Optimizing Inspections Through the Application of Statistical Quality Management Techniques

PSQT/PSTT Conference

Ellen George
Vice President
PS&J Software Six Sigma

Agenda

- Inspection Goals and Measurements
- Closed Loop Inspection Process
- Measuring ROI
- Optimization Strategies
- Defect Prevention
- Results

Inspections Goals and Measurement

- Goals
  - Lower development cost and cycle time
  - Improve delivered product quality
- Strategy
  - Use inspections to remove defects early at a lower overall cost and a reduction in integration and test time
- Measurements of the inspection process are key to achieving the goals
  - "You can't manage what you can't measure"
  - An inspection process that is not actively managed will probably be less effective in achieving its goals. It might even be counterproductive

Types

- Personal reviews
  - Single person bench check
  - Checklist based
  - Performed by author immediately after producing the product

- Team Inspections
  - Checklist based
  - Product review prior to inspection meeting
  - Inspection meeting focus on defect identification
  - Example: Fagin inspections

Measurements

- Only three basic measurements
  - Effort: the effort required to prepare for, hold, and fix the defects found in, the inspection
  - Size: the size of the work product inspected, often measured in lines of code (LOC)
  - Defects: the number and type of defects, effort required to fix, point of injection and point of removal, description

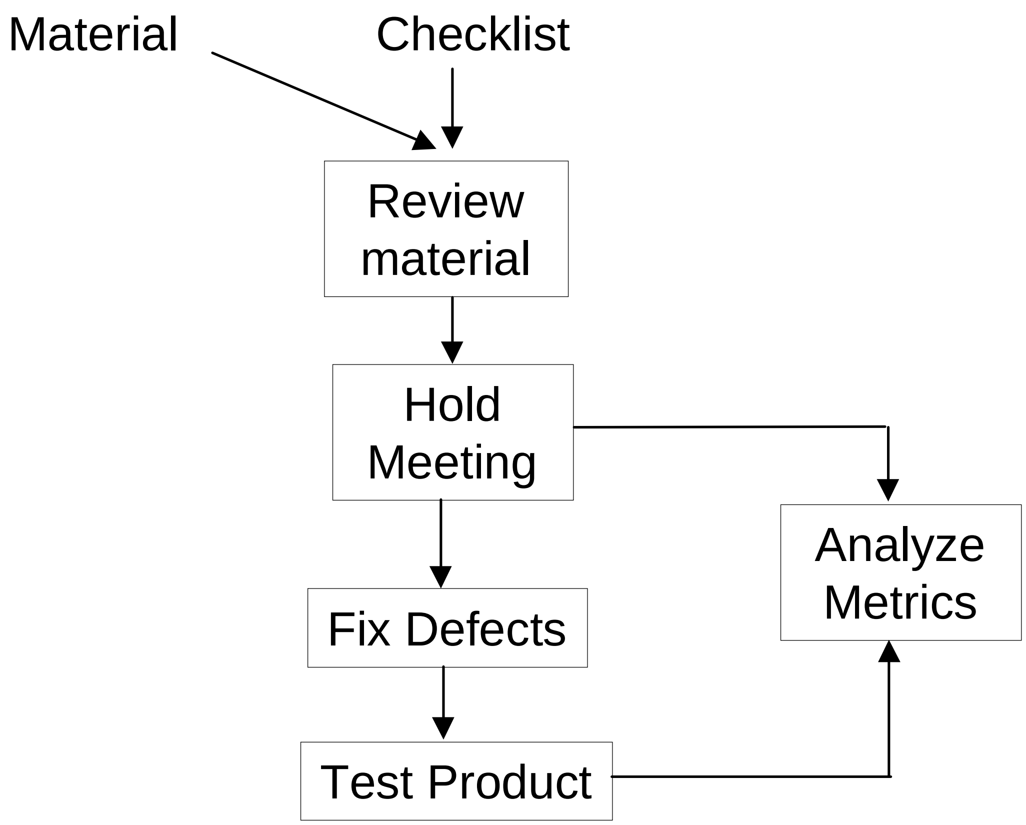
- Simple and economical to collect in-process with an automated tool

- All other metrics are derived from these three measurements

Derived Metrics

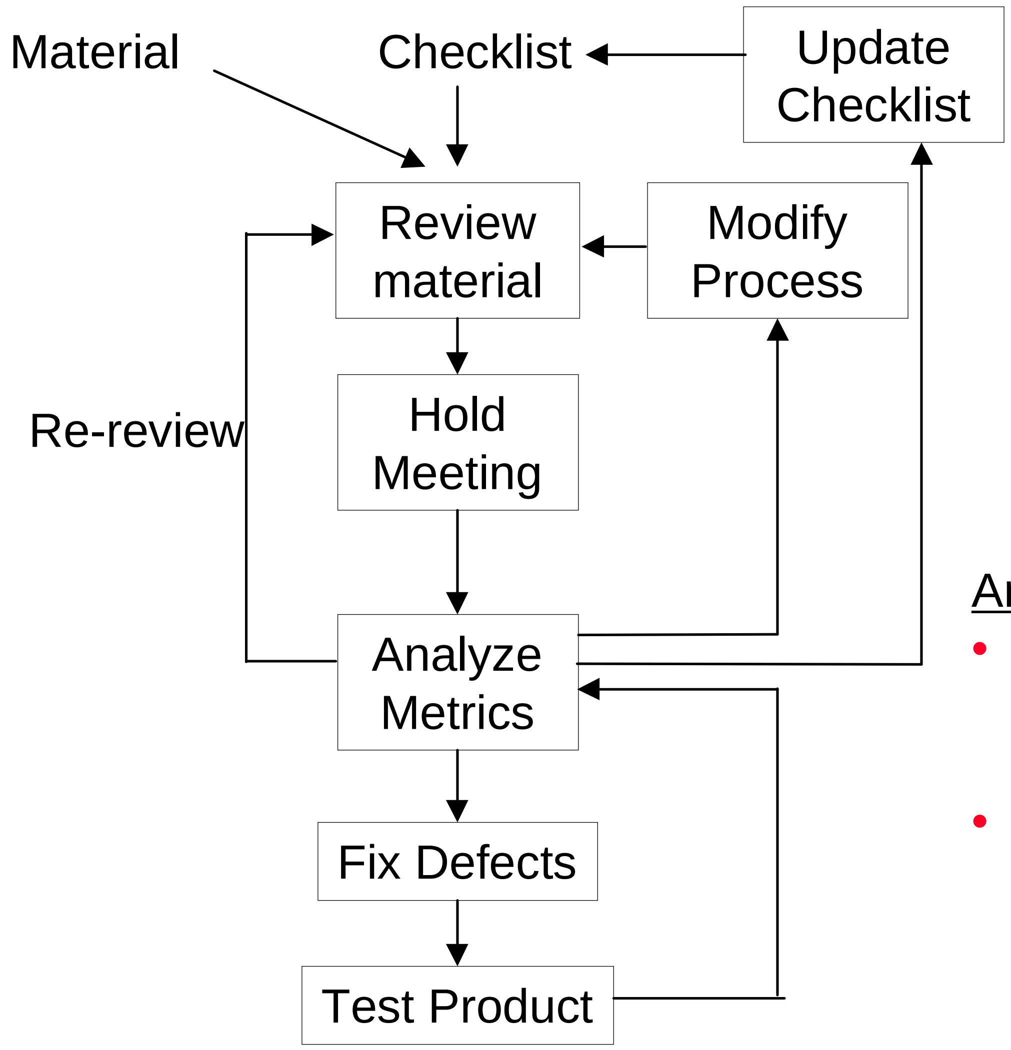
- Review Rate - LOC/hr
- Defect Density - Defects/KLOC
- Defect Removal Rate - Defects/hr
- Yield - Defects Removed/Defects Present
- Defect Removal Leverage - Inspection Removal Rate/Test Removal Rate
- Appraisal Cost of Quality – cost of all inspection activities expressed as a % of project cost
- Failure Cost of Quality – cost of all re-work related activities required to complete compilation and test expressed as a % of project cost

Open Loop Inspection Process - Tracking



- Track process metrics:
  - rate vs yield

Closed Loop Inspection Process - Managing



<u>Update Checklist</u>

- Remove questions that are not catching defects.
- Add questions to catch defects that are leaking out to test.

<u>Modify Process</u>

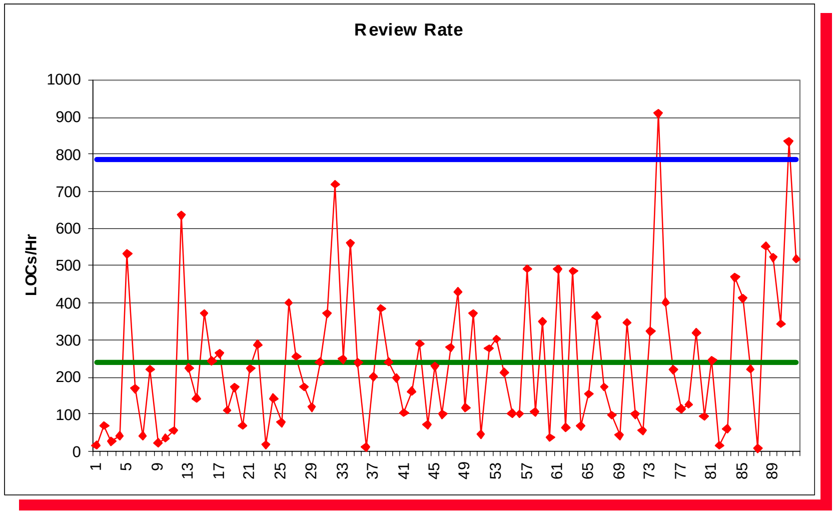
- Modify review rate
- Vary size of material reviewed

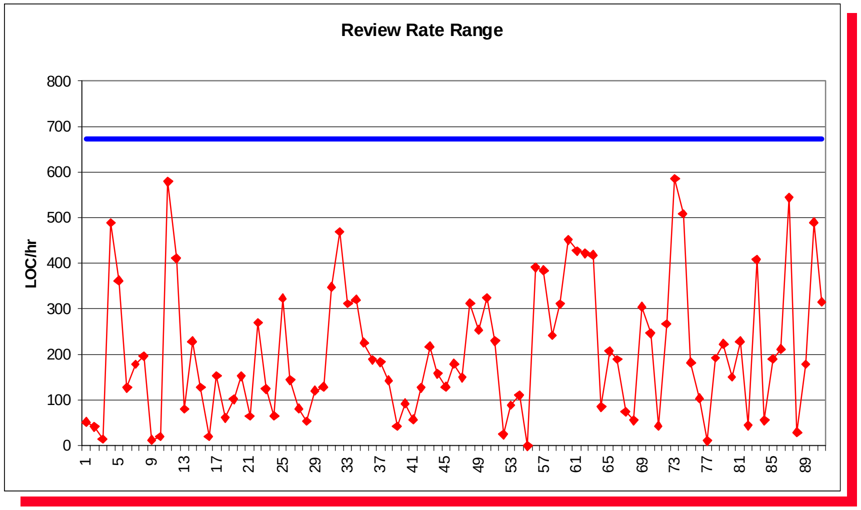
<u>Analyze Metrics</u>

- Process metrics:
  - Rate vs Yield
  - Effectiveness of checklist questions
- Product metrics:
  - Compare yields to quality plan
  - Consider re-review of products that fall outside quality thresholds
  - Buggiest products list

Run Charts

- **Measurements must be put into a temporal context and tested for statistical significance before they can serve as inputs to a decision making process**
- **Individual XBAR-R charts can be used to characterize process**





- **Out of control point on range chart indicates process instability**
- **Out of control point on data chart indicates assignable cause**

Yields and Quality Planning and Management

- Inspection process can be characterized by its yield
- Historical yields allow you to plan the number of defects that will be removed
- Manage to the plan by taking corrective action when actuals diverge from plan

	Defects leaked from prev phase	New Defects Injected	Phase Yield	Defects Contained	Defects Leaked	Defect Removal Cost	Total Removal Cost (hrs)
Design	0.0	40	0%	0.0	40.0	n/a	0.00
Design Bench Check	40.0	0	0%	0.0	40.0	10 mins	0.00
Design Inspection	40.0	0	0%	0.0	40.0	30 mins	0.00
Code	40.0	60	0%	0.0	100.0	n/a	0.00
Code Bench Check	100.0	0	0%	0.0	100.0	5 mins	0.00
Compile	100.0	0	50%	50.0	50.0	1 min	0.83
Code Inspection	50.0	0	0%	0.0	50.0	15 mins	0.00
Unit Test	50.0	0	50%	25.0	25.0	15 mins	6.25
Integration Test	25.0	0	35%	8.8	16.3	18 hrs	157.50
System Test	16.3	0	35%	5.7	10.6	18 hrs	102.38
CUSTOMER	10.6						267

Quality Plan for 1 KLOC Embedded Code

A Control System Viewpoint

- **The outputs of a process, y, are usually a function, f, of a set of control variables, x, and include a process noise component ε:**

$$y = f(x) + \varepsilon$$

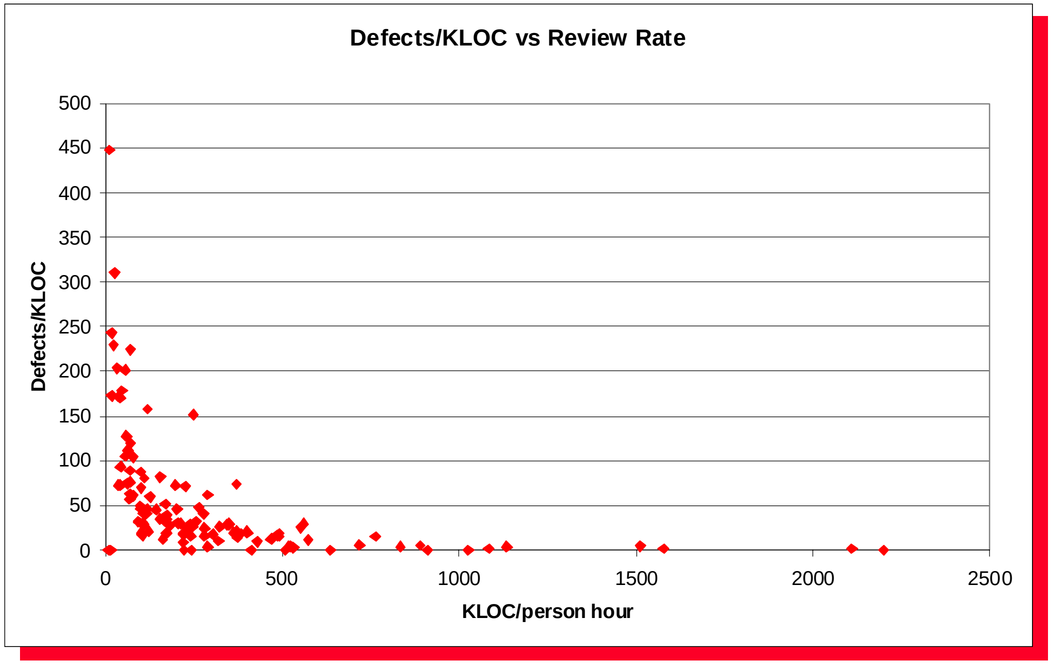
  - **The y's are not directly controllable, but they can be controlled by the directly controllable x's.**
  - **Statistical measurements are necessary to avoid re-acting to the noise ε**

- **Ideally we would like software inspection process that acts like a responsive, "closed loop" control system driving the x's to planned values and through their relationship to the y's, achieving overall product goals**

Our experience has shown that review rate is the x that drives the inspection yield

Review Rate as a Control Variable

- Yields are useful for planning, but the down side is that they are lagging indicators
- Not good for process control
- Need correlated predictor – review rate
- Reading rate: 1000 - 3500 LOC/hr



- Optimal review rate: around 200 LOC/hr
- Without guidelines and training in the proper use of checklists, engineers reviewed considerably faster than the optimal review rate
- Requires active management by the review leader

Calculating Return on Investment - 1

- **Costs can be directly measured**
  - **training, tools, performing the inspections**
- **The dominant costs are the inspection prep and the meeting time**
- **Savings require estimating the difference in cost between finding a defect in review and finding it later in the process**

	Defects leaked from prev phase	New Defects Injected	Phase Yield	Defects Contained	Defects Leaked	Defect Removal Cost	Total Removal Cost (hrs)
Design	0.0	40	0%	0.0	40.0	n/a	0.00
Design Bench Check	40.0	0	0%	0.0	40.0	10 mins	0.00
Design Inspection	40.0	0	0%	0.0	40.0	30 mins	0.00
Code	40.0	60	0%	0.0	100.0	n/a	0.00
Code Bench Check	100.0	0	0%	0.0	100.0	5 mins	0.00
Compile	100.0	0	50%	50.0	50.0	1 min	0.83
Code Inspection	50.0	0	0%	0.0	50.0	15 mins	0.00
Unit Test	50.0	0	50%	25.0	25.0	15 mins	6.25
Integration Test	25.0	0	35%	8.8	16.3	18 hrs	157
System Test	16.3	0	35%	5.7	10.6	18 hrs	102
CUSTOMER	10.6						267

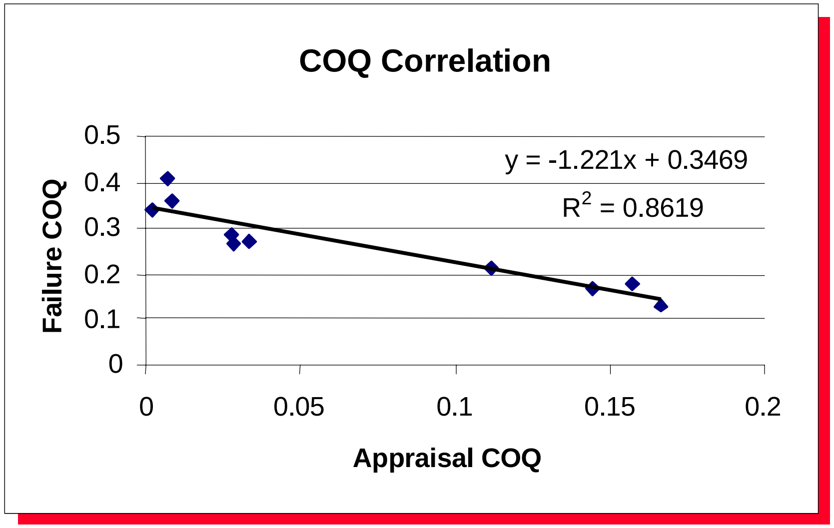
- **Without inspections, the cost of defect removal is 267 hrs per KLOC**

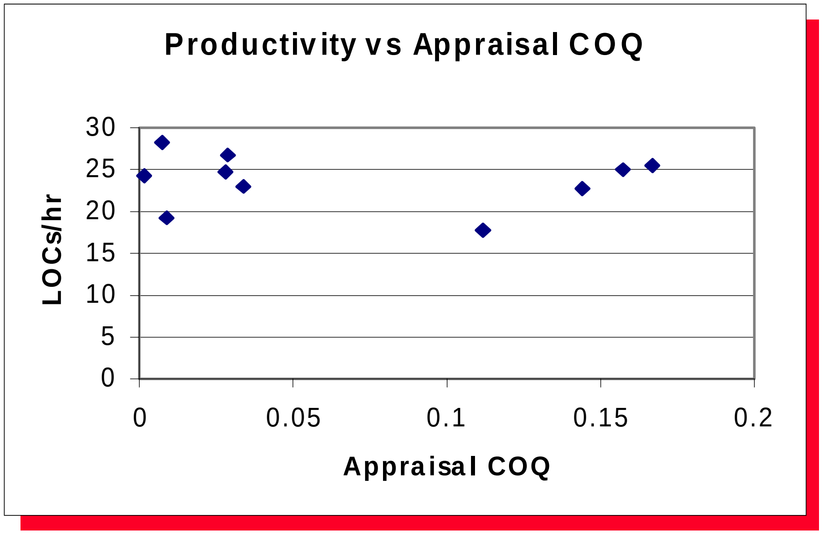
Calculating Return on Investment - 2

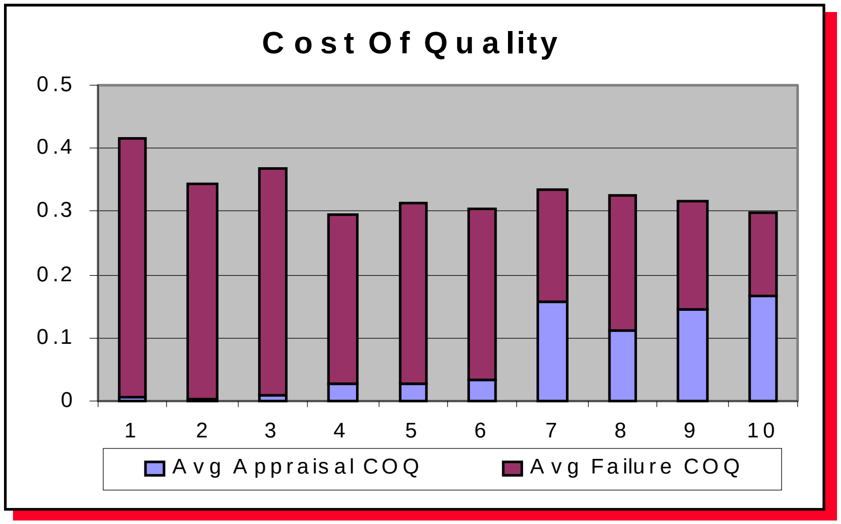
	Defects leaked from prev phase	New Defects Injected	Phase Yield	Defects Contained	Defects Leaked	Defect Removal Cost	Total Removal Cost (hrs)
Design	0.0	40	0%	0.0	40.0	n/a	0.00
Design Bench Check	40.0	0	50%	20.0	20.0	10 mins	3.33
Design Inspection	20.0	0	50%	10.0	10.0	30 mins	5.00
Code	10.0	60	0%	0.0	70.0	n/a	0.00
Code Bench Check	70.0	0	70%	49.0	21.0	5 mins	4.08
Compile	21.0	0	50%	10.5	10.5	1 min	0.18
Code Inspection	10.5	0	60%	6.3	4.2	15 mins	1.58
Unit Test	4.2	0	50%	2.1	2.1	15 mins	0.53
Integration Test	2.1	0	35%	0.7	1.4	18 hrs	13.23
System Test	1.4	0	35%	0.5	0.9	18 hrs	8.60
CUSTOMER	0.9						37

- **With inspections, the cost of defect removal drops to 37 hours, a savings of 230 = 267 – 37 hours**
- **The cost of holding the inspections is about 40 hours (at 200 LOC/hr), so the net savings is 190 hours**

Quality is Free







- As appraisal cost increases
  - Failure costs decrease
  - Overall COQ remains constant
  - Productivity remains constant
- No net cost to performing appraisals
- Appraisal cost is more controllable than failure cost
- Results in more accurate estimates, fewer defects to integration and system test

Optimization Strategy

- Personal reviews performed prior to team inspections
  - Remove all the errors the author can detect at the lowest possible inspection cost
  - Checklist derived from author's own list of compilation and test defects flags high risk areas where author has a history of making mistakes
- Frequent short team inspections
  - Checklists focus on interface and requirements related issues that can't easily be found in the personal review
  - Small teams that include the internal "customers" for the product
  - Focus on a few hundred lines of code at a time
- Periodic Defect Prevention meetings provided the development team with an opportunity to review their data and define approaches to detect defects earlier or prevent or prevent them entirely
- Defect prone products "pulled" from integration and test and re-inspected

Goal: Minimize review cost while maximizing yield

Optimization Strategy Advantages

- **Doesn’t waste team’s time with defects the author can easily find**
- **By inspecting a few hundred lines at a time, preparation time required is on the order of an hour**
- **Reviewers can stay focused and inspection can be held on the same day that product is available**
- **Eliminates lags, removes the temptation for the author to move forward into test before the review takes place**
- **Entire cycle can take as little as 2 – 3 hours from product availability to end of inspection**
- **Developers use their own data for defect prevention**
  - **Eliminates handoffs**

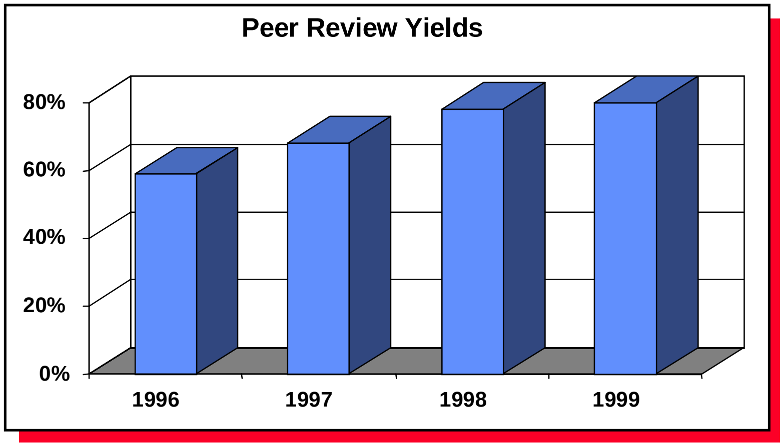
Defect Prevention

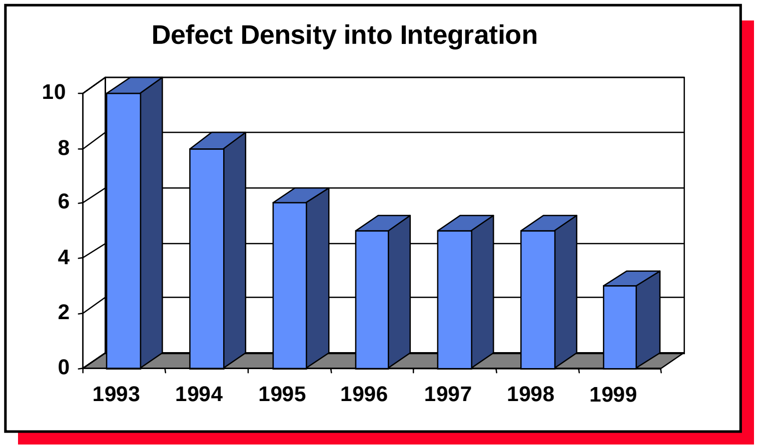
- Defect Prevention can be implemented by an organization that is performing inspections and collecting defect data.
- A Defect Prevention team sets and manages to their own goal.
- They use their own defect data, captured during inspections.
- Defects are analyzed using Pareto charts to identify most expensive, most frequent, etc.
- Actions are taken to prevent a targeted defect type from occurring in the future.
  - Modify checklists, change coding and design standards
- The team members convince themselves of the value of the activity by calculating their own ROI.
- Lessons Learned are shared with other Defect Prevention teams on a periodic basis.

Data must be regularly used by the people collecting it, otherwise they will stop collecting it!

Results

- Over a period of 5 years, we gradually implemented the strategies described
- As Peer Review yields increased from 60% to 80% and we introduced personal reviews, defects into integration were reduced from 10/KLOC to 3/KLOC
- At the same time, cost of performing peer reviews decreased by 40% as we reduced the size of the inspection teams





The organization realized a net improvement of 190 hrs / KLOC!

References

- Ellen George: Honeywell Presentation at the SEI Symposium in Washington, DC, September 2000 - ***"Honeywell PSP Deployment Strategy"***
- Steve Janiszewski: Honeywell Presentation at the SEPG 2000 Conference in Seattle, WA, March 2000 - ***"PSP, TSP and Six Sigma"***
- Steve Janiszewski: Honeywell Presentation at the NJ Spin Meeting, Piscataway, New Jersey, March 2001 - ***"Introduction to PSP & TSP"***

www.SoftwareSixSigma.com

Contact Information

Ellen George

(201) 358-8828

EllenGeorge@SoftwareSixSigma.com

Steve Janiszewski

(201) 947-0150

SteveJaniszewski@SoftwareSixSigma.com