# *QUANTITATIVE PROCESS MANAGEMENT AND SOFTWARE QUALITY MANAGEMENT*

*Ray Madachy*

*USC Computer Science*

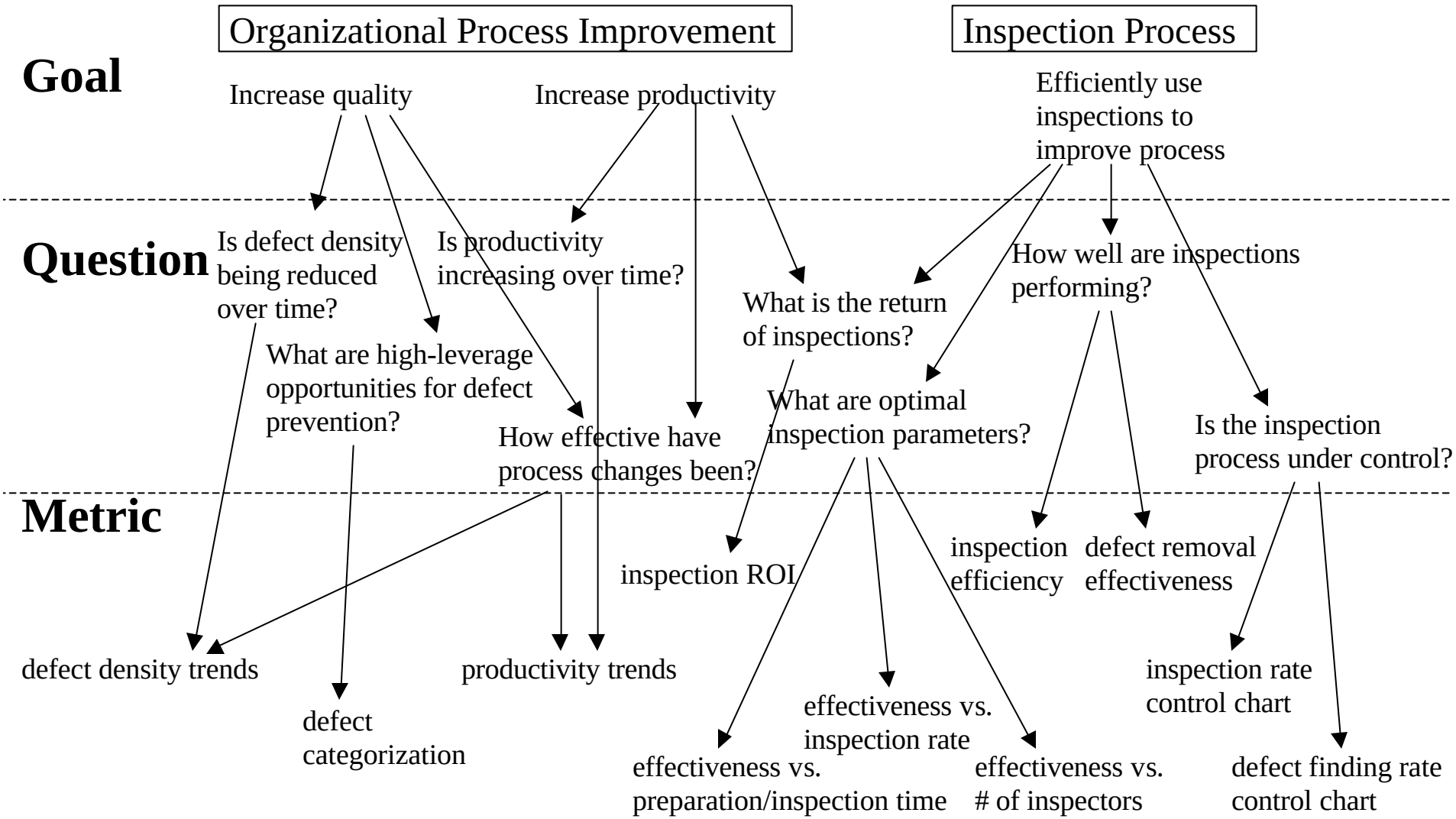
# *Outline*

- *Metrics Refreshers*
- *KPA Overviews*
- *Project Procedures*
- *Sample Metrics Set*
- *Analysis Overview*
- *Litton Implementation*
- *Summary*

# *Goal-Question-Metric Paradigm*

- *Goal-Question-Metric (GQM) is a framework for developing a metrics program*
- *Steps:*
  - *generate a set of organizational goals*
    - *what do you want to improve?*
  - *derive a set of questions relating to the goals*
    - *answers provide visibility into meeting the goals*
  - *develop a set of metrics needed to answer the questions*
- *See EP-07 and web for further information*

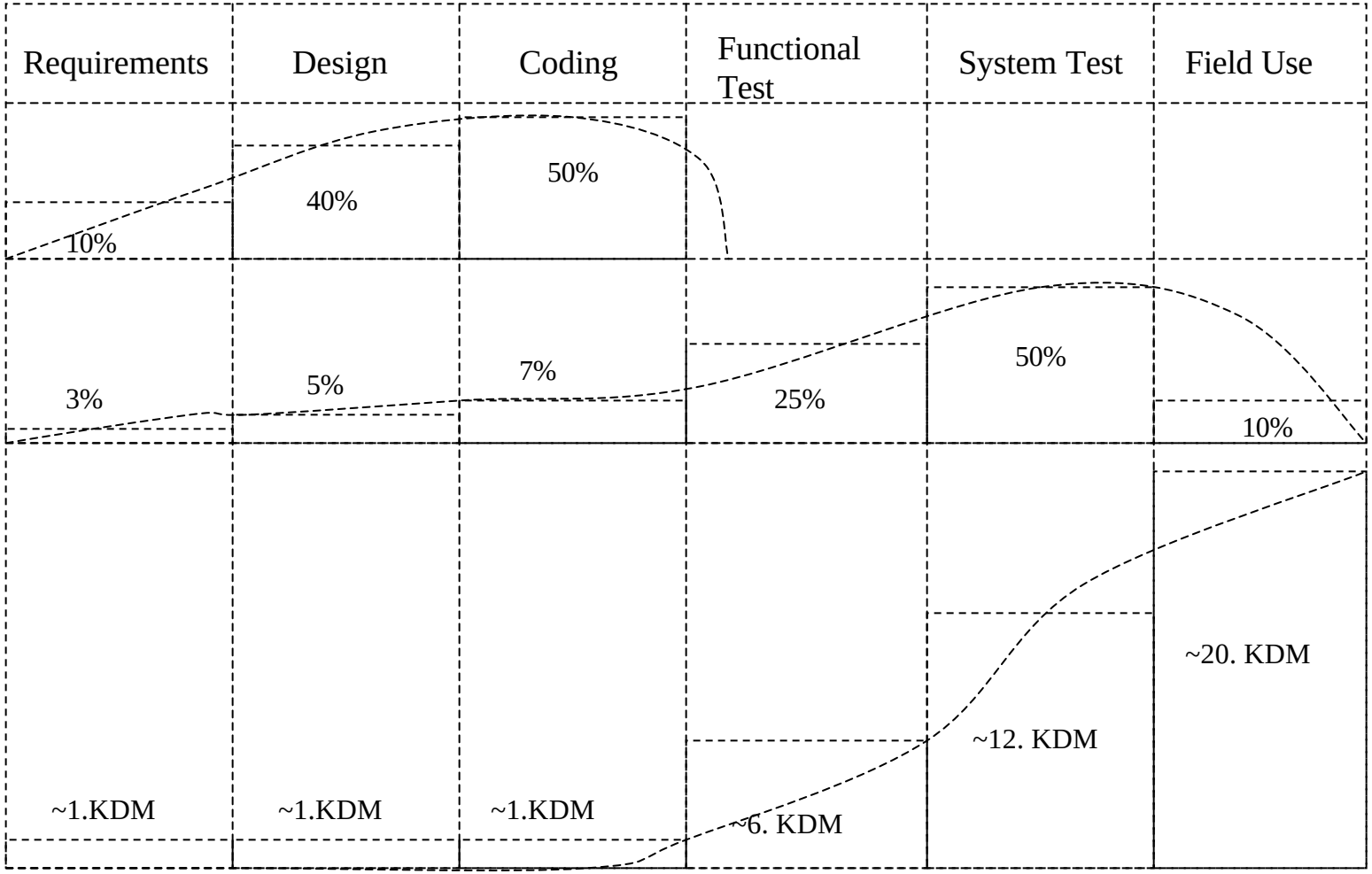
# *Sample GQM Roadmap*



# *Defect Analysis*

- *Defect: any flaw in the specification, design, or implementation of a product.*
- *Facilitate process improvement through defect analysis*
  - *defect categorization to identify where work must be done and to predict future defects*
  - *causal analysis to prevent problems from reoccurring*

# *Fault Distributions*




KDM=kilo deutsch marks

# *Fault Distributions (cont.)*

| **Process Maturity Level** | Requirements | Design | Coding | Functional Test | System Test | Field Use | Phase |
|---|---|---|---|---|---|---|---|
| | 10% | 40% | 50% | | | | **Fault Introduction Distribution** |
| 5 | 5% | 20% | 40% | 20% | 10% | <5% | |
| 4 | 3% | 12% | 30% | 30% | 20% | 5% | |
| 3 | 0% | 2% | 20% | 38% | 32% | 8% | **Fault Detection Distribution** |
| 2 | 0% | 0% | 3% | 30% | 50% | 17% | |
| 1 | 0% | 0% | 2% | 15% | 50% | 33% | |
| | 1 | 1 | 1 | 6 | 12 | 20 | **Relative Fault Cost** |

# *Defect Flow Tracking*

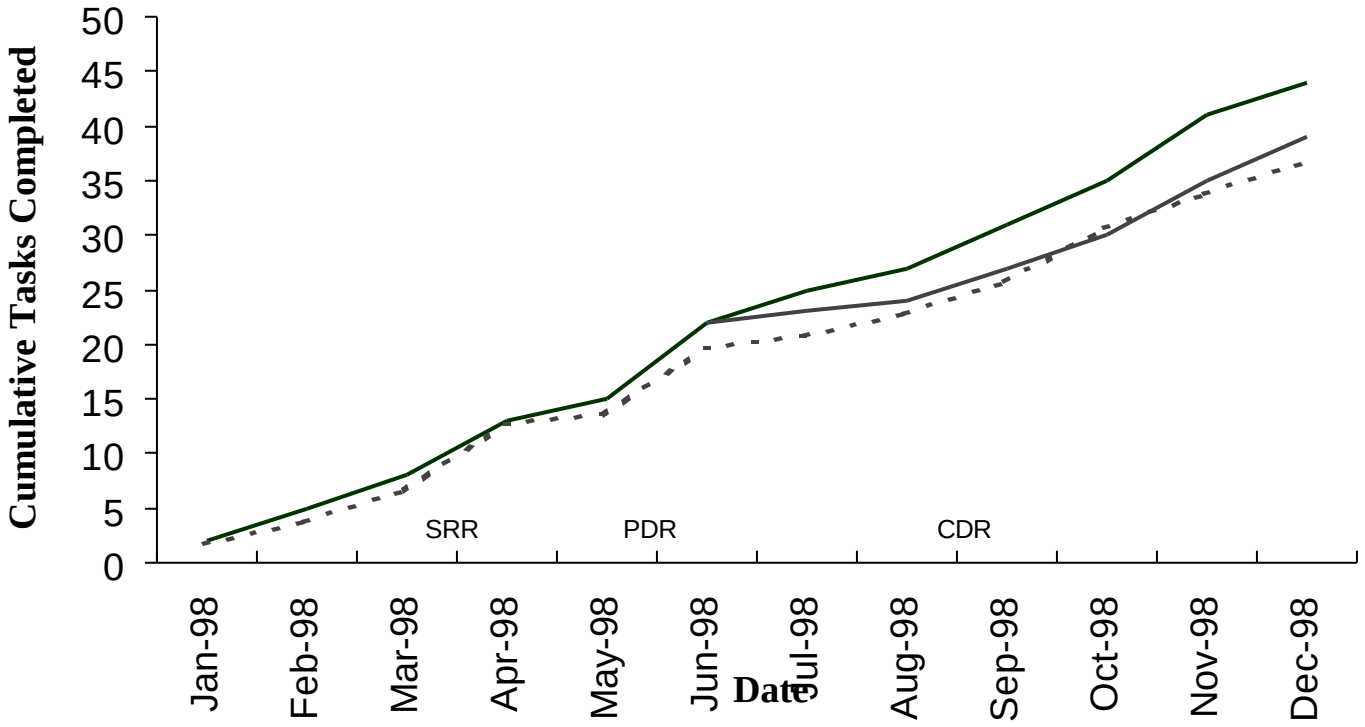
- *A defect introduction and removal matrix can be generated and used as a basis for defect analysis and prevention.*

| | **Percentage of Defects** | | | | |
|---|---|---|---|---|---|
| | **Phase injected** | | | | |
| **Phase detected** | Requirements | Preliminary design | Detailed design | Code/unit test | Total |
| Requirements | 37% | | | | 8% |
| Preliminary design | 22% | 38% | | | 16% |
| Detailed design | 15% | 18% | 34% | | 17% |
| Code/unit test | 7% | 24% | 28% | 43% | 25% |
| Integration testing | 7% | 9% | 14% | 29% | 14% |
| System testing | 11% | 12% | 24% | 29% | 19% |
| Total | 100% | 100% | 100% | 100% | 100% |

# *Project Tracking Indicators*

- *The following are standard indicators used for tracking a project. Many of them serve as as a basis for Level 4 metrics.*

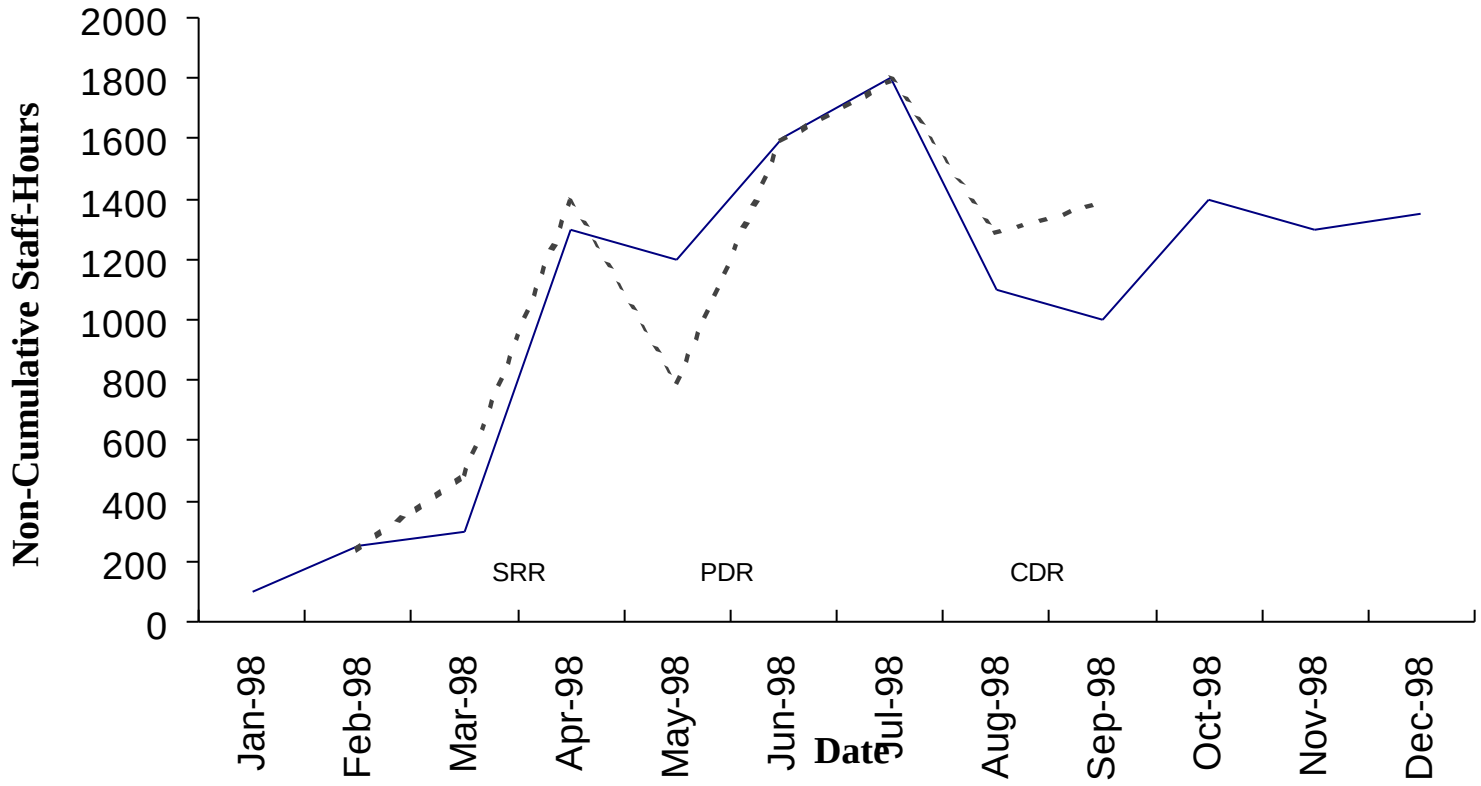
# *Progress*




| | Jan-98 | Feb-98 | Mar-98 | Apr-98 | May-98 | Jun-98 | Jul-98 | Aug-98 | Sep-98 | Oct-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Initial Plan | 2 | 5 | 8 | 13 | 15 | 22 | 25 | 27 | 31 | 35 | 41 | 44 |
| Replan | | | | | | 22 | 23 | 24 | 27 | 30 | 35 | 39 |
| Actual | 2 | 4 | 7 | 13 | 14 | 20 | 21 | 23 | 26 | 31 | 34 | 37 |

**CS577b 2/16/00**

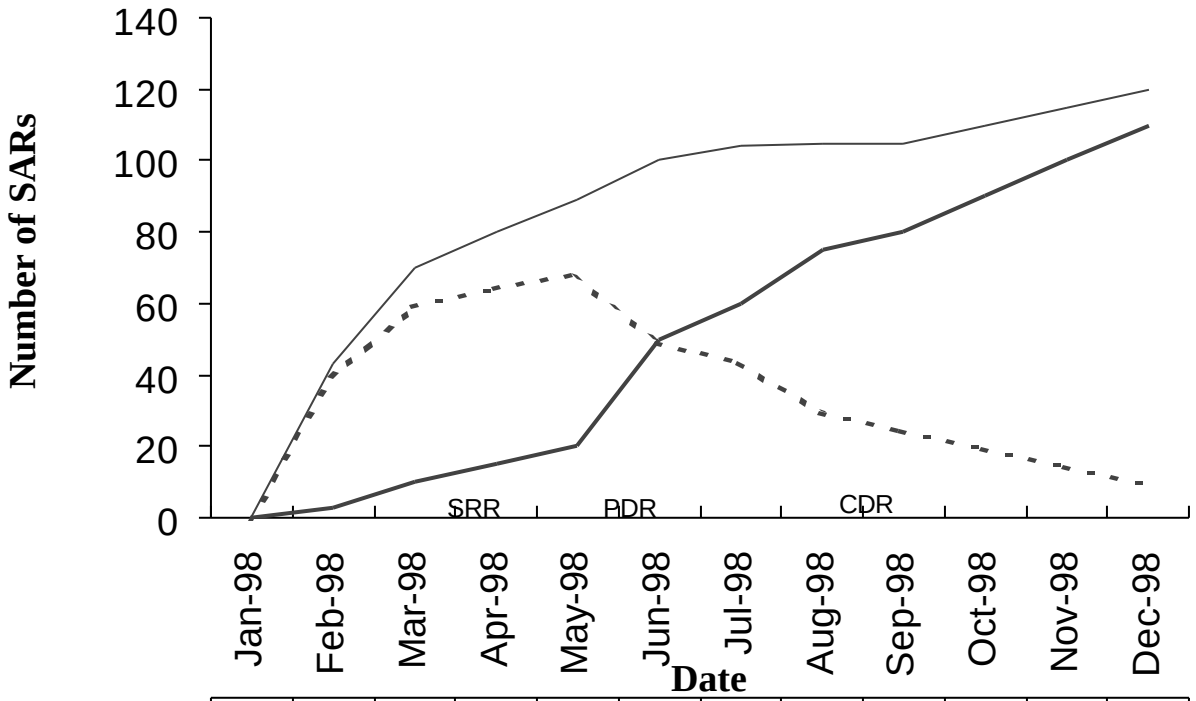
# *Effort*




| | Jan-98 | Feb-98 | Mar-98 | Apr-98 | May-98 | Jun-98 | Jul-98 | Aug-98 | Sep-98 | Oct-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Planned | 100 | 250 | 300 | 1300 | 1200 | 1600 | 1800 | 1100 | 1000 | 1400 | 1300 | 1350 |
| Actual | | 250 | 500 | 1400 | 800 | 1600 | 1800 | 1300 | 1400 | | | |

**CS577b 2/16/00**

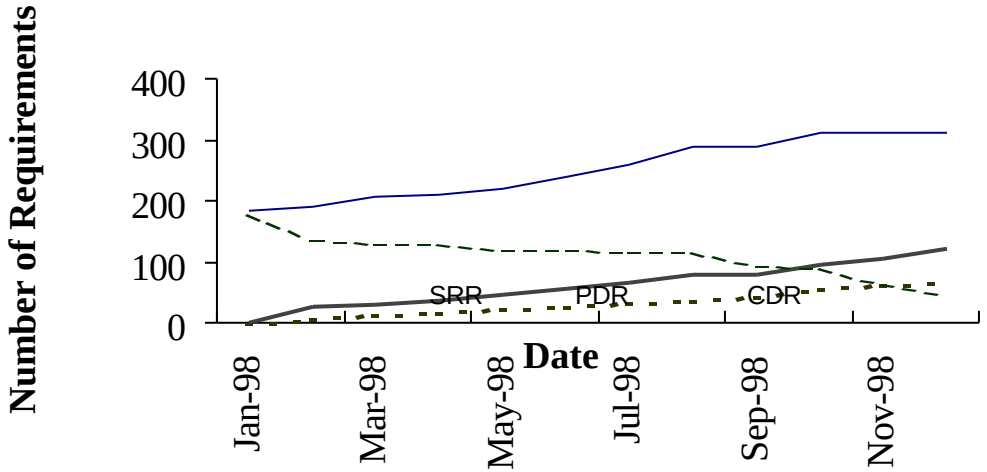
# *Software Action Request*




| | 1-Jan-98 | 1-Feb-98 | 1-Mar-98 | 1-Apr-98 | 1-May-98 | 1-Jun-98 | 1-Jul-98 | 1-Aug-98 | 1-Sep-98 | 1-Oct-98 | 1-Nov-98 | 1-Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Total SARs | 0 | 43 | 70 | 80 | 89 | 100 | 104 | 105 | 105 | 110 | 115 | 120 |
| Total SARs Closed | 0 | 3 | 10 | 15 | 20 | 50 | 60 | 75 | 80 | 90 | 100 | 110 |
| SARs Currently Open | 0 | 40 | 60 | 65 | 69 | 50 | 44 | 30 | 25 | 20 | 15 | 10 |

**CS577b 2/16/00**

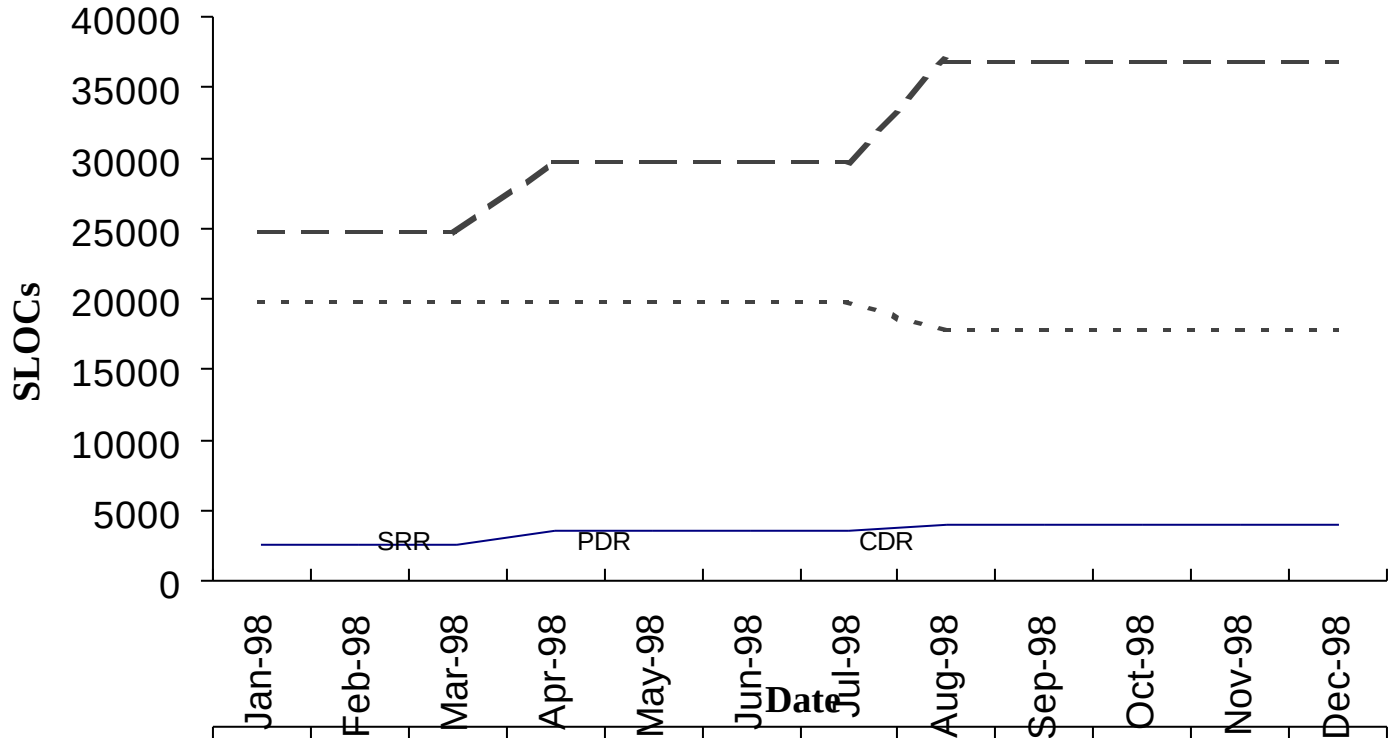
# *Requirements Stability*




| | Jan-98 | Mar-98 | Apr-98 | Jun-98 | Aug-98 | Oct-98 | Nov-98 |
|---|---|---|---|---|---|---|---|
| Total Requirements | 185 | 205 | 210 | 240 | 290 | 310 | 310 |
| Cum. Customer Changes | 0 | 15 | 20 | 30 | 40 | 60 | 65 |
| Cum. Internal Changes | 0 | 30 | 37 | 55 | 80 | 95 | 105 |
| TBDs | 185 | 135 | 133 | 125 | 120 | 95 | 70 |

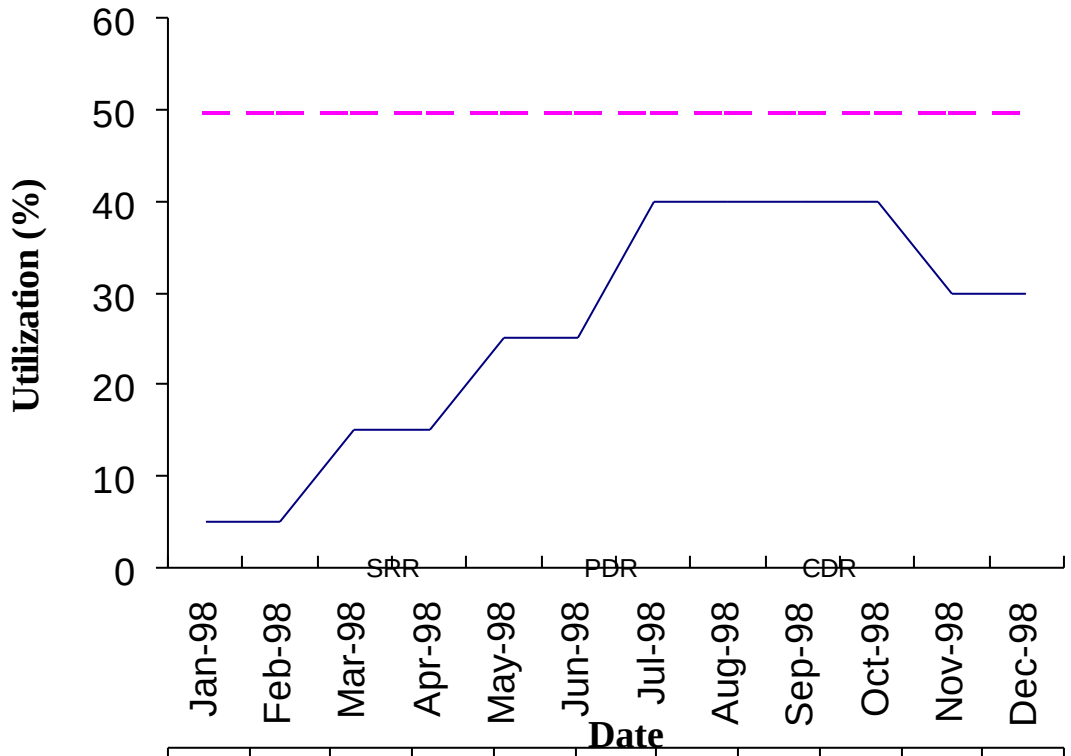
**CS577b 2/16/00**

# *Size Stability*



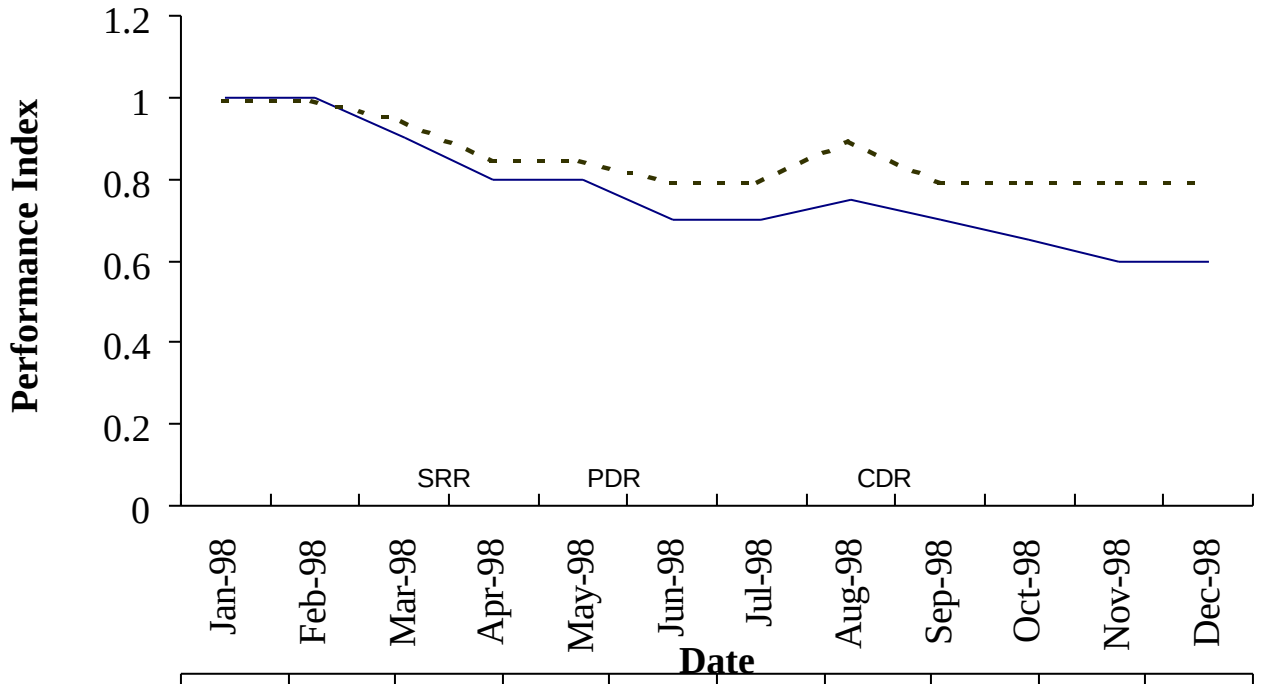

| | Jan-98 | Feb-98 | Mar-98 | Apr-98 | May-98 | Jun-98 | Jul-98 | Aug-98 | Sep-98 | Oct-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Final Modified | 2550 | 2550 | 2550 | 3530 | 3530 | 3530 | 3530 | 4000 | 4000 | 4000 | 4000 | 4000 |
| Final Reuse | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 18000 | 18000 | 18000 | 18000 | 18000 |
| Final New | 25000 | 25000 | 25000 | 30000 | 30000 | 30000 | 30000 | 37000 | 37000 | 37000 | 37000 | 37000 |

# *Computer Resource Utilization*




| | Jan-98 | Feb-98 | Mar-98 | Apr-98 | May-98 | Jun-98 | Aug-98 | Sep-98 | Oct-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CPU utilization | 5 | 5 | 15 | 15 | 25 | 25 | 40 | 40 | 40 | 30 | 30 |
| Maximum CPU Allowed | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 50 |

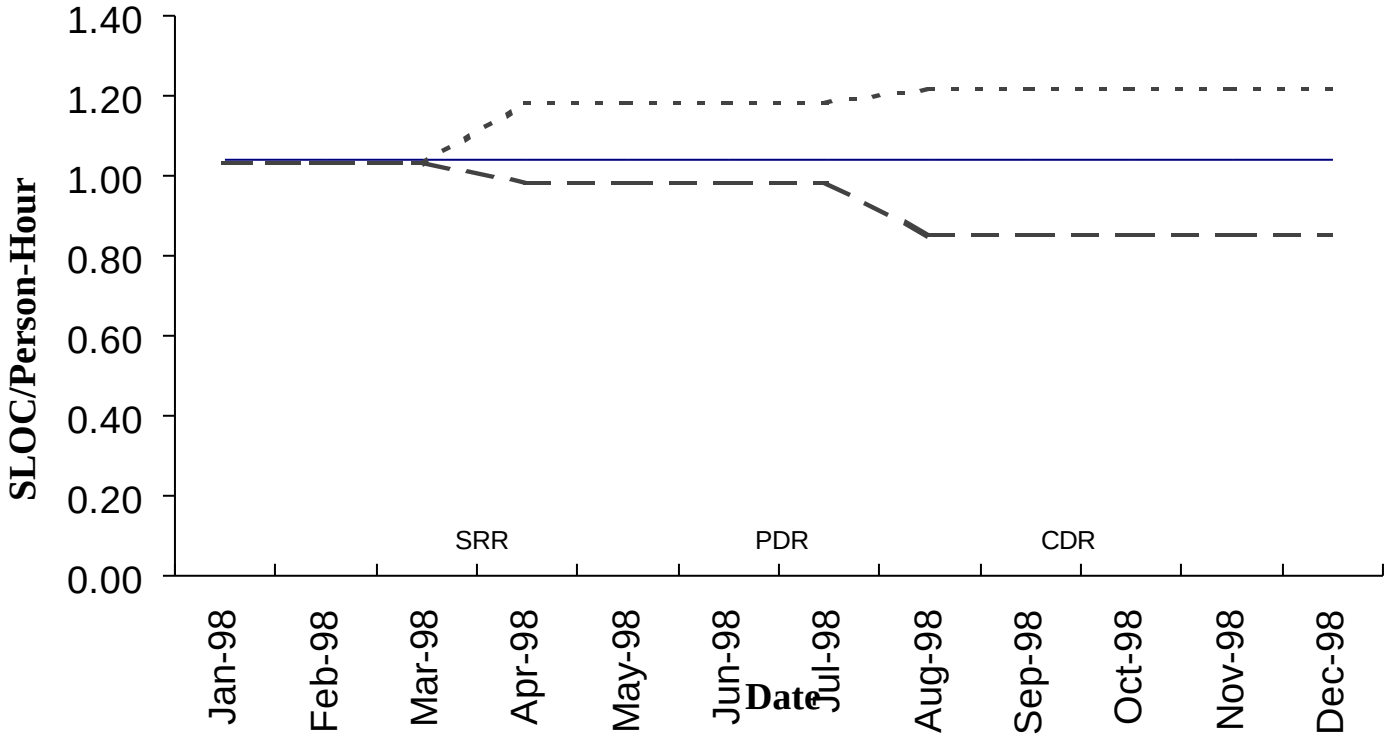
# *Cost & Schedule Performance*




| | Jan-98 | Feb-98 | Mar-98 | May-98 | Jun-98 | Jul-98 | Aug-98 | Sep-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|
| SPI (BCWP/BCWS) | 1 | 1 | 0.9 | 0.8 | 0.7 | 0.7 | 0.75 | 0.7 | 0.6 | 0.6 |
| CPI (BCWP/ACWP) | 1 | 1 | 0.95 | 0.85 | 0.8 | 0.8 | 0.9 | 0.8 | 0.8 | 0.8 |

**CS577b 2/16/00**

# *Estimated Productivity*




| | Jan-98 | Feb-98 | Mar-98 | Apr-98 | May-98 | Jun-98 | Jul-98 | Aug-98 | Sep-98 | Oct-98 | Nov-98 | Dec-98 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bid Prod | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 | 1.04 |
| Est Prod | 1.04 | 1.04 | 1.04 | 1.19 | 1.19 | 1.19 | 1.19 | 1.22 | 1.22 | 1.22 | 1.22 | 1.22 |
| Should Bid | 1.04 | 1.04 | 1.04 | 0.99 | 0.99 | 0.99 | 0.99 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 |

**CS577b 2/16/00**

# *CMM Level 4 Introduction*

- *In CMM Level 4, software process and product quality are quantitatively understood and controlled via collection of detailed measurements.*
- *In QPM, projects use quantitative techniques to take process measurements, analyze their software process, identify special causes of variations in the performance of the process, control the performance of the process within well-defined limits, and report their results.*
- *In SQM, projects take measurement and analyze product quality, identify special causes of variations, control the quality within well-defined limits, and report their results.*

# *QPM Goals*

- *The quantitative process management activities are planned.*
- *The process performance of the project's defined software process is controlled quantitatively.*
- *The process capability of the organization's standard software process is known in quantitative terms.*

# *SQM Goals*

- *The project's software quality management activities are planned.*
- *Measurable goals for software product quality and their priorities are defined.*
- *Actual progress toward achieving the quality goals for the software products is quantified and managed.*

# *Quality in the CMM*

- *At level 2, quality is "conformance to requirements"*
- *In levels 3 and 4, the emphasis moves to better understanding the needs of:*
  - *customer*
  - *end users*
  - *developing organization*
- *Customer determines what quality is*
  - *customer satisfaction becomes a goal*

# *Additional CMM Considerations*

- *QPM/SQM should not be considered in isolation, since other KPAs require measurements.*
- *The recommended metrics also consider the goals and activities of Level 5 KPAs:*
  - *Defect Prevention*
  - *Technology Change Management*
  - *Process Change Management.*

# *Key Concepts and Sample Metrics*

- *QPM is controlling the process performance; keeping it stable within acceptable limits*
  - *productivity*
  - *quality*
  - *cycle time*
  - *defect removal effectiveness*
- *SQM is identifying product quality goals and quantitatively controlling the quality*
  - *defect density*
  - *computer resource utilization*
  - *product functionality*
  - *reliability/maintainability*
  - *all traceable to customer requirements*

# *Quantitative Process Management (QPM) ETVX Diagram*

The purpose of QPM is to control the process performance of the software project quantitatively. Software process performance represents the actual results achieved following a software process.

**ENTRY**

1. Policy for measuring and quantitatively controlling performance of project's defined process (C1)
2. Policy for analyzing the process capability of the organization's standard software process (OSSP) (C2)
3. Group exists for coordinating QPM (Ab1)
4. Adequate resources/funding (Ab2)
5. Support for collecting, recording, and analyzing data (Ab3)
6. Training for individuals implementing or supporting QPM activities (Ab4)
7. Training for participants (Ab5)
8. Procedures for Ac1, Ac4, Ac5, & Ac7

**TASK**

1. Develop Project's QPM plan (Ac1)
2. Perform the QPM plan (Ac2)
3. Determine a strategy for data collection and quantitative analysis based on project's process (Ac3)
4. Collect measurement data to control project's process (Ac4)
5. Analyze the project's process to bring under quantitative control (Ac5)
6. Prepare and distribute reports of QPM results (Ac6)
7. Establish and maintain the process capability baseline for the OSSP (Ac7)

**VERIFICATION**

1. Reviews with senior management (V1)
2. Reviews with project manager (V2)
3. Reviews/audits by SQA (V3)
4. Measurement of status of QPM activities (M1)

**EXIT**

1. QPM activities are planned (G1)
2. Project's process performance is controlled quantitatively (G2)
3. Process capability of OSSP is known quantitatively (G3)

# *Software Quality Management (SQM) ETVX Diagram*

The purpose of SQM is to develop a quantitative understanding of the quality of the project's software products and achieve specific quality goals

**ENTRY**

1. Policy for managing project's defined quality (C1)
2. Adequate resources/funding (Ab1)
3. Training for individuals implementing or supporting SQM activities (Ab2)
4. Training for participants (Ab3)
5. Procedures for Ac1

**TASK**

1. Develop Project's SQM plan (Ac1)
2. Perform the activities in the SQM plan (Ac2)
3. Project's quality goals are defined, monitored, and revised throughout the life cycle (Ac3)
4. Project's actual quality is measured, analyzed, and compared to the goals on an event driven basis (Ac4)
5. Quality goals are allocated appropriately to sub contractors (Ac5)

**VERIFICATION**

1. Reviews with senior management (V1)
2. Reviews with project manager (V2)
3. Reviews/audits by SQA (V3)
4. Measurement of status of SQM activities (M1)

**EXIT**

1. SQM activities are planned (G1)
2. Measurable quality goals and their priorities are defined (G2)
3. Progress towards quality goals is quantified and managed (G3)

# *Common Threads*

- *Similar themes in both QPM and SQM:*
  - *plans are required*
  - *plans must be performed*
  - *goals must be set and tracked*
  - *data must be analyzed and corrective actions taken*
  - *institutionalization requirements*
- *Key difference is process or product view*

# *Level 4 Based on Previous KPAs*

- *SPP: provides a basis for plans and estimates*
- *SPTO: provides a basis for reviewing the process activities; basis for tracking and revising the quality goals*
- *SSM: goals are allocated to subcontractors, as appropriate*
- *SCM: data analysis should be managed and controlled*
- *SQA: reviews/audits performed; participates with project*
- *OPF: SEPG coordinates data activities across organization*
- *OPD: data and analysis is included in the data repository; organization processes and procedures may be changed as result of analysis*
- *TP: required training/orientation is provided*
- *ISM: project processes and procedures may be changed as a result of analysis*
- *SPE: defect data results from testing and peer reviews; other data results also available from SPE activities*
- *IC: the customer's voice is made clearer*
- *PR: provides data from peer reviews*

# *Level 4 Relationship to Level 5 KPAs*

- *Data analysis from Level 4 activities enables focusing the performance of Defect Prevention (DP), Technology Change Management (TCM), and Process Change Management (PCM)*

# *Participants / Responsibilities*

- *SEPG*
  - *analysis of process and product data*
  - *establishment of baselines*
  - *QPM/SQM tool support*
- *Senior and project management*
  - *setting of quantitative process and product goals*
  - *review and oversight*
- *Projects*
  - *creation of QPM and SQM plans*
  - *implementation of those plans*
- *Software Quality Assurance (SQA)*
  - *auditing of QPM/SQM activities*

# *Project Procedures - Planning*

- *Projects create process and product plans adjunct with Software Development Plans.*
- *Must include the following:*
  - *identify the activities to be performed*
  - *identify process performance and product quality goals*
  - *define process performance and product quality thresholds*
  - *describe methods to assess and track process and quality goals*
  - *describe corrective actions to take when indicators cross upper and lower limits (goals are jeopardized)*
  - *describe procedures for adjusting the goals/thresholds and replanning*
  - *describe schedule of activities*

# *Project Procedures - Development*

- *Project personnel must implement plans.*
  - *analyze process/product data against goals*
  - *determine causes of variation*
  - *implement corrective action*
  - *report the results.*
- *Reviews are held with senior management and project management.*
  - *determine whether goals are being met*
  - *assign actions when necessary*
  - *revise plans as necessary*

# *Project Procedures - Reviews*

- *Project representatives and senior management hold periodic QPPM status review meetings and document the results of the activities.*
- *Meetings provide a forum for presenting and analyzing the process performance on the project.*
  - *If the process is out of limits, then the causes for the variation are determined and appropriate courses of action are taken.*
  - *Corrective actions are tracked and statused.*
- *SQA reviews/audits the activities and work products.*

# *QPM Metrics Set*

- *Projects use standard metric indicators in conjunction with established indicator limits for their quantitative analysis.*
- *Process goals (productivities, defect densities, review effectiveness) are calculated from combining indicator data.*
- *Examples:*
  - *estimated in-process productivity = estimated size * progress in % complete / staffing level*
  - *defect density = defects found / estimated size*
  - *inspection effectiveness = inspection defects / inspection effort.*

# *Sample Process Goals*

- *Numeric values for goals are based on historical data and models*
- *The following productivity and quality measurements are derived from the progress, effort and defect indicator data*
- *Goal: control productivity*
  - *SLOCS/person-month (requirements)*
  - *SLOCS/person-month (design)*
  - *SLOCS/person-month (coding)*
  - *SLOCS/person-month (integration and test)*
  - *SLOCS/person-month (system test)*
  - *SLOCS/person-month (overall total)*

# *Sample Process Goals (cont.)*

- *Goal: control quality*
  - *Defects/KSLOC (introduced in requirements)*
  - *Defects/KSLOC (introduced in design)*
  - *Defects/KSLOC (introduced in coding)*
  - *Defects/KSLOC (found in requirements)*
  - *Defects/KSLOC (found in design)*
  - *Defects/KSLOC (found in coding/unit test)*
  - *Defects/KSLOC (found in integration)*
  - *Defects/KSLOC (found in system test)*
  - *Defects/KSLOC (found overall)*

# *Sample Process Goals (cont.)*

- *There are productivity and quality analogs for object-oriented development.*
  - *SLOCS replaced by classes or other OO artifacts*
- *Goal: control the peer review process*
  - *The following peer review measurements are derived from inspection data sheets.*
    - *Inspection (and other review) effectiveness*
    - *Inspection (and other review) rates*

# *Sample Product Quality Goals*

- *Defect density (defects/KSLOC)*
- *Critical computer resource utilization (%)*
- *Additional goals based on customer or organization requirements/priorities*
  - *functionality*
    - *response time*
    - *throughput*
    - *database size*
  - *reliability/maintainability (mtbf/mttr)*
  - *size (KSLOCs)*

# *Sample Defect Data*

- *Defect data should be collected by:*
  - *detection activity*
  - *when detected*
  - *introduction phase*
  - *type*
  - *mode*
- *A defect introduction and removal matrix can be generated and used for defect prevention to help answer "what are high-leverage opportunities for defect prevention / cost containment?".*

# *Use of Models*

- *Necessary in order to develop process goals, plans (including indicator trends), and define appropriate limits.*
- *Encapsulate our understanding of development processes (and support organizational learning).*
- *Benchmark process improvement since models are calibrated to our process environment.*
- *Examples:*
  - *productivity - COCOMO, SEER-SEM*
  - *defect density and defect rates - defect introduction and removal model, COCOMO II.1998, SEER-SEM, system dynamics process model*
  - *progress - S shaped curves, system dynamics process model*
  - *effort - Rayleigh curves, process model*
  - *any process parameter - statistical variation properties of actuals, Delphi poll results.*

# *Analysis Fundamentals*

- *Look for causes of variation in data*
  - *system or chance*
    - *normally can't change these*
    - *continuously active in the process*
    - *are part of the process*
  - *special or assignable*
    - *something can be done about it*
    - *not always active in the process*
    - *are "extraordinary" events*

# *System Causes of Variation*

- *When variation is due only to system causes, the variations will tend to take on a normal distribution*
- *The process is by definition running under stable conditions*
- *If special causes are present, the curve will be distorted or lost its normal distribution*
- *Need to understand significance of variations in order to control them*

# *Process Capability*

- *Process capability is concerned with the variation caused by all the sources within the system that can affect:*
  - *people, methods, machines, material, environment*
- *A capable process is under statistical control where assignable causes have been eliminated; it is a "stable" process*

# *Controlled Performance*

- *Given that the capability for a process is known, then*
  - *we can measure each new instance of the performed process against the capability definition*
  - *and when a special cause or a pattern is noted, understand what it means and correct it (if necessary) to bring future performance under control*

# *Statistical Process Control Techniques*

- *scatter diagrams*
- *check sheets*
- *Pareto diagrams*
- *run charts*
- *boxplots*
- ***histograms***
- ***control and attribute charts***
- *cause and effect diagrams*
- *process capability indices and ratios*

# *Frequency Histograms*

- *Take a snapshot of the process in time*
- *Can help keep track of variations*
- *Ask:*
  - *Do measures distribute into a bell curve?*
  - *What is the average?*
  - *Do we meet specifications?*
- *They do not provide information about patterns over time*

# *Control Chart*

- *Enables visibility into:*
  - *when the process is running satisfactorily*
  - *when something is different than expected and may need correction*
  - *patterns over time*
- *Measurements to learn capability must be collected over reasonable time period to include variations from all sources, e.g. 30 days*
- *Capability may change over long-term, so control charts should be revised periodically*

# *Control Chart Limits*

- *Calculate upper and lower control limits*
  - *e.g. UCL = average + 3*SQRT(average)*

    *LCL = average - 3*SQRT(average)*
- *Possibilities after limits are calculated:*
  - *process spread is normal and coincides with specification limits (process is capable)*
  - *spread is greater than specification limits (process is not capable)*
  - *spread is less than specification spread, but one of the control limits is outside the specification limit (adjustment required)*
  - *spread is centered on the specification and is less than the specification (ideal condition)*

# *Litton Implementation*

- *Quantitative Process and Product Management (QPPM) is our combined term that encapsulates both Level 4 KPAs Quantitative Process Management (QPM) and Software Quality Management (SQM).*

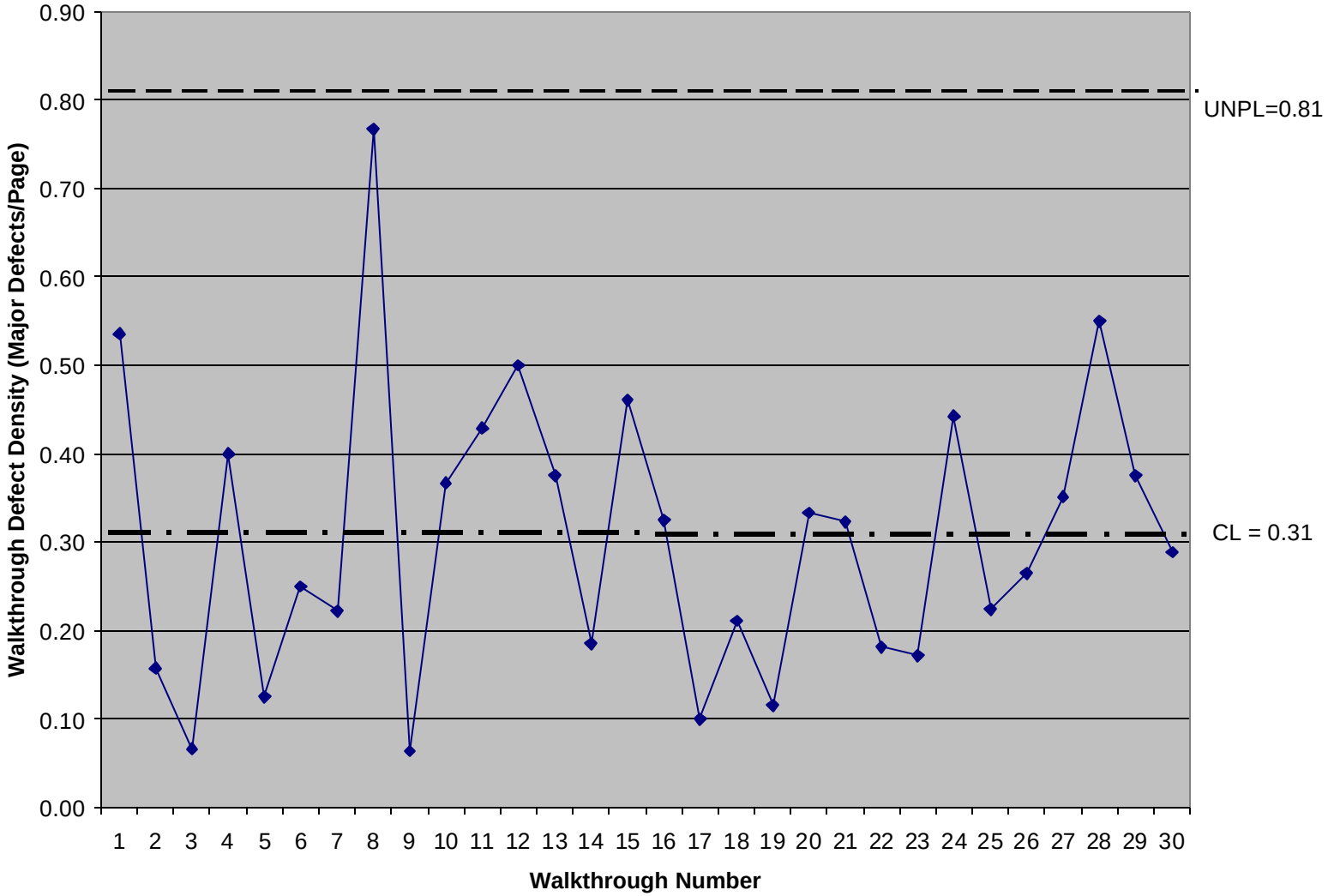
# *QPPM Documents and Data*

- *CSM 111 - Quantitative Process & Product Management*
- *CSM 209 - Quantitative Process & Product Management Plan*
- *SMH Ch. 8 - Process Measurement*
- *SEPG Handbook 950 - Quantitative Process & Product Management Analysis*
- *QPPM Plan template*
- *QPPM Project Plans*
- *PAL Process Database metrics*
- *QPPM Report*

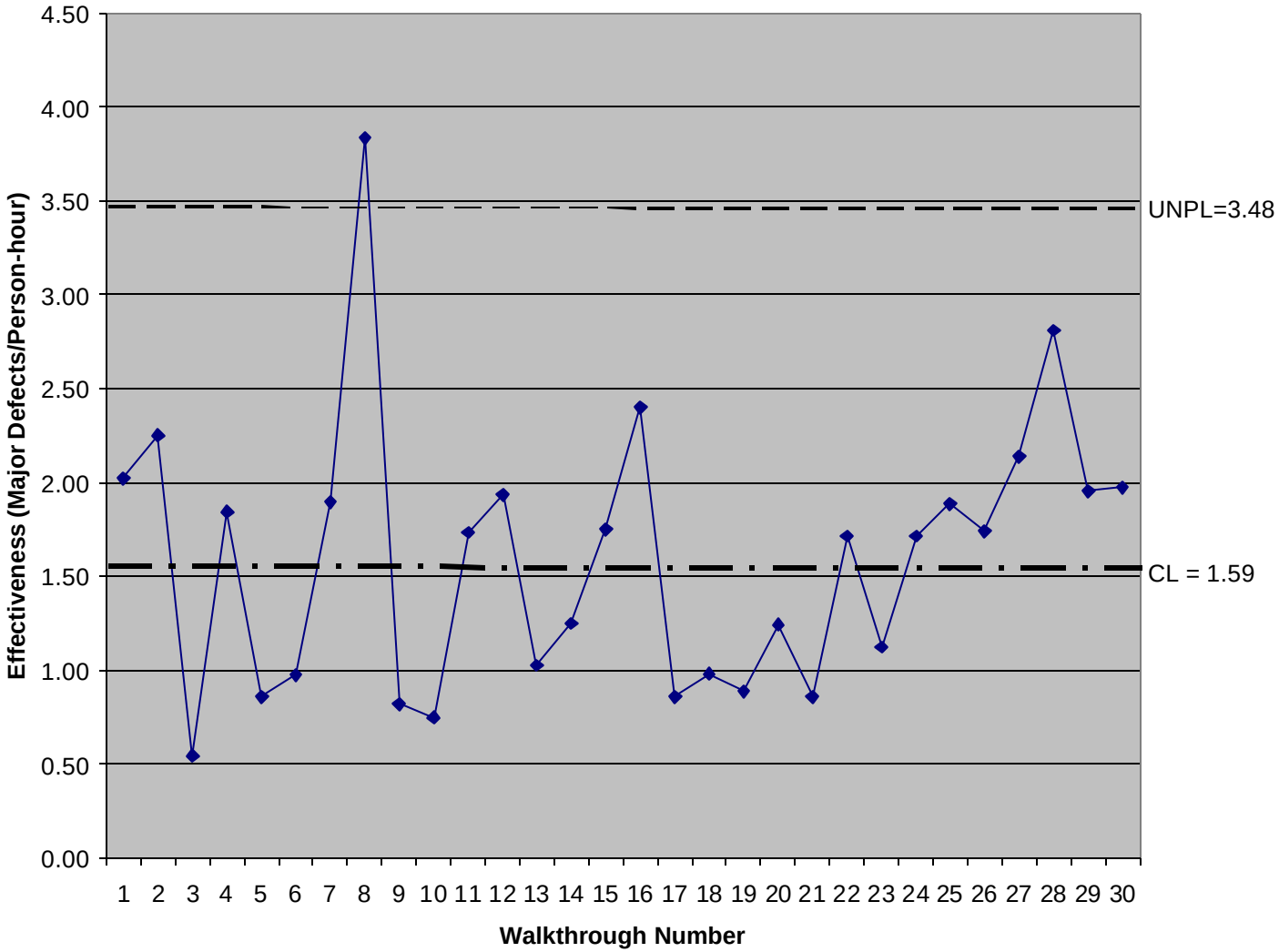
# *Sample QPPM Metrics*

- *Current measures (see following pages for examples from the PAL):*
  - *Productivity*
  - *Resource Utilization (CPU, Memory)*
  - *Defects*
  - *Peer Review Control Charts*

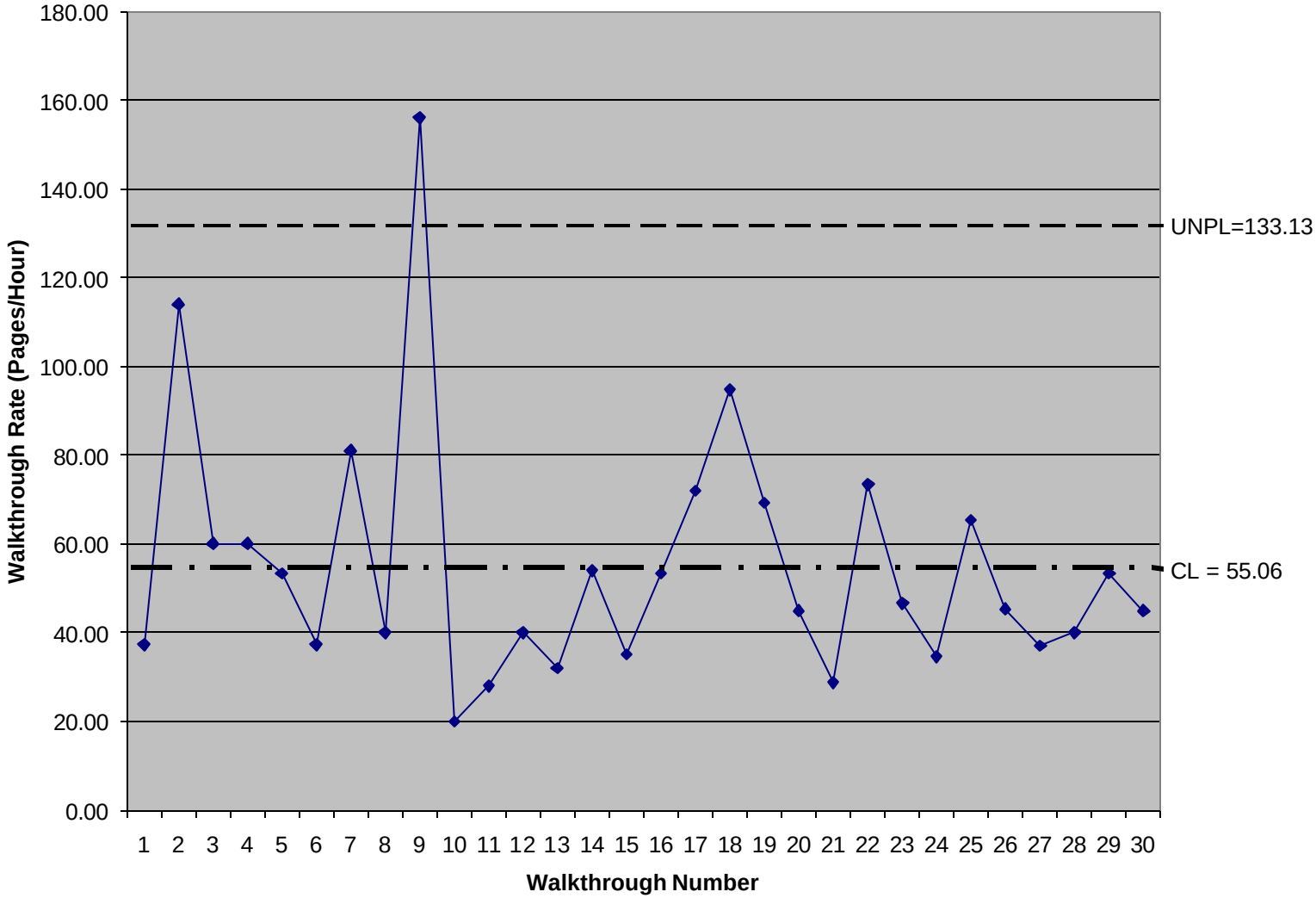
# *Design Walkthrough Defect Density*



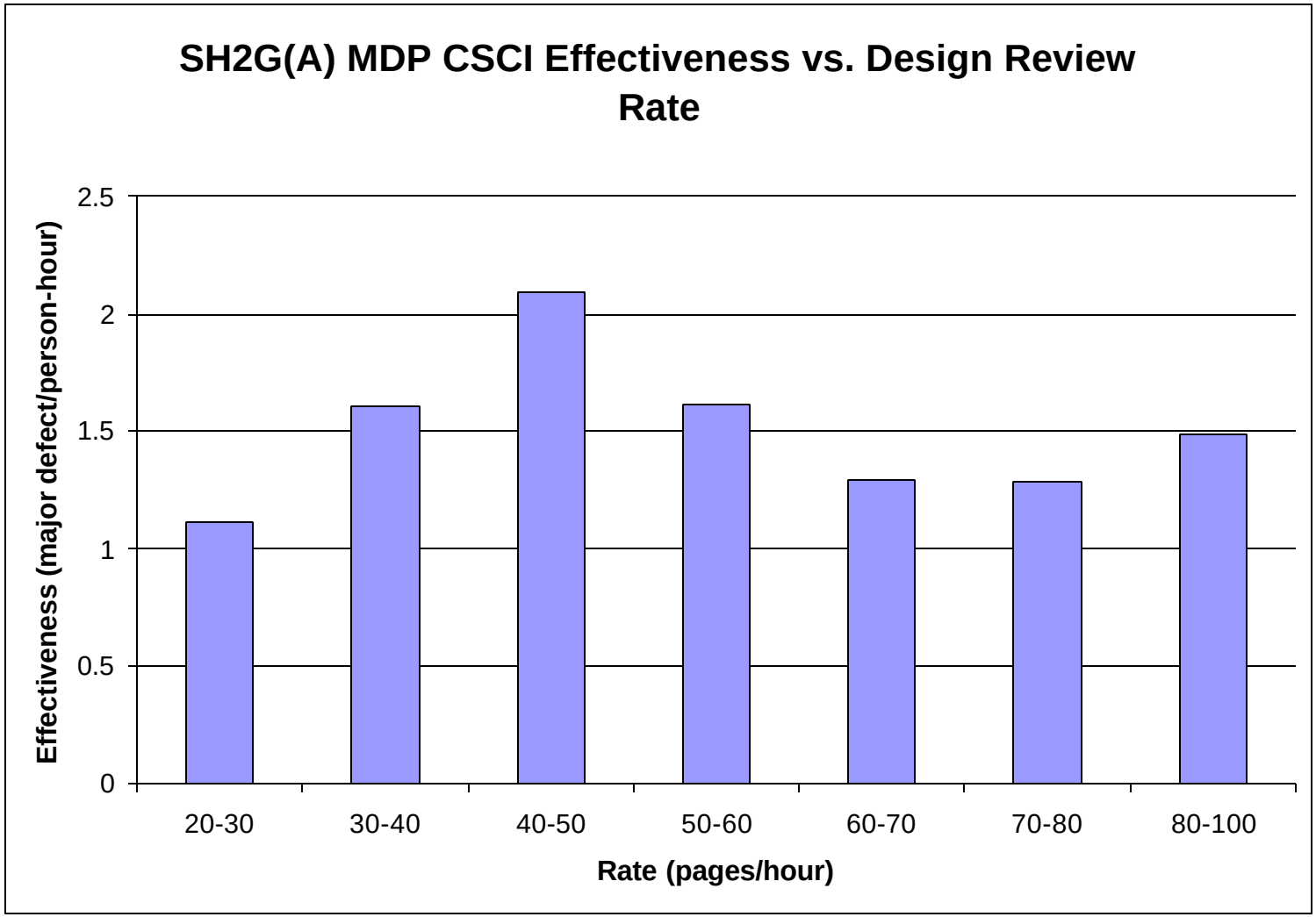
# *Design Walkthrough Effectiveness*



# *Design Walkthrough Rate*



# *Effectiveness vs. Review Rate*

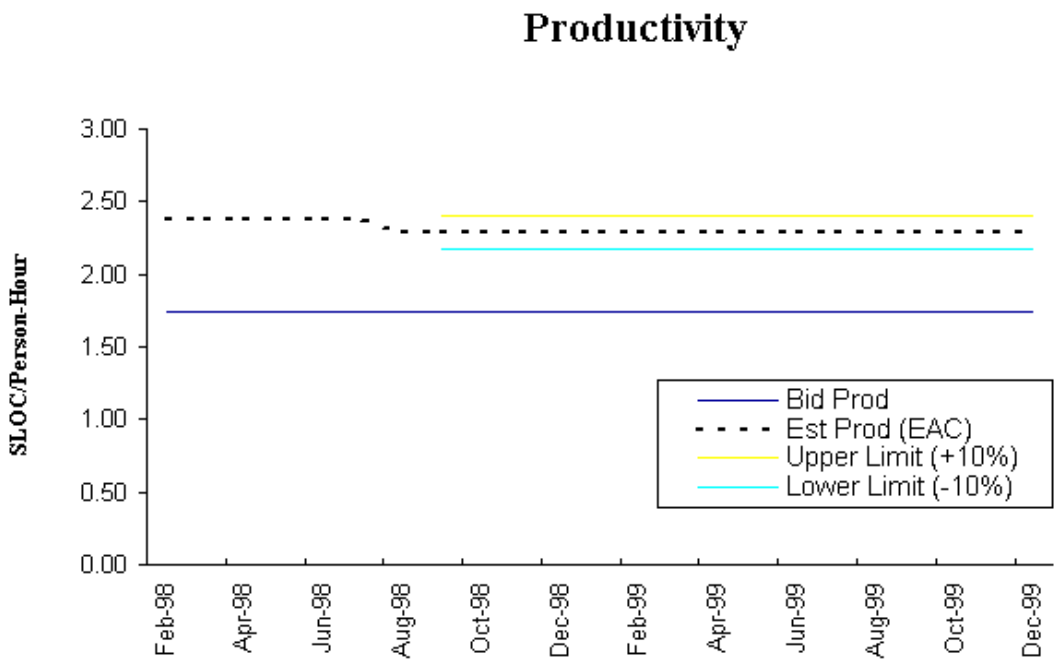


# *Productivity*

**Assumptions and rationale**

Assume final productivity per October 98 re-estimate.

Requires moderately tight control. A low-moderate 10% deviation limit is used.

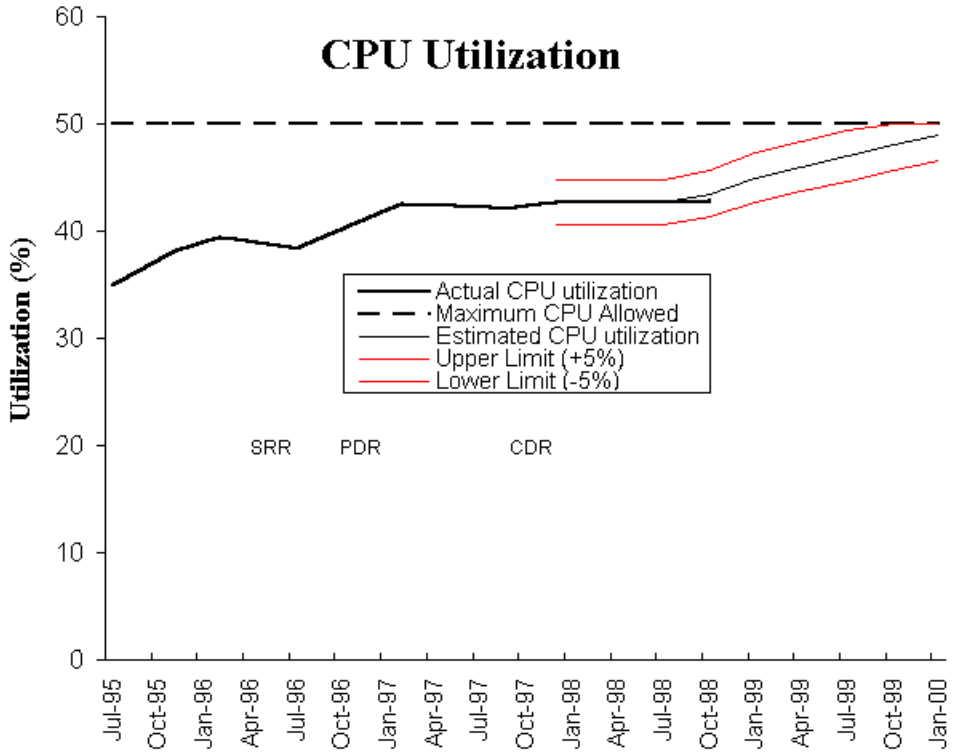


# CPU Utilization

**Assumptions and rationale**

Assume 15% more utilization for new application expansion from 10/97 level.

Requires very tight control and constant evaluation. A tight 5% deviation limit is used, with clipping to the maximum.

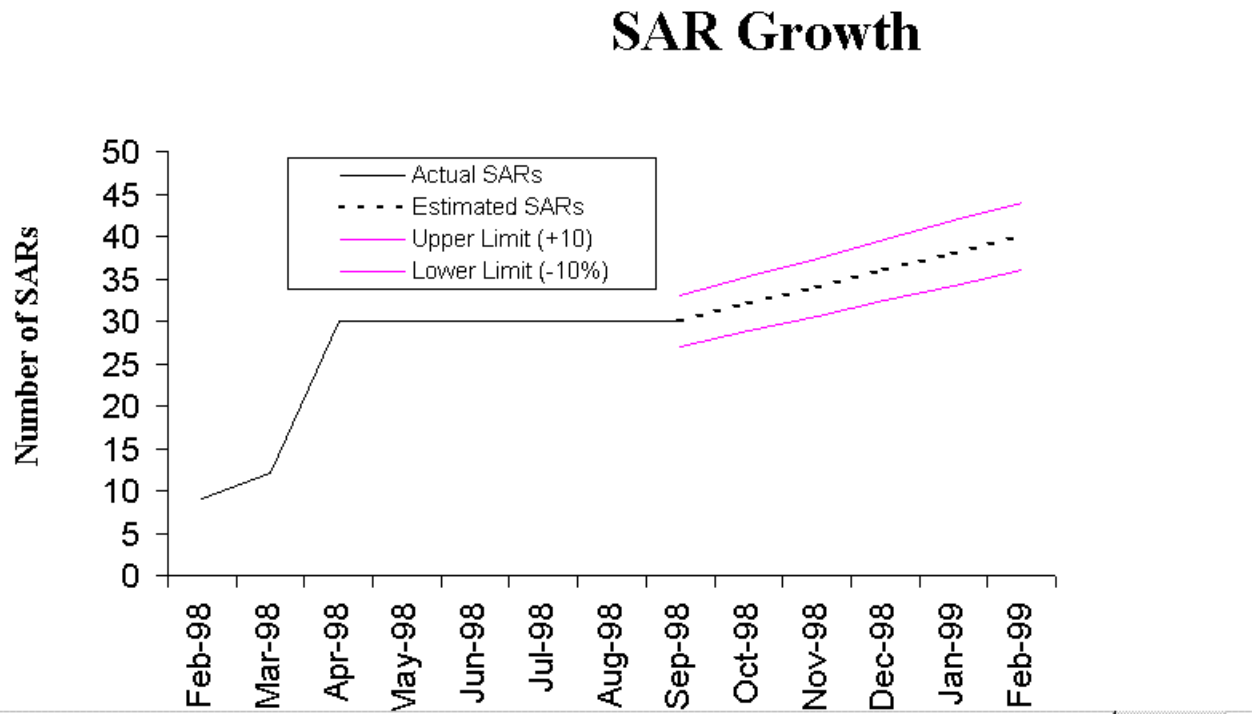


# *Defect Density*

**Assumptions and rationale**

Assume about 2 additional SARs per month beyond 9/98 level. With a product size of 63 KSLOC, this translates into a SAR Defect Density of 0.6 SARs/KSLOC at delivery time.

A moderately tight 10% deviation limit is used, since the system environment is well-known.



# *Level 4 Benefit Summary*

- *Increased process and product predictability*
- *Process capability quantitatively known and in line with business objectives*
- *Quality goals aligned with customer needs*
- *Capability baselines used to manage organization and set goals*
- *Ability to make informed decisions and make corrective actions sooner*
- *Ability to make accurate decisions*
- *Ability to support continuous process improvement*
- *Ability to compete in marketplace*
- *Minimizes project risk*