

## Workshop #9: Defects/Anomalies Estimation and Management

27 July 2006

### Agenda

- Address questions such as setting goals for defect discovery rates, use of mathematical techniques, etc.; state-of-the-practice; what is the future of defect estimation techniques
- Address the goals of the workshop
  - Discussion notes (intros from each representative)
    - Raytheon
      - Capture more inspection and requirements level defect profiling
      - Organizational-level defect rates
      - Reliability estimation needs (from customer? From organization?)
      - Problems with the program shifts from one major phase into another (data starts fresh); need for defect data prediction
        - Reuse?
    - Lockheed IS&S
      - Need for more estimation of this data
      - Data exists, but it's all rear-view mirror data (trailing)
    - FMI Solutions
      - “bathtub curve” of defects – when you buy a new system, defects decrease, then level off, then near the end of the life cycle, defect increase again
      - defects more related to the team itself, less the item
      - they had the defect rates, need more analysis; threshold ranges
      - used waterfall process, but didn't understand problems until too late
  - need to understand where the defect was created, not just where it was found (accountability has a lot to do with this)
    - Many in the commercial world don't even want to know about defects – it's more crucial in the military
  - Manage toward the talent of your people
    - Liveware, Argentina
      - Consulting
      - Defects as related to business efficiency
      - The customers are telecom, banks
      - Need for more refined, more specific estimation; moving toward defect fitting
  - SPC

- The other side of finding these is not creating these in the first place
  - Lockheed MS2
    - Working to define the process
    - Programs have tendency to tailor-out the defects
    - Still have need to roll data up to organizational level
  - L3 communications, Iliff Comm.
    - Quality, process improvement effort – need to better analyze rolled-up data
    - Concerned mainly with % of rework
- Root-cause analysis
- Exploratory data analysis (EDA); focus on problem areas (forest / trees) – don't let the data overwhelm you
  - Boeing, Integrated Defense Systems
    - Defect profiling
    - Working to establish baselines, predictions
    - They have systems with zero defects?!
- Mathematical defect prediction? Cost prohibitive?
  - LM Space Systems
    - Struggling with common definitions of terms (due in part to legacy of diverse defense background – GE, Martin Marietta, etc.)
    - Trying to get legacy programs to adopt new standards
    - Working to achieve commonality across organizations
    - Need a defect density profile that is unique to their organization
  - Countrywide
    - Currently performing root-cause analysis
    - Migration schedule to new systems is based in part on defect rates
    - Still developing defect estimations
    - In-house software tools
  - GD, Canada
    - Data difference between programs
    - They are standardizing definitions, life cycle phase definitions, etc.
    - Performing SPC on the data, focusing on the inspection process
    - Trying to learn from successful programs
    - Next step is defect estimation – “it's hard to find anyone who really believes this will work”
- Perception is reality – sometimes this process is CMMI-independent, you have to get buy-in from the individuals
- The use of the estimation model sometimes will actually help the process improvement itself; it can help with definitions and understanding the process, which leads to natural tendencies to examine the process

- “the big bang always fails”
  - LM Corporate
    - Struggling to get program managers to understand that most of the time, they aren’t that different from historical data and what the estimates really are
    - “what is the value of metrics,” “what is the value to ME”
- Much better to get the bad news early, than face the gung ho “we’ll get it done”
  - Has been asked to roll-up defect data – the data is very unique, so be careful with the roll-ups
- LM Space Systems II
  - Working to understand defect definitions
  - LM21, brainstorming for process improvement
- USC / (Don)
  - Use of models to set goals – pre-release models, and post-release models
  - Rayleigh works very well pre-release, but due to 18-month schedules, data perturbations post-release
  - ODBC, orthogonal defect analysis
  - Look for root cause!
  - #1 question to answer – “have I tested enough?”

### Progressed into guided Rayleigh (and other models) discussion

- Rayleigh curves are not the only ones, decaying exponential, etc. – some curves may fit better to different programs, but Rayleigh tends to fit most (STEER, SWEEP)

(break)

surveys to gather state of defect data gathering, estimation

the higher up you go in CMMI maturity, the more important the defect tracking/estimation is; process optimization is also crucial

- Concern that there is a class of customer that will not drive you to a higher level of maturity

(hitting the slides)

- Trouble defining some key terms
- Potential trouble in testing – causes spike in the “ideal” defect discovery rates after release
  - Difference between finding the faults and finding the failures (should be the faults!)
- Time-based, phase-based (2 terms – phase, activity)
- Rayleigh model is just one model in a set of models called the Weibull models
  - They all have 1 peak

- All tend toward infinity at the right (in theory, you could work forever to find all the defects)
  - Cumulative version Weibull equation
  - The point is that with some data, even if it's not REAL accurate, you can predict defect rates later in the lifecycle
  - Area under the curve is the total cumulative number of defects
  - Usually needs 2 lifecycle phases; however, if due to management needs, some key data points can be estimated to plug into the equation
  - Data normalization is important to fit the curve
- One of the goals is to look for patterns – anomalous data is sometimes not representative of something else
- Assumption – your processes are consistent
- When you apply the model, keep in mind the constraints (apply intelligently)
- Very difficult, if not impossible, to verify interfaces; you will likely never be able to get rid of dynamic testing
- Are all items found inspections quantified as defects? Answer – you must clarify that with the program; this is something that must be bought-in to; the right answer to that question will vary between organizations and companies, but as long as the answer is standard, the data is useful.
  - Also must be consistent and not track defects per KSLOC in one phase and defects per page in another phase
- This applies not just to software – it could apply to number of drawings, or whatever
  - The units are unimportant (KSLOC, function points, etc.) – as long as they are consistent throughout the lifecycle
- The curve may not always be what you'd like it to be, but based on past history, "it is what it is" may apply