7th Annual CMMIR Technology Conference & Users Group

November 12-15, 2007
Denver, CA

Statistical Process Control Applied to Software Requirements Specification Process

Al Florence

The MITRE Corporation

The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or view points expressed by this author.

Overview

- Introduction
  - Background of Statistical Process Control
- Overview of Software Engineering Institute
  - Capability Maturity Model Integration
  - Quantitative Project Management and related Process Areas
- Statistic Process Control
  - Overview of Control Charts
- Examples of Control Charts
  - Applied to the Requirements Specification Process
- Conclusion
- Contact Information
- Acronyms/Abbreviations

Introduction

- Statistical Process Control (SPC) has been applied to manufacturing processes very effectively for many years.
- Recently software organizations, with higher process maturity levels, have started to apply SPC to their software development processes.
- Applying SPC to requirements efforts sets the stage for applying it to subsequent development activities.
- This may provided the biggest pay-off since most problems in software engineering can be directly traced to improper definition and specification of requirements.

Software Engineering Institute CMMI® (1 of 2)

- Capability Maturity
- CMMI® Level 4 - Quantitative Project Management
  - » SG 2 Statistically Manage Sub-process Performance
    - > The performance of selected sub-processes within a project's defined process is statistically managed.
      - SP 2.1 Select Measures and Analytic Techniques
      - SP 2.2 Apply Statistical Methods to Understand Variation
      - SP 2.3 Monitor Performance of the Selected Sub-processes
      - SP 2.4 Record Statistical Management Data

CMMI is a registered trademark of the SEI

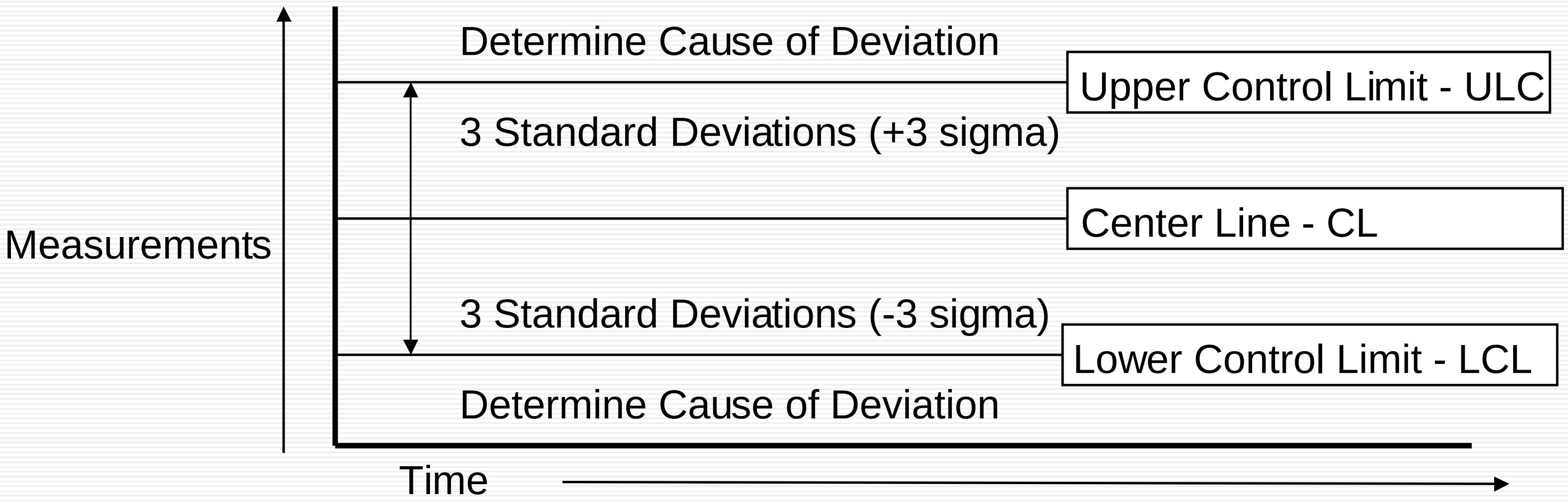
Software Engineering Institute CMMI® (2 of 2)

- CMMI® Other Process Areas
- CMMI® Level 5 - Causal Analysis and Resolution
  - » SG 1 Determine Causes of Defects
    - > Root causes of defects and other problems are systematically determined.
      - SP 1.1 Select Defect Data for Analysis
      - SP 1.2 Analyze Causes
  - » SG 2 Address Causes of Defects
    - > Root causes of defects and other problems are systematically addressed to prevent their future occurrence.
      - SP 2.1 Implement the Action Proposals
      - SP 2.2 Evaluate the Effect of Change
      - SP 2.3 Record Data

Statistical Process Control

- The intent of SPC:
  - » Is to better understand and monitor process behavior and to bring it under control when required.
  - » Is not necessarily to monitor products per se, although this may be a by-product of SPC.

Control Charts (1 of 9)



- According to the normal distribution, 99% of all normal random values lie within +/-3 standard deviations from the norm
- If a process is under Statistical Process Control, all measurements should fall within the 3-sigma limits
- If not, the anomaly needs to be investigated for cause and the process brought back under control

Control Charts (2 of 9)

- Control charts:
  - » Separate signal from noise
    - > so when anomalies occur they can be recognized
  - » Identify undesirable trends
    - > they point out:
      - Fixable problems
      - Potential process improvements
  - » Show the capability of the process
    - > so achievable goals can be set
  - » Provide evidence of process stability
    - > which justifies predicting process performance

Control Charts (3 of 9)

- Control charts use two types of data:
  - » variables data
  - » attributes data
- Variables data are usually measurements of continuous phenomena such as:
  - » elapsed time
  - » effort expended
  - » memory/CPU utilization

Control Charts (4 of 9)

- Attributes data are usually measurements of discrete phenomena such as:
  - » number of defects
  - » number of source statements
  - » number of people
- Most measurements in software used for SPC are attributes data.

Control Charts (5 of 9)

- The following are control charts that should be used for variables data and for attributes data:
  - Attributes Data
    - u charts
    - Z charts
    - XmR charts
  - Variables Data
    - X-bar charts
    - R charts
    - XmR charts

Control Charts (6 of 9)

- u charts are used when the data are samples from:
  - » a Poisson distribution, and
  - » the areas of opportunity are not constant
- Z charts can be used to avoid variable control limits for both large and small variations

Control Charts (7 of 9)

- XmR charts can be useful
  - » when little is known about the underlying distribution, or
  - » when the justification for assuming a binomial or Poisson process is questionable
- X-bar and R charts are used to portray process behavior when you have the option of collecting multiple measurements within a short period of time under basically the same conditions

Control Charts (8 of 9)

- Other sigma limits for homogeneous sets of data (The Empirical Rule)
  - » 1 sigma
    - > Roughly 60% to 70% of data will be located within 1 sigma
  - » 2 sigma
    - > Roughly 90% to 98% of data will be located within 2 sigma
  - » 3 sigma
    - > Roughly 99% to 100% of data will be located within 3 sigma

Control Charts (9 of 9)

- Tests for out-of control situations
  - » Test 1
    - > A single point falls outside the 3-sigma control limits
  - » Test 2
    - > At least 2 out of 3 successive points fall on the same side of, and more that 2-sigma units from, the center line
  - » Test 3
    - > At least 4 out of 5 successive points fall on the same side of, and more that 1-sigma unit from, the center line
  - » Test 4
    - > At least 8 successive values fall on the same side of the center line

Project Examples (1 of 2)

- A government agency, while re-developing legacy systems, reverse engineered the existing software requirements
- Five teams were assigned to reverse engineer related sets of functional requirements
- This author was assigned as a consultant to support the agency in the proper specification of the requirements

Project Examples (2 of 2)

- The examples illustrate:
  - » the proper specification of requirements
    - > Specification in this context means “writing” the requirements
  - » the application of control charts applied to the requirements specification process

Attendee Participation

What is wrong with this requirement?

After the system receives the Validation file, the system shall:

- notify the individual about acceptance or rejection.
- the acceptance file must contain the name and ZIP code of the individual.
- rejected validation request must include the Reason Code.

Criteria for Specifying a Good Requirement (1 of 4)

- The following are some critical attributes that requirements must adhere to:

  (used to critique the requirements)

- Completeness: Requirements should be complete
  - » They should reflect system objectives and specify the relationship between the software and the rest of the subsystems
- Consistency: Requirements must be consistent with each other; no requirement should conflict with any other requirement
  - » Requirements should be checked by examining all requirements in relation to each other for consistency and compatibility

Criteria for Specifying a Good Requirement (2 of 4)

- Feasibility: Each requirement must be feasible to implement
  - » Requirements that have questionable feasibility should be analyzed during requirements analysis to prove their feasibility
- Traceability: Each requirement must be traceable to some higher-level source, such as a system- level requirement
  - » Each requirement should also be traced to lower level design and test abstractions such as high-level and detailed-level design and test cases

Criteria for Specifying a Good Requirement (3 of 4)

- Testability: All requirements must be testable in order to demonstrate that the software end product satisfies its requirements
  - » In order for requirements to be testable they must be specific, unambiguous, and quantitative whenever possible. Avoid negative, vague and general statements
- Unique identification: Uniquely identifying each requirement is essential if requirements are to be traceable and testable
  - » Uniqueness also helps in stating requirements in a clear and consistent fashion

Criteria for Specifying a Good Requirement (4 of 4)

- Design Free: Software requirements should be specified at a requirements level not at a design level
  - » The approach should be to describe the software requirement functionally from a system (external) point of view, not from a software design point-of-view, i.e. describe the system functions that the software must satisfy.
- Use of “shall” and related words: In specifications, the use of the word "shall" indicates a binding provision
  - » Binding provisions must be implemented by users of specifications. To state non-binding provisions, use "should" or "may". Use "will" to express a declaration of purpose (e.g., "The Government will furnish..."), or to express future tense. MIL-STD-490A

Background

- It needs to be noted that requirements do not “live alone”
  - » They depend on other requirements and/or
  - » on clarifying comments

  to present a complete view of the functionality associated with a related set of requirements.
- A related set of functional requirements may be introduced with a preamble describing the capability of the functional set.
  - » The preamble does not itself establish requirements; this is done later in the requirements’ specifications.
- Some requirements may be amplified with clarifying comments which are, again, not part of the requirements, but add understandability.

Background

- Some requirements are documented sequentially with the requirements stated first setting the "stage" for the following requirements which add more and more capability.
  - » The later stated requirements depend on the earlier requirements to complete their functionally.
  - » An example may be the use of the word "processing". If the processing of a functional set of related requirements has been described in earlier requirements the later requirements may amplify and/or reference the processing without having to restate the processing.
- This is the case in the following examples; they have been extracted from a larger set of functionally related requirements and may not present a complete picture of the entire set.
- If a single requirement was to be a complete picture of a complex capability, one requirement would have to describe the entire capability making it extremely complex and difficult to understand, implement, and test.

Background

- The first set of requirements were received from the teams before they had been exposed to the critical attributes while the subsequent sets were received after they had incorporated review comments and had been trained on using the attributes.
- Later sets of requirements still had defects which were detected in subsequent critiques and used to create the control charts related to those iterative sets.
  - » This continued for several months until it was felt that the process was under statistical process control and that requirements were well specified.
  - » Because of this some readers may want to find additional issues associated with these examples, other than the ones listed in the critiques.
  - » Also, there may be issues with the re-specifications, but keep in mind that these hopefully would be identified in subsequent critiques.

Examples (1 of 2)

- The following examples illustrate the application of SPC to the process of specifying requirements
- The first two examples show some requirements
  - » As initially specified by the teams
  - » Followed by this authors critique against the critical attributes of requirements
  - » The re-specification of the requirements

Each violation against the critical attributes will be recorded as a defect to be used to construct control charts.

Examples (2 of 2)

- The next three examples show control charts applied to the specification of the requirements
  - » The first control chart example depicts the requirements specification process as being out of statistical process control
  - » The next control chart shows the process on the path towards being brought under control
  - » The third one shows the process under statistical process control

Example 1 (1 of 2)

- Initial specification:

  3.4.6.3 The system shall prevent processing of duplicate electronic files by checking the SDATE record. An e-mail message shall be sent.

- Critique:

1. Two “shalls” under one requirement number.
2. When is the SDATE record checked?
3. Against what other records is the SDATE record checked?
4. What is checked in the SDATE record?
5. To whom is the email message sent?
6. What does the email message say?
7. When is the email message sent?
8. The requirement has design implications, SDATE record.
   - A requirement should specify what the data in the record are and not the name of the record as it exists in the design and implementation

8 critical attributes violations (defects)

Example 1 (2 of 2)

- Re-specification:

  3.4.6.3 The system shall:

  a. Prevent processing of duplicate electronic files by immediately checking the date and time of the submission against prior submissions, and

  b. Immediately send the following e-mail message to submitter:

     1. Request updated submission date and time, if necessary, and
     2. State that the submission was successful, when successful.

Example 2 (1 of 2)

- Initial specification:
- After the system receives the Validation file, the system shall:
  - » Notify the individual about acceptance or rejection
  - » The acceptance file must contain the name and ZIP code of the individual
  - » Rejected validation request must include the Reason Code
- Critique:
  1. The second and third bullets don't make sense, try to read them as such:
     - the system shall the acceptance file must...
     - the system shall rejected validation…
  2. Use of both "shall" and "must"
  3. Where are the reason codes?
  4. Who is notified?
  5. How is the individual notified?
  6. No unique identifier
  7. Use of bullets, bullets are difficult to trace

7 critical attributes violations (defects)

Example 2 (2 of 2)

- Re-specification:

  3.2.7.3 When the system receives a validation file, the system shall:

  a. Reject the file if it does not contain the individual's

  1. name, and
  2. ZIP code, and

  b. Notify the individual via electronic transmission about acceptance or rejection with a reason code for rejection. (Reference Reason Code, Table 5.4.8), and

  c. Request corrected resubmission, if rejected.

Example 3 (1 of 4)

Out of Statistical Process Control

- Example 3 will show a control chart of all teams' attempts at the initially specification of the requirements
- This was before they received guidance on the critical attributes

Raw data collected from the initial specification of the requirements

Teams	No. Rqmts	Defects	*DefectsX100/ No of Rqmts
1	105	305	290.48
2	134	172	128.36
3	98	105	107.15
4	201	205	101.16
5	196	407	207.66
Totals	734	1194	

Defects normalized to 100 requirements

Example 3 (2 of 4)

Calculations to be used to construct the control chart

- Plot = Number of defects X 100 / requirements specified [calculated for each team's data]
- CL = (total number of defects/total number of requirements) X 100
- UCL = CL+3(SQRT(CL/a1) [calculated for each team's data]
- LCL = CL-3(SQRT(CL/a1) [calculated for each team's data]
- a1= Requirements specified/100 [calculated for each team's data]

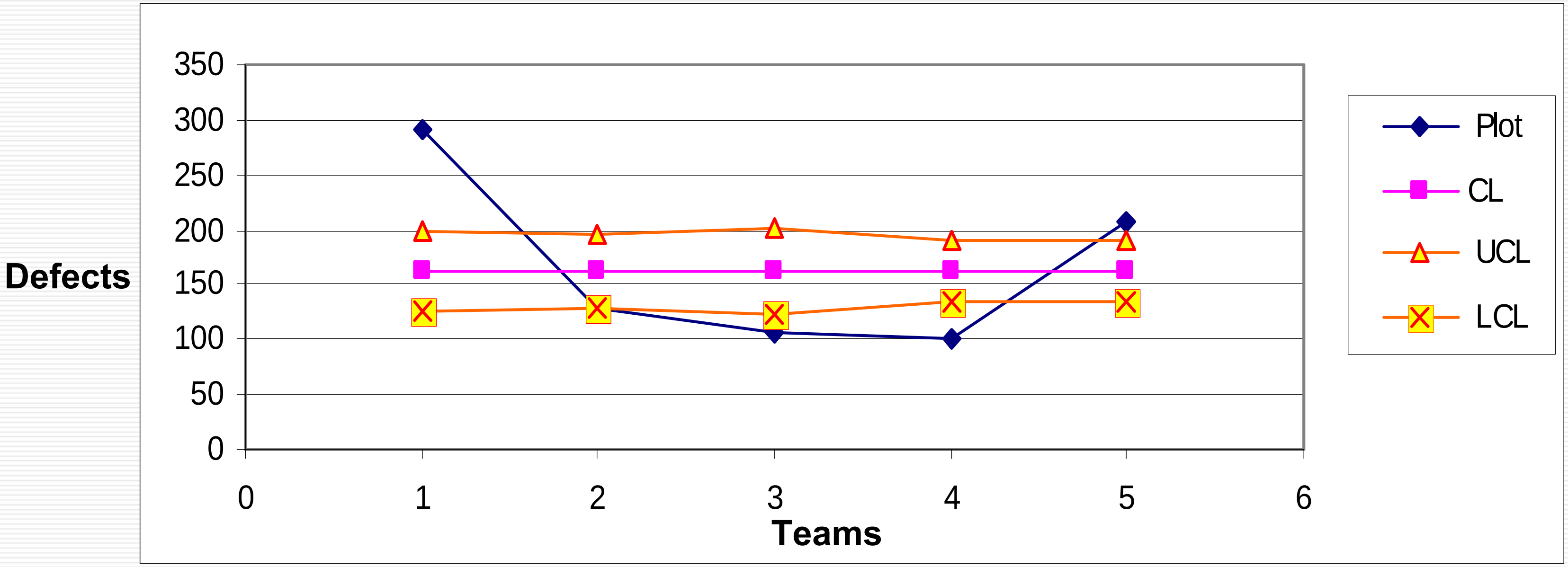
Example 3 (3 of 4)

Calculations to be used to construct the control chart

Teams	Plot	CL	UCL	LCL	a_1
1	290.48	162.67	200.01	125.33	1.05
2	128.36	162.67	195.72	129.62	1.34
3	107.15	162.67	201.32	124.03	0.98
4	101.10	162.67	189.66	135.68	2.01
5	207.66	162.67	190.00	135.34	1.96

Example 3 (4 of 4)

Control Chart for the Initial Specification of Requirements



- For control charts to be valid, they need to be used on processes that are mature and conducted consistently and on measurements that are valid, i.e. correctly depict the process
- This control chart showed that the process was immature and out of statistical process control
- The teams had not received guidance on the critical attributes of requirements, i.e., were not following a consistent process

Example 4 (1 of 3)

Toward Being Brought Under Statistical Process Control

- Example 4 will show a control chart of all teams' subsequent attempts at the specification of the requirements. New sets of requirements were included.
- The teams had been trained in the critical attributes and most had resolved the critique issues

Example 4 (2 of 3)

Raw Data

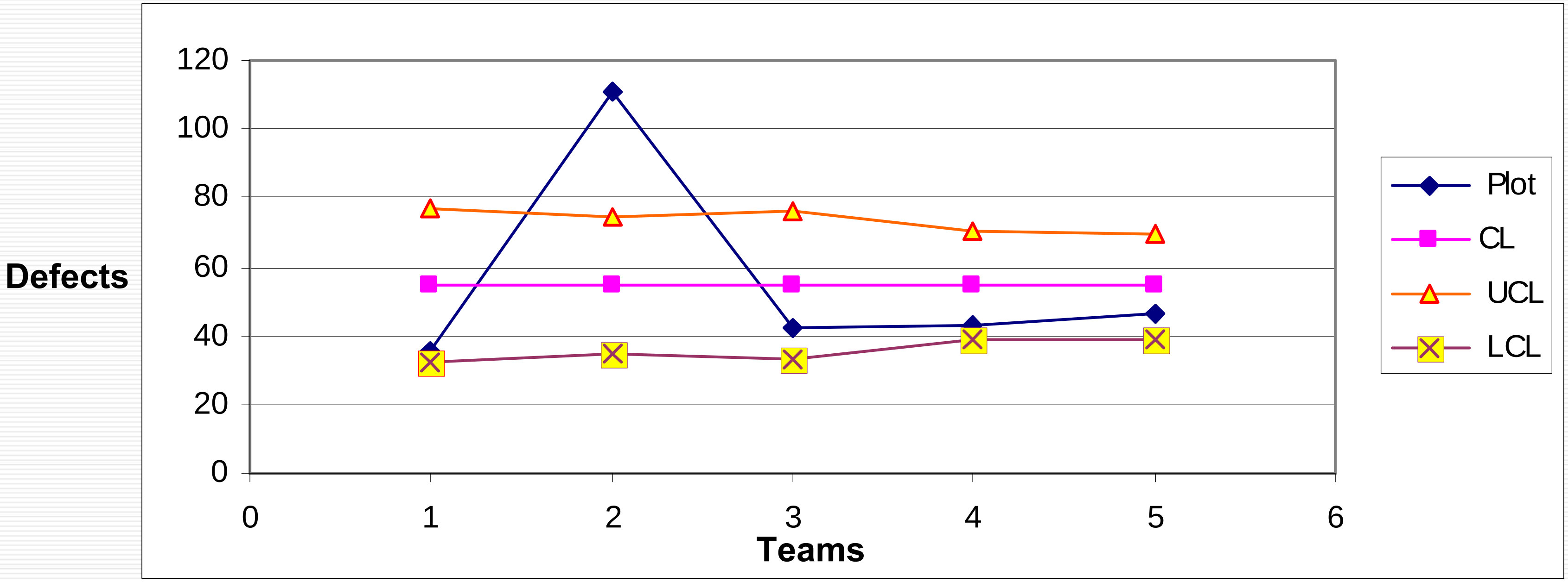
Teams	No. Rqmts	Defects	DefectsX100/ No. of Rqmts
1	98	35	35.71
2	125	139	111.20
3	107	45	42.06
4	198	85	42.93
5	205	95	46.34
Totals	733	399	

Calculations

Teams	Plot	CL	UCL	LCL	a_1
1	35.71	54.43	76.79	32.08	0.98
2	111.20	54.43	74.23	34.64	1.25
3	42.06	54.43	75.83	33.04	1.07
4	42.93	54.43	70.16	38.70	1.98
5	46.34	54.43	69.89	38.97	2.05

Example 4 (3 of 3)

Control Chart for Subsequent Specification of Requirements



An anomaly occurred with the second team's effort

Causal analysis revealed that the second team had not implemented the critique's findings nor analyzed new requirements against the critical attributes.

Example 5 (1 of 3)

Under Statistical Process Control

- Example 5 will show a control chart of all teams' subsequent attempts at the specification of the requirements. New sets of requirements were included.
- Management ensured that the second team resolved the issues identified in the critique and that they analyze additional requirements against the critical attributes.

Example 5 (2 of 3)

Raw Data

Teams	No. Rqmts	Defects	DefectsX100/ No. of Rqmts
1	105	2	1.90
2	116	4	3.45
3	101	6	5.94
4	205	9	4.39
5	298	14	4.70
Totals	825	35	

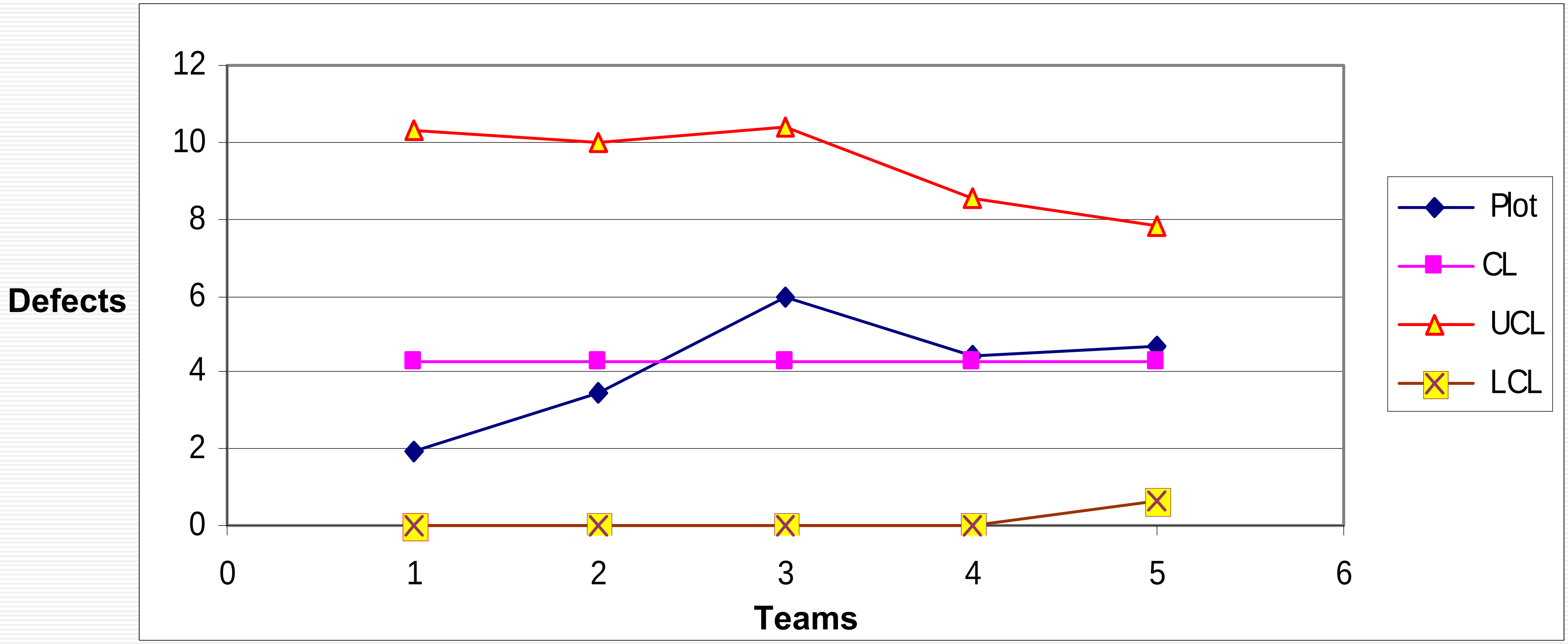
Calculations

Teams	Plot	CL	UCL	LCL	a_1
1	1.90	4.24	10.27	0	1.1
2	3.45	4.24	9.98	0	1.2
3	5.95	4.24	10.40	0	1
4	4.40	4.24	8.56	0	2.1
5	4.70	4.24	7.82	0.66	3

When the LCL is negative it is set to zero.

Example 5 (3 of 3)

Control Chart for Subsequent Specification of Requirements

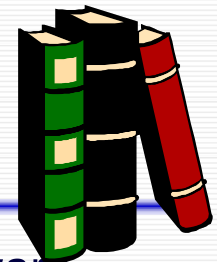


The requirements specification process is, for now, under statistical process control.

Conclusion

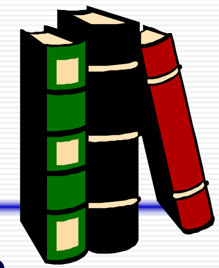
- The examples demonstrate the use of SPC applied to the requirements specification process. Many more control charts were constructed and analyzed. The ones use here were selected to succinctly demonstrate their use.
- The use of statistics using SPC control charts and other statistical methods can easily and effectively be used in a software setting. SPC can identify undesirable trends and can point out fixable problems and potential process improvements and technology enhancements.
- Using SPC, beginning with requirements analysis, can provide the biggest payoff. It is a well-known fact that if requirements are properly defined early in the development life cycle, the migration of problems into the later phases will be mitigated.

References and Suggested Readings (1 of 2)



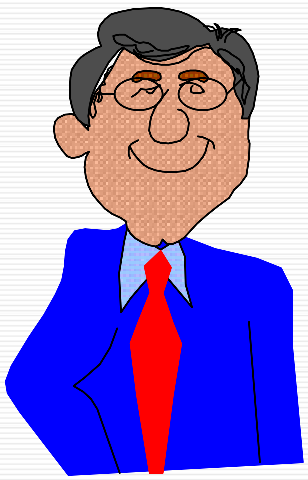
- *Capability Maturity Model® Integration (CMMISM), Version 1.1*. Software Engineering Institute. December 2001
- Chambers, David S., Wheeler, Donald J., *Understanding Statistical Process Control*, SPC Press. 1995
- Florac, William, Carleton , Anita D., *Measuring the Software Process, Statistical Process Control for Software Process Improvement*, Addison Wesley. 1999
- Florence, Al, *CMM Level 4, Quantitative Analysis and Level 5, Defect Prevention*. Software Technology Conference. 2000; CrossTalk, The Journal of Defense Software Engineering. February 2001
- Florence, Al, *Reducing Risk with the Proper Specification of Software Requirements*. PSQT/PSTT Conference, March 2002. CrossTalk, The Journal of Defense Software Engineering. April 2002

References and Suggested Readings (2 of 2)



- Florence, Al: Bail, William, *Tutorial: Effective Requirements Practices*. Software Technology Conference, 2003
- *IEEE Recommended Practices for Software Requirements Specifications*. IEEE Std 830-1998, IEEE Computer Society. October 20, 1998
- Institute of Electrical and Electronics Engineers. *Recommended Practice for Software Requirements Specifications*. IEEE Std 830-1993
- Pyzdek, Thomas, *An SPC Primer*, Quality America, Inc. 1984
- Wheeler, Donald J., *Advanced Topics in Statistical Process Control*, SPC Press. 1995

Contact Information



Al Florence
florence@MITRE.org
(703) 983-7476

Abbreviations

- CL - Center Line
- CMMI® - Capability Maturity Model Integration
- ET - Eastern Time
- FA - Financial Agent
- LCL - Lower Control Limit
- SPC - Statistical Process Control
- UCL - Upper Control Limit