# Defects/Anomalies Estimation and Management Workshop
# 2006 PSM Workshop, July 27, 2006

**Facilitators: John Gaffney, Lockheed Martin: j.gaffney@lmco.com 301-721-5710**
**Chris Miller, SSCI: miller@systemsandsoftware.org 703-742-7284**
**Jeff Meek, Lockheed Martin, took notes: Jeffrey.meek@lmco.com 303-971-4376**

**Contents:**



## Pre-Workshop Description

**This is a half-day workshop**

***Prerequisites***

**Knowledge about and interest in defects/anomalies in software and/or systems (including hardware). It is desirable that attendees are interested in obtaining and using measures and indicators, such as the number of defects discovered in inspections and tests in order to improve both their software and systems development processes and to be able to establish goals for and manage toward the realization of such measures as latent defect content (i.e., number of defect in a delivered product) and reliability.**

***Materials to Bring***

**Data and/or experience information and/or company or organizational experience in setting goals for defect discovery in development and sustainment projects and for estimating/project defects during such projects. It would helpful if workshop participants share their experience and techniques for estimating defects, procedures for establishing defect-related goals and managing to them. Above all, bring interest and experience, and yes, questions that you have and are willing to share.**

***Discussion:***

**It is desirable that attendees are interested in obtaining and using measures and indicators such as the number of defects discovered in inspections and tests in order to improve both their software and systems development process and to be able to establish goals for and manage toward the realization of such measures as latent defect objectives (i.e., number of defect in a delivered product) and reliability. We might try to answer questions such as:**

- **Do you set goals for defect discovery rates and related measures such as latent defect rates and number of escapes? If you do, what drives the selection of the goals, process improvement objectives, customer requirements, or what?**
- **Do you use mathematical techniques for defect estimation and projection? If so, what are they? If so, are these techniques imbedded in a tool, and if so, what tool?**
- **Where do you see defect estimation techniques and the like going in the future? Do you perceive a business need driving their (increasing?) use or not?**

***Goals/Products***

**Share experience and data if possible. Document some sense of use and practice in defect/anomaly management, including setting goals, tracking/estimating, and taking action. Also, document perceived need for improvements in defect modeling, management of defects, and related matters.**

## List of Participants

| Name | Affiliation/Company | e-mail Address | Phone Number |
|---|---|---|---|
| Jennifer Davenport | Raytheon | Jennifer.l.davenport@raytheon.com | (972) 205-4301 |
| Joe Lindley | Raytheon | Joe.h.lindley@raytheon.com | (972) 205-7281 |
| Jeff Meek | Lockheed Martin | Jeffrey.meek@lmco.com | (303) 971-4376 |
| Ray Ashman | FMI Solutions | Ray.ashman@fmisolutions.com | +442073314315 |
| Alejandro Bianchi | Livewire S.S.A. | Alejandro.bianchi@livewire.com | +4114601700 |
| Shally Malhotra | Lockheed Martin | Shally.malhotra@lmco.com | (856) 359-1805 |
| Vickie Papia | L-3 Communications – ILEX Sys | Vickie.papia@L-3com.com | (732) 380-9400 |
| Marie Mueller | Boeing | Marie.a.mueller@boeing.com | (310) 364-9984 |
| Craig Stauffer | Countrywide | Craig.stauffer@countrywide.com | (805) 577-3960 |
| Mike Ferris | General Dynamics Canada | Mike.ferris@gdcanada.com | (613) 356-4578 |
| Elliot Troy | LM SSC | Elliot.troy@lmco.com | (303) 971-3119 |
| Gary Hafen | Lockheed Martin | Gary.hafen@lmco.com | (661) 572-7178 |
| Michael Aberg | Lockheed Martin | Michael.aberg@lmco.com | (408) 743-7846 |
| John Gaffney | Lockheed Martin | j.gaffney@lmco.com | (301) 721-5710 |
| Chris Miller | SSCI | miller@systemsandsoftware.org | (703) 742-7284 |
| Don Reifer | Reifer Consultants & USC | dreifer@earthlink.net | (310) 530-4493 |

## Notes on Workshop  Including Matters of Discussion

**The Workshop discussion consisted of three main parts:**

1. **Discussion of the state of the practice in defects/anomalies management;**
2. **Poll of the participants in response to some questions relating to the state of practice;**
3. **Presentation of discussion of some of the techniques of defect analysis (by John Gaffney).**

## State of Practice in Defects/Anomalies Management

Sources of comments indicated by affiliation

- Raytheon
    - Capture more inspection and requirements level defects
    - 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 is 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

- Instead of finding defects, it preferable to not create them 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 an estimation model sometimes will actually help improve the process; it can help facilitate or lend impetus to sharpening definitions and enhance understanding 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?”

## Poll of the Participants in Response to Questions Relating to the State of Practice

1 Do you collect defect data? A majority said “yes.”

2. Is there “buy-in” (sponsorship) for doing defect modeling? A majority said “yes, BUT, this is NOT at all levels of the organization.

3. What do you view as your next major challenge in defect modeling?

4. Do you use tools to support defect analysis? All said that they used something, at least a tool developed in-house to facilitate data collection. Two participants said that they use a third party tool, such as SWEEP.

5. What drives your process improvement activities in the defect area? Various drivers were cited, including: desire/need to attain higher CMMI levels; corporate management; customer; desire to reduce rework; desire to reduce the number of escapes/leakage

6. Do you set defect discovery goals? There seemed to be little of this done. There was some discussion about the seeming desirability of “managing” defect discovery n terms of setting goals fro defect discovery, monitoring their degree f realization, and taken corrective or other action based on discrepancies if any between the goals and the actuals. It was further noted that such a process could be viewed as analogous to that of managing cost and schedule

## Presentation of Some Slides Relating To Analysis

Discussion of Rayleigh and other models

- 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)

- Sometimes, it is difficult to obtain agreement or common understanding on the definition of some key terms, such as “defect.”
- 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, except for the Exponential.
    - All tend toward infinity at the right (in theory, you could work forever to find all the defects)
    - The point is that with some data, even if it’s not very accurate, you can predict defect rates later in the lifecycle
    - Area under the curve is the total cumulative number of defects
    - Usually need data for 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

## Slides Presented by John Gaffney

### Problems or Failures

- **Various terms are used for *failure, problem,* etc.**
  - **There are no really universally agreed-upon definitions, and often one that is used in one instance may not be desirable in another due to the "political" baggage that it carries.**
- **The fundamental idea is that a *problem, defect, failure,* etc. are words to cover the concept of deviations of a system or of a software or a hardware element of a system from its requirements or the standards to be followed in its construction.**
- **The focus here is to how to determine (estimate) the mean (average or expected) time between *countable or relevant* failures, commencing at some *point in time after delivery* of the system.**
  - **The estimate is based on data obtained during the development and testing of the system plus data about prior systems. Therefore, the better the data and projection models, the better the estimate.**

### Major Uses of Defect Models

- **Estimation/Prediction: Use to ensure (at some level of confidence) that a proposed system will be able to meet its requirements. Will it be feasible with respect to defect based measures (e.g., reliability)?**
- **Comparative Analyses: What is the defect content of other (similar, if possible) systems at delivery or at some particular time after?**
- **Development Control: We should set goals for the defect discovery of the software and the hardware in a system to be developed**
- **What do we have to do to have confidence that the system that we are developing will meet its defect-related objectives?**

**Development methodology? Test methodology? Estimation methodology?**

**Overview of Time-Based Software Failure and Reliability Models-1**

- **Software failures are typically modeled as though the failure rate (failures per unit time) is a function of calendar time (it is actually a function of use). The reliability is the inverse of the failure rate (times a constant).**
  - **Over the life cycle, commencing at the beginning of integration, the failure rate typically initially increases and then decreases.**
    - **Often modeled as a Rayleigh curve (one of the family of Weibull curves)**
    - **Cumulative Version of Weibull:**

    **N(t)=E*(1-(t/c)x);where: E=total number of findable failures or defects; N(t)=number of failures from time 0 to t; x=shape parameter (x=1 for exponential and 2 for Rayleigh );c=scale parameter.**
      - **More convenient form: $N(t)=E*(1-b*t^x)$; where: $b=1/cx=v/t_p^x$**
      - **V=a number that depends on x ;$t_p$ is the location of the peak (for x>1.0) of the curve (failures or defects found versus time).V=0.5 when x=2.0 (for a Rayleigh distribution).**
    - **Post-integration and post-delivery, the rate is modeled as a monotonically decreasing function of time**
      - **Often modeled as a decaying exponential curve (also one of the family of Weibull curves)**

**Overview of Time-Based Software Failure and Reliability Models-2**

- **Although the Weibull models represent the post-delivery rate as a decreasing function of time:**
    - **It may be convenient for planning purposes to model the post-delivery failure rate for software as a constant, at least after some point in time.**
    - **It is likely that there will be a “defect surge,” a “bump” in defect discovery, for a period**

**immediately after delivery, because of additional error paths opening up due to differences of the testing environment from the operational environment. Model this as an addition or "delta" on top of the Weibull curve.**

- **When we have estimated the mean value function for failure occurrence, $\lambda(t)$ , we can obtain the corresponding estimate for the mean time between failures, MTBF, as $(1/ \lambda(t))$ .**
  - **For example, if $\lambda(t) = 5$ failures per day, at some value of t, then the MTBF= 0.2 days between failures, or perhaps more conveniently expressed, 4.8 hours between failures, at that time.**
  - **At each point in time, t=t0 (think of an interval of time, practically speaking), the expected number of defects to be found is $\lambda(t0)$, and the actual number is distributed according to a Poisson distribution, with mean $\lambda(t0)$ and standard deviation = $\mathrm{sqrt}(\lambda(t0))$.**

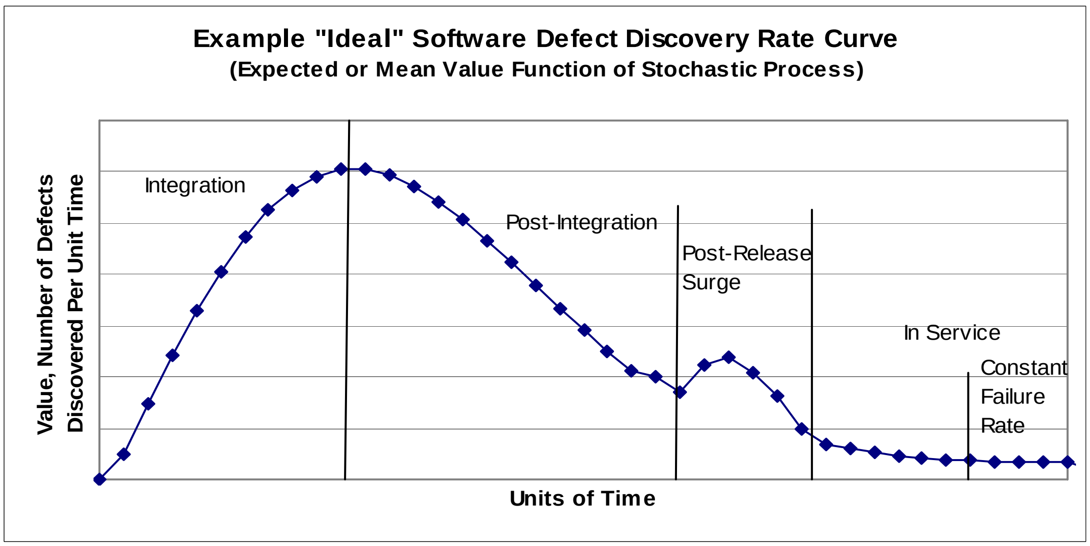
**Defect Data Fitting and Projection Using the STEER II Model**

- **STEER II is the latest version of a tool (currently excel-based) that was originated in the former IBM Federal Systems Division, a predecessor organization of Lockheed Martin IS&S, developed circa 1985.**
- **A subsequent version of the tool was developed at the Software Productivity Consortium, now the SSCI.**
- **STEER II develops fits and projections for phase-based and time-based software defect discovery data.**

## Final View

- **Care should be given to the definitions used for *defect, problem,* etc. when fitting data to models.**
- **Estimates are only as good as the data and the models used to compose them.**

- **The Weibull family of models has been found quite useful in estimating reliability and availability.**
- **Don't wait until testing data is available (from "dynamic' verification stages) to make defect discovery and reliability estimates for your project.**
  - **Initially, make a phase-based estimate using data from inspections and other "static" verification stages.**
  - **When sufficient time-based data is available, update the estimate.**




**Example of STEER II Phase-Based Data Fit/Projection**