

Achieving Six Sigma Software Quality Through the Use of Automatic Code Generation

Bill Potter
Senior Staff Engineer
Honeywell International

© 2005 The MathWorks, Inc.



MIADC2005

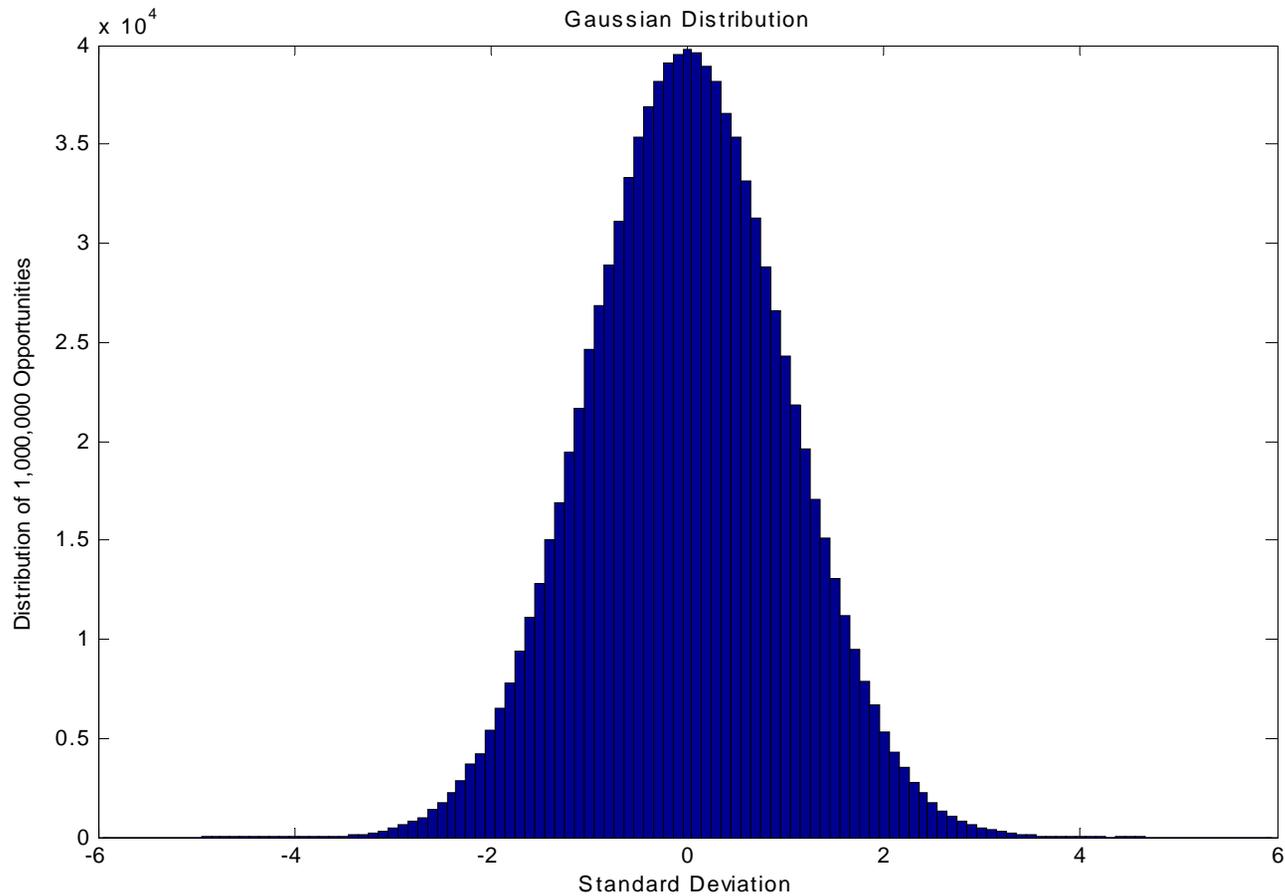
The MathWorks
International Aerospace and Defense
Conference

Accelerating the Pace of Engineering in Aerospace and Defense

Six Sigma

- A common approach to Six Sigma quality is eliminating defects through DMAIC (define, measure, analyze, improve, control).
- To achieve Six Sigma, the defect rate must be no more than 3.4 defects per 1 million opportunities.
- Six Sigma has been successfully applied to manufacturing processes and it is currently being attempted on hardware and software development processes at some companies.

Gaussian Distribution Plot



Where is industry today?

- A typical software process today injects 100 defects per 1000 lines of source code using manual processes.
 - 40 defects are due to design errors (3.25 sigma)
 - 60 defects are due to coding errors (3.05 sigma)
- If Detroit built your car the way companies develop software, what would it look like?

Car built via typical software process



Use of Real-Time Workshop® Embedded Coder at Honeywell

- Honeywell has used Simulink®, Stateflow® and Real-Time Workshop Embedded Coder to produce production code since 1996
 - Limited subset of Simulink blocks
 - Limited subset of StateFlow constructs
 - Fixed set of code generator options

Coding Defect Measurement Data for 2003 and 2004 on actual production programs

- During these two years, 1.6 million source lines of code were generated using Real-Time Workshop Embedded Coder and compiled using a Borland compiler.
- One defect was found during a code review.
- No defects were found during software component testing.

Six Sigma Achieved!

- One defect per 1.6 million lines of code = 6.35 sigma
 - This was the defect injection rate for the coding process
 - Zero coding defects went to the customer

Resolution of the defect

- The one defect was that the multi-port switch code was applying loop rolling at a threshold of 2 instead of a threshold of 5.
- The code passed testing and achieved MCDC coverage (per DO178B Level A) even with this coding defect.
- The TLC file for this block was modified to eliminate any further defects.

Conclusion

- Manufacturing processes have been automated in order to achieve Six Sigma quality levels.
- When we automate the manufacturing of software, through the use of automatic code generation, we are also able to achieve Six Sigma quality levels.
- So, what does that car look like when it is automatically coded?

Car built via automatic code process



Editors note: Author's hobby is demolition derby driving