

Charting the Progress of System Development Using Defect Data

Copyright © 1999 Rex Black. All Rights Reserved.

Rex Black

President and Principal Consultant

RBCS

7310 Beartrap Lane

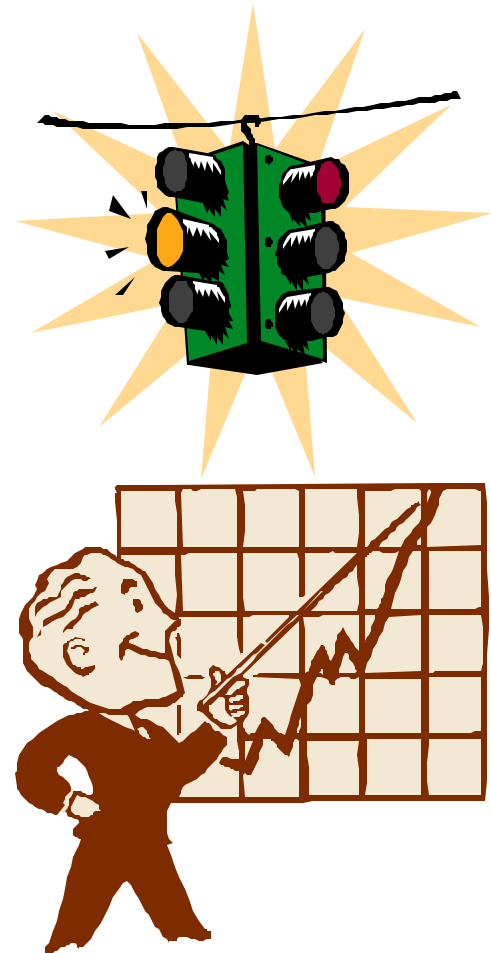
San Antonio, TX 78249

www.rexblackconsulting.com

Rex_Black@rexblackconsulting.com

Benefits of Charting Bug Data

- Assess Product, Process, and Project Quality
 - Product stability
 - Defect removal trends
 - Root cause analysis
 - Bug management
 - Hot spots
- Communicate Status to Peers and Management
 - Summarize key facts and underlying trends
 - Get the point across quickly.
 - Manage key indicators, not the crisis du jour.



Case Study: SpeedyWriter

<i>Phase</i>	<i>Cycle</i>	<i>Start Date</i>	<i>End Date</i>	<i>Bugs Found</i>
Component test	1	7/19/99	7/25/99	25
	2	7/26/99	8/1/99	20
	3	8/2/99	8/8/99	5
Integration test	1	8/2/99	8/8/99	20
	2	8/9/99	8/15/99	15
	3	8/16/99	8/22/99	5
System test	1	8/16/99	8/22/99	10
	2	8/23/99	8/29/99	5
	3	8/30/99	9/5/99	0
First customer ship (FCS)	—	9/13/99	—	—

Key fields in bug tracking database for metrics

Bug report opened date

Bug root cause

Bug report closed date

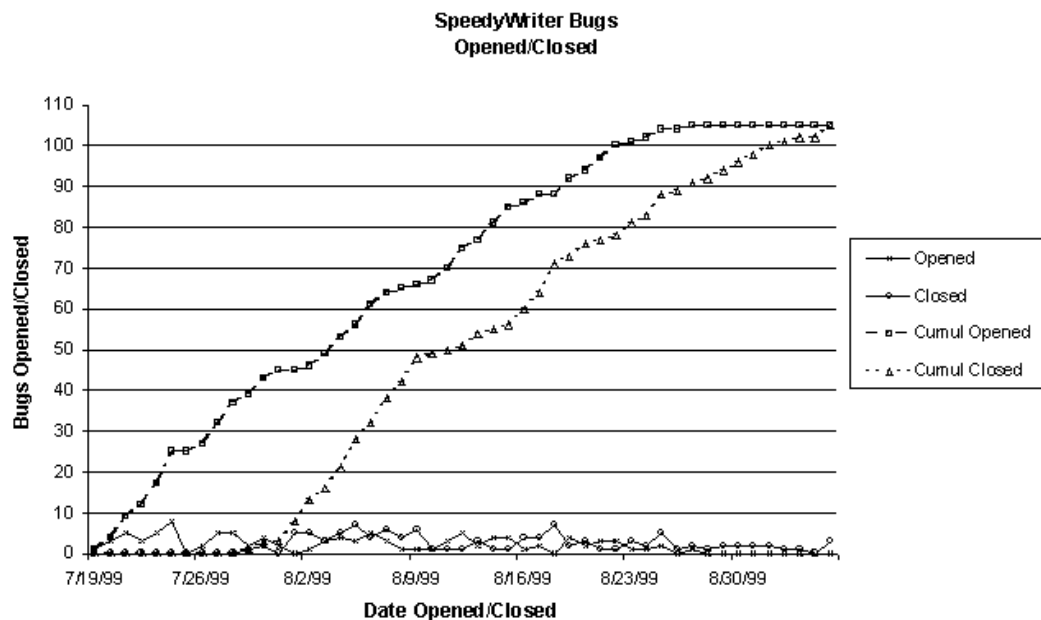
Affected subsystem

Export the bug reports to a spreadsheet to prepare charts

Bug reports approximate (linearly) underlying bugs (about 27% high)

Bugs Opened/Bugs Closed

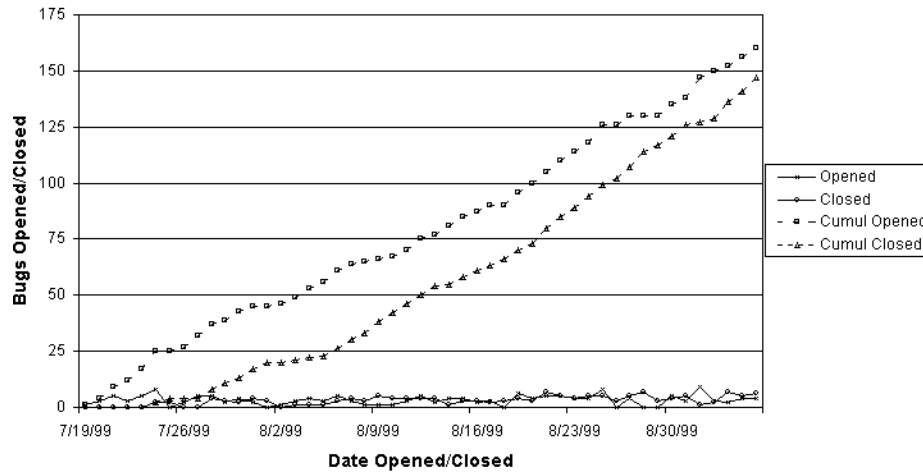
- Cumulative opened curve becomes flat as system under test stabilizes and test system finds all the bugs it can identify.
- Cumulative closed curve converges to cumulative open curve, closing the quality gap, as system under test approaches “customer ready” state.
- A cumulative closed curve that tracks the cumulative open curve indicates crisp bug management.



- Milestones in the project show up as changes in the shape of the curves.
- In a worksheet, use Excel COUNTIF () function for opened and closed bug reports by date and two running totals to create the chart.

Opened/Closed: Trouble Indicators

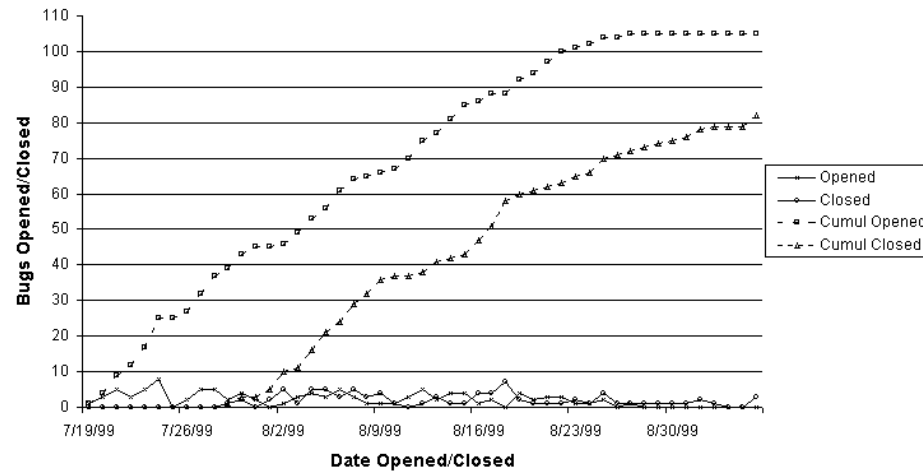
SpeedyWriter Bugs
Opened/Closed



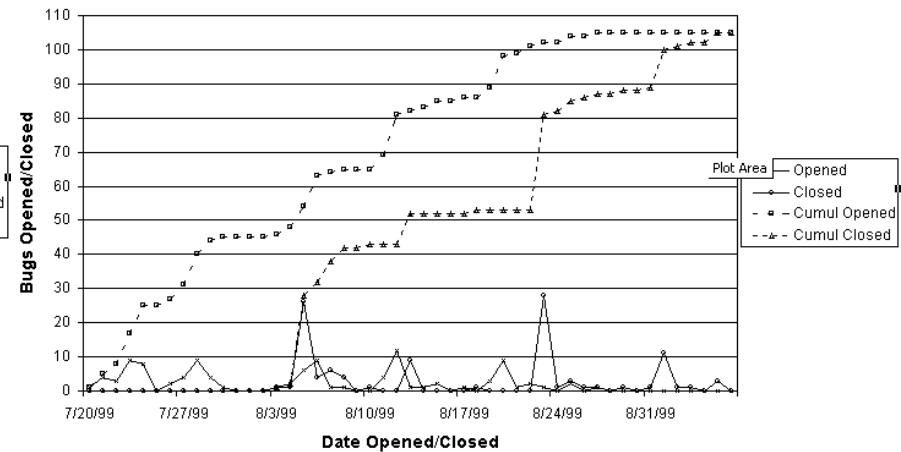
Counterclockwise from left:

- Endless bug discovery (product never stabilizes)
- Ignored bug reports (persistent quality gap)
- Poor report management (sloppy open/close process)

SpeedyWriter Bugs
Opened/Closed

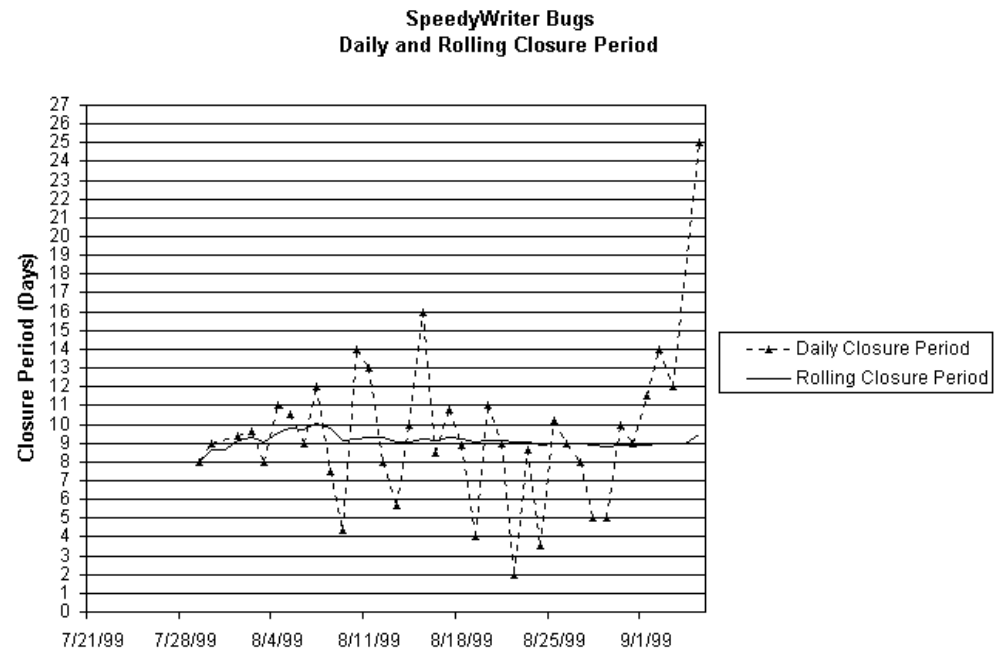


SpeedyWriter Bugs
Opened/Closed



Closure Period

- Measures turnaround time on reported bugs (developer responsiveness):
 - Daily closure period is average turnaround for all bugs closed on a given day
 - Rolling closure period is average turnaround for all project bugs closed to date
- A stable closure period implies a smooth rolling closure period and daily closure periods that are bounded at the top end by two or three times the test cycle duration (in this example, each cycle is one week).
- An acceptable closure period shows bugs moving to resolution in the time specified by project management.
- This chart measures the quality of the overall process, including development and test, for bug fixing.



Calculating Closure Period

1. Find the closure period for each closed bug report.

Microsoft Excel - Good SW OC RC SS CG Analysis.xls

File Edit View Insert Format Tools Data Window Help

D2 = =IF(B2>0,IF(C2>0,IF(ISNUMBER(C2),C2-B2,NA()),NA()),NA())

	A	B	C	D	E	F	G	H	I	J
1	Bug ID	Date Opened	Close Date	Closure Gap						
2	1	7/21/99	8/3/99	13						
3	2	7/21/99	7/30/99	9						
4	3	7/24/99	8/1/99	8						
5	4	7/23/99	8/2/99	10						
6	5	7/24/99	8/1/99	8						

2. Calculate the total closure period for all bug reports by date.

Microsoft Excel - Good SW OC RC SS CG Analysis.xls

File Edit View Insert Format Tools Data Window Help

B2 = =SUMIF('Closure Gap-Calc'!\$C\$2:\$C\$500,\$A2,'Closure Gap-Calc'!\$D\$2:\$D\$500)

	A	B	C	D	E	F	G	H
1	Date	Closure Period	Cum Closure Period	Inactive	Cum Inacti	Daily Closure Period	Rolling Closure Period	
2	7/21/99	0	0	0	0	#N/A	#N/A	
3	7/22/99	0	0	0	0	#N/A	#N/A	
4	7/23/99	0	0	0	0	#N/A	#N/A	
5	7/24/99	0	0	0	0	#N/A	#N/A	
6	7/25/99	0	0	0	0	#N/A	#N/A	

Calculating Closure Period (cont)

3. Count the number of bugs closed on each date.

Microsoft Excel - Good SW DC RC SS CG Analysis.xls

File Edit View Insert Format Tools Data Window Help

D2 = =COUNTIF('Closure Gap-Calc'!\$C\$2:\$C\$500,\$A2)

	A	B	C	D	E	F	G	H
1	Date	Closure Period	Cum Closure Period	Inactive	Cum Inacti	Daily Closure Period	Rolling Closure Period	
2	7/21/99	0	0	0	0	#N/A	#N/A	
3	7/22/99	0	0	0	0	#N/A	#N/A	
4	7/23/99	0	0	0	0	#N/A	#N/A	
5	7/24/99	0	0	0	0	#N/A	#N/A	
6	7/25/99	0	0	0	0	#N/A	#N/A	

4. Determine the daily and rolling closure periods.

Microsoft Excel - Good SW DC RC SS CG Analysis.xls

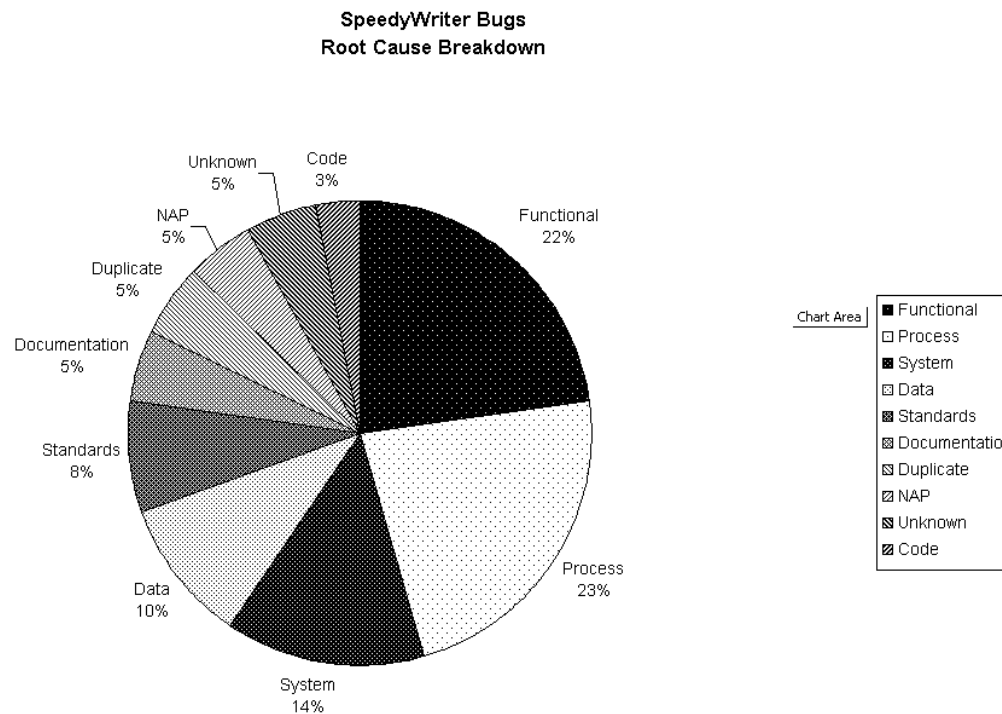
File Edit View Insert Format Tools Data Window Help

G2 = =IF(C2>0,IF(E2>0,C2/E2,NA()),NA())

	A	B	C	D	E	F	G	H
1	Date	Closure Period	Cum Closure Period	Inactive	Cum Inacti	Daily Closure Period	Rolling Closure Period	
2	7/21/99	0	0	0	0	#N/A	#N/A	
3	7/22/99	0	0	0	0	#N/A	#N/A	
4	7/23/99	0	0	0	0	#N/A	#N/A	
5	7/24/99	0	0	0	0	#N/A	#N/A	
6	7/25/99	0	0	0	0	#N/A	#N/A	

Root Cause Breakdown

- Understand the underlying mistakes, miscommunications, and other errors that lead to bugs and bug reports.
- Allow for course-correction during the project.
- Provide information for long-term development process improvement.



Subsystem Breakdown

- Analyze which portions of the system experience the most problems.
- Increase testing of the subsystems that produce the most bug reports (where there's one bug, there's another).
- Improve the development processes (including internal testing) for the subsystems that are most error-prone (preventing the additional bugs is cheaper than finding them).

