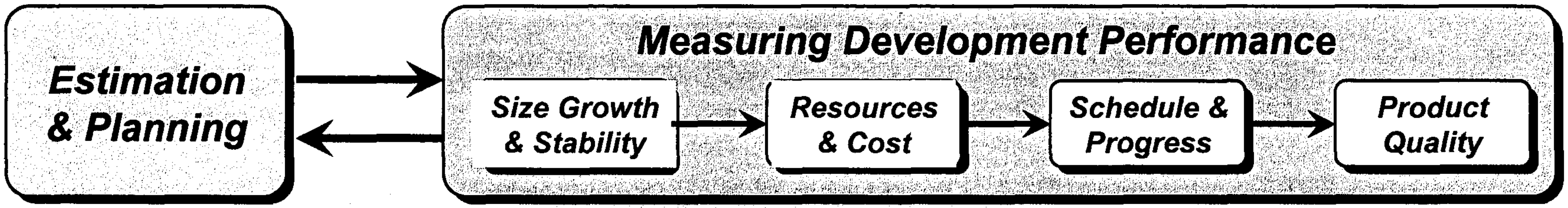
Software Quality Improvement Project

January 29. 2004

Measurement & Benchmarking

“Creating a Quantitative SW Management Culture”

Helping Projects



Estimation & Planning

Measuring Development Performance

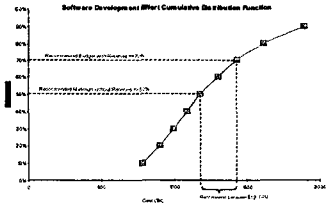
Size Growth & Stability

Resources & Cost

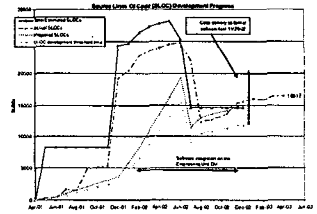
Schedule & Progress

Product Quality

- SW Cost Handbook
- Quantitative SW Management Class
- Estimation Support
- Estimation Tools
  - –Flight SW Cost Model
  - –Probabilistic COCOMO
  - –Probabilistic Sizing Tool

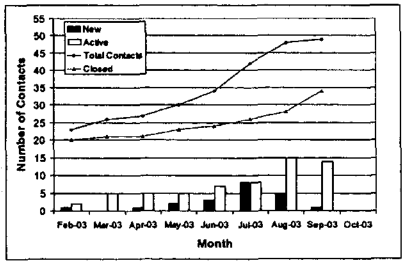


- Project SW Measures Guide
- Quantitative SW Management Class
- Measurement Support
- Software Repositories
  - –Cost
  - –Defects
  - –Foundation Measures
- SW Engineering Models to support task planning



Organizational and Process Measures

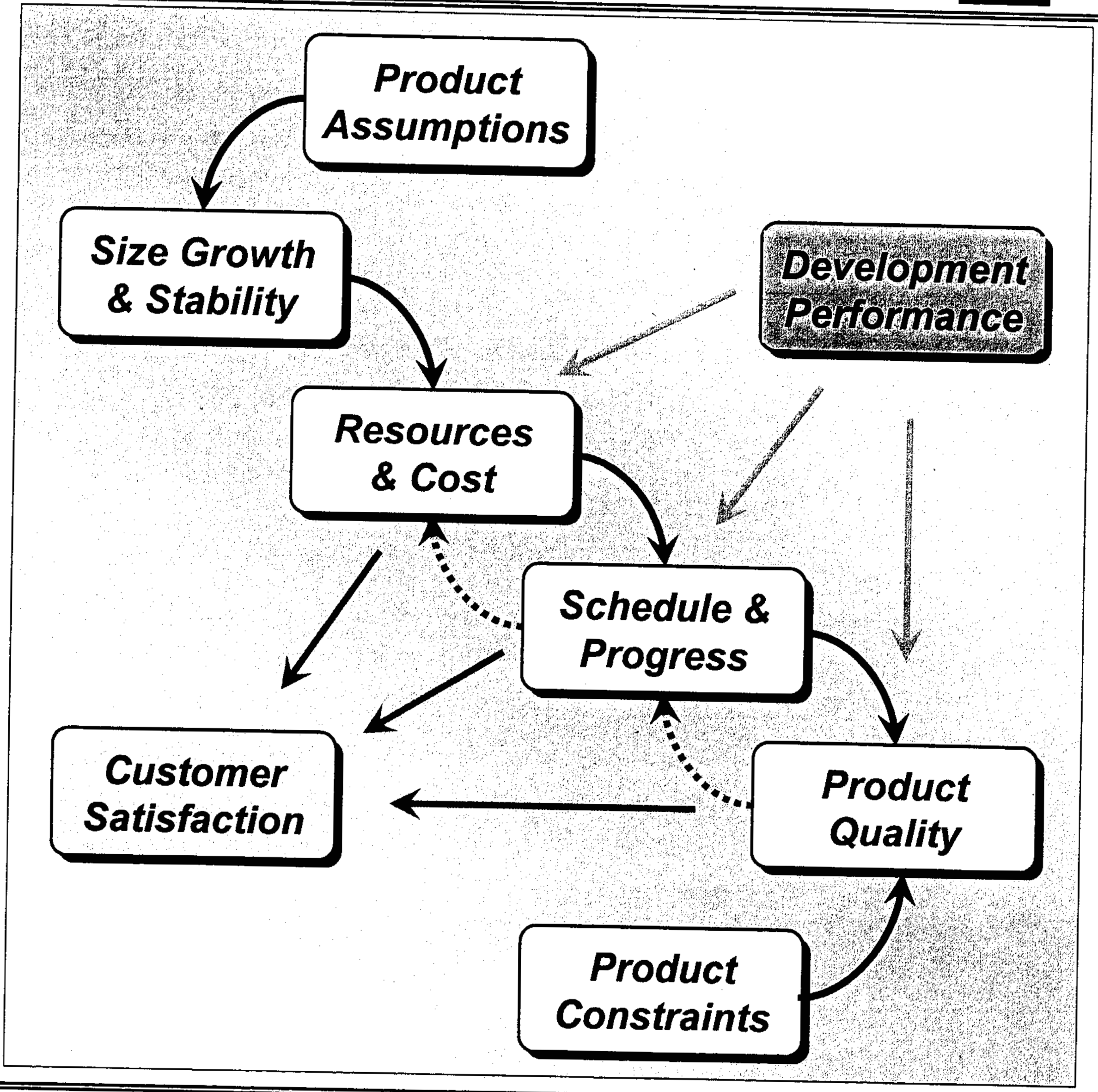
- JPL SW baselines and trends
- Measure Impact of SQI
- Benchmarking
  - –Establish methods for conducting survey and experience gathering
  - –Engage and collaborate with industry for best practices



SQI Approach to Software Measurement

There are many relationships among software measures hence you need more than one metric but too many just leads to confusion

In the Quantitative SW Mgmt Class we recommend at least 1 metric in each of 4 key areas



Basic Evaluation Criteria for 'Good' Metrics

- Metrics are a waste of time and money unless they
  1. have well defined goal/purpose
  2. are reviewed regularly and acted upon

- Metrics are more "accurate" when they are derived from
  3. Well defined completion criteria for products and intermediate products
  4. Disciplined development process

- Metrics will be maintained and not perceived as a burden when the
  5. Raw data used to construct the metrics are recorded as a natural part of work/process
  6. Artifacts and data are in electronic form

Lessons Learned and Current Project Experience

- We and our contractors have been more successful with Phase C/D metrics because code is 'naturally' consistent with items 1-6
  - –Code and unit test which are placed in a CM system
  - –Integration and test activities because you pass/fail the test
    - •There are still problems here because we do not use PFR system properly

- Phase B products are requirements, design, implementation plans. risk management plans, etc. and require some effort on the part of the project to manage and measure
  - –Phase B product components are more difficult to clearly identify
  - –Component quality and completion/acceptance criteria are more difficult to clearly identify

- Because of the nature of Phase B products completion and quality metrics must arise from
  - –peer reviews with check lists and defect/AI tracking
  - –MDS like development process which has requirements, implementation and verification packages (makes requirements more like code)
  - –Treat each requirement like a product component and then tracking requirements in a DOORS like tools (MRO)

Recommendations

- Except that there are too many metrics the proposed SIM metrics appear to meet the metrics evaluation criteria.

- Phase B metrics (in addition to standard cost and schedule reports)
  - –Earned Value like metric (very similar to our point counting/EV Light Methods)
    - •Total life Cycle Approach (makes requirements more like code)
    - •RTC SS Concept
  - –SIM Functionality Metric
    - •Combined with defect/AI tracking from peer reviews
  - –Requirements (similar to MRO)
    - •Traceability
      - –ICD's
    - •Volatility
  - –Risk & Lien Lists
    - •Liens as % of SW budget
    - •Liens as percent of reserves (use a 33-50% of available reserves rule)