Software Cost and Productivity Model

presented to

Ground Systems Architecture Workshop 2004

Manhattan Beach, California

presented by

J. E. Gayek, L. G. Long, K. D. Bell, R. M. Hsu, and R. K. Larson*

The Aerospace Corporation

31 March 2004

*Currently with the National Aeronautical and Space Administration

THE AEROSPACE CORPORATION

© 2004 The Aerospace Corporation

Acknowledgements

- Sponsorship and funding for this research came from an FY03 Aerospace Corporation internal research & development initiative

- Special thanks are extended to following for their support:
  - Nancy Kern
  - Carolyn Latta

Topics of Presentation

- Background
- Data Collection & Definitions
- Data Characterization
- Software Metrics
- Conclusion

Topics of Presentation

- Background
- Data Collection & Definitions
- Data Characterization
- Software Metrics
- Conclusion

Rationale for Project

- Software is a vital part of space & ground systems
- Software development is often a cost & schedule risk to acquisition
  - Amount of effort is under-estimated
  - Productivity of staff is over-estimated
  - Cost for unit of labor is under-estimated
- Assessing the severity of cost & schedule risk of new software can be gauged by examining development of completed modules of similar functionality
- Aerospace previously developed database & estimating relationships (1996) to address this need
  - Used by SMC / NRO / NASA
- Update needed to investigate impact of current techniques and languages

Areas of Investigation

- Productivity
  - What is the likelihood the contractor can achieve a development rate of X source lines of code (SLOC) per developer-month (DM)?
  - What is the approximate distribution of productivity?
- Schedule
  - What kind of schedule duration can be expected for a given SLOC size?
- Effort
  - How many DM are required to develop software of a given SLOC size, operating environment, and application domain?

Topics of Presentation

- Background
- Building the Analysis Database
- Data Characterization
- Software Metrics
- Conclusion

Building the Analysis Database

- Call for data went to:
  - Aerospace program offices
  - Government organizations that Aerospace supports (e.g., SMC, Air Force Cost Analysis Agency)
  - DoD and civilian agencies Aerospace does not support on routine basis (e.g., Navy Cost Analysis Division)
  - Contractors in the space industry
- Data sought on development efforts that commenced or completed after January 1, 1996
  - Software development comprises product design, code, and CSCI testing
  - Front-end software system requirements and tail-end system-level integration and test not included

Software Complexity

- Software complexity expressed in terms of operational environment and application domain

- Operational Environment (Platform)
  - Military Ground
  - Military Mobile
  - Avionics
  - Unmanned Space

- Application Domain
  - Command/Control
  - Database
  - Mission Planning
  - Operating System (OS) / Executive
  - Signal Processing
  - Simulation
  - Support
  - Test

Definitions of Operating Environments

Military Ground	This environment includes ground-based mission-critical software. The hardware platform is usually located at a fixed site where diagnosis of faults can be performed readily. The developer may even be on-site or readily available for updates and revisions. Since consequences of failure are least here (being readily fixable) of the four environments, demands on reliability are not as high.
Military Mobile	Software residing in a vehicle, such as van, trailer, deployable structure, or ship. Operational consequences of a software fault are higher than for fixed sites. Software may interact with hardware through real-time measurements or open-loop control of servomechanisms.
Mil-Spec Avionics	Software for operation or control of aircraft or similar vehicle control. Often real-time; may contain closed-loop control algorithms or control of hardware. May operate critical instrument gauges, perform measurements or perform data reduction from sensors. Required reliability is high. Software errors may result in loss of mission, vehicle, and/or life.
Unmanned Space	Similar to MIL-SPEC Avionics in requirements for reliability and impact of mission. Defects in system control software may result in loss of mission. Other software may be revised and uploaded depending on circumstance and vehicle design. Operating environment is often difficult to replicate on the ground, so data-processing software may require extensive advance simulation to validate its effectiveness. Often operates in real-time, performs measurements with hardware interaction, controls vehicle with closed-loop feedback.

SEER-SEM Manual, version 6.0.28, December 2002 (Galorath Inc., 100 N. Sepulveda Blvd., El Segundo, Calif., 90245)

Definitions of Software Domains (1/2)

Command & Control	Examples include network monitoring, network control and switching, sensor control, signal/telemetry processing, message processing, data reduction/analysis, mission control, and command processing.
Database	Software that collects, stores, organizes and indexes information. Examples include database generation and database management systems.
Mission Planning	Software used to support mission-planning activities such as space mission planning, aircraft mission planning, scenario generation, feasibility analysis, route planning, and image/map manipulation.
Operating System/Executive	Software that controls basic hardware operations and serves as a platform for applications to run. Multi-user operating systems provide management and security of system users. Operating system functions may include network, security, file management, device drivers, display drivers, multi-processing, multi-tasking, multi-threading, and real time operating systems.
Signal Processing	Software used to enhance, transform, filter, convert, or compress data signals. Signal processing has application in many areas such as communications, flight system, sonar, radar, and medical systems. Large volumes of data are processed using complex algorithms, often with real time operating requirements.

SEER-SEM Manual, version 6.0.28, December 2002 (Galorath Inc., 100 N. Sepulveda Blvd., El Segundo, Calif., 90245)

Definitions of Software Domains (2/2)

Simulation	Software that evaluates numerous scenarios and summarizes processes or events to simulate physical processes, complex systems or other phenomenon that may not have simple empirical relationships. Examples include environment simulation, system simulation, emulation, process flow, network simulation, operations flow, and system reliability programs.
Support	All software used to aid the development, and testing and support of applications, -systems, test and maintenance, and trainer software
Test	Software used for testing and evaluating hardware and software systems. Examples include test case generation, test case data recording, test case data reduction/analysis, and test driver / stub tools and programs.

SEER-SEM Manual, version 6.0.28, December 2002 (Galorath Inc., 100 N. Sepulveda Blvd., El Segundo, Calif., 90245)

Data Requested

- Program name
- CSCI Name / Functional Description
- Language*
- SLOC (New/Actuals)*
- SLOC (Original Estimate)*
- SLOC (Revised Estimate)
- SLOC (Reuse)*
- SLOC (Equivalent)*
- Labor Effort (Man-months)*
- Labor Effort (Hours)
- COTS Packages (Type)
- COTS Integration Code (SLOC)
- COTS Effort (Man-months)
- COTS Effort (Hours)
- Software Level*
- Operating Environment*
- Application Domain*
- Software Development Completion (Year)
- Peak Development Staff (Number)
- Data Source / Comments

***Key quantity to basic analyses**

Data Dictionary (1/2)

Program Name	The name of the program for the information being provided.
CSCI Name/Functional Description	The name of the CSCI and/or Functional Description.
Language	The primary computer programming language used in the software development.
SLOC (New/Actuals)	Enter the actual source lines of code (SLOC) provided from the delivered development.
SLOC (Original Estimate)	Enter the estimated source lines of code (SLOC) provided from the beginning of the project.
SLOC (Revised Estimate)	Enter any revised SLOC estimate since the beginning of the project but prior to the delivered development.
SLOC (Reuse)	Enter the SLOC size designated as reuse or modified code.
SLOC (Equivalent)	Enter the equivalent SLOC size when the element contains a combination of new and reused code.
Labor Effort (Man-months)	Enter the labor effort in man-months required to develop the software.
Labor Effort (Hours)	Enter the labor effort in hours required to develop the software.

Data Dictionary (2/2)

COTS Packages (Type)	Major emphasis of COTS package (e.g. scientific, administrative).
COTS Integration Code (SLOC)	Amount of code required to integrate a COTS product.
COTS Effort (Man-months)	Effort required to get familiar with data & testing of COTS products in manmonths.
COTS Effort (Hours)	Effort required to get familiar with data & testing of COTS products in labor hours.
Software Level	The software level of the data record (e.g. Project, CSCI, CSC)
Operating Environment	The environment which best describes the primary mission of the software (e.g. mil-ground, unmanned space).
Application Domain	The application which best describes the primary function of the software (e.g. mission planning, command & control, test).
Software Development Completion (Yr)	The year in which the software development was completed.
Peak Development Staff (Number)	The peak staffing per month, in man-months for this development.
Data Source/Comments	List the source of data record and other comments that may be warranted.

Adjustments to Data for Analysis

- Not all responses included SLOC(Equivalent) when SLOC(New/Actuals) and SLOC(Reuse) are given
  - SLOC(Equivalent) is a key parameter for the analyses
  - Created an estimate of SLOC(Equivalent) using SEER-SEM® (default values) with the given values of SLOC(New/Actuals), SLOC(Reuse), operating environment and application domain
- Not all responses reporting software levels as CSCI and Project made sense
  - For purposes of analysis, activities with SLOC(Equivalent) < 200K were treated as CSCI; otherwise, the activity was treated as a Project

From here onward, SLOC will refer to SLOC(Equivalent), unless stated otherwise

Topics of Presentation

- Background
- Building the Analysis Database
- Data Characterization
- Software Metrics
- Conclusion

Data Record Summary

	ENVIRONMENT			
APPLICATION	MILITARY GROUND	MILITARY MOBILE	MIL-SPEC AVIONICS	UNMANNED SPACE
COMMAND/CONTROL	45	35	2	25
DATABASE	13	5	4	0
MISSION PLANNING	45	14	1	1
OS/EXECUTIVE	9	3	9	5
SIGNAL/PROCESSING	27	21	2	11
SIMULATION	19	0	1	25
SUPPORT	104	1	1	4
TEST	13	2	4	1
SUBTOTALS	275	81	24	72
TOTAL DATA BASE SIZE = 452 Records				

Observations

- None of the data records contain COTS information
  - Is it being collected?
  - Did respondents choose not to provide data on CSCIs with COTS?
  - Is there less use of COTS than expected?
- None of the data records contain labor effort in hours
  - Use reported labor in man-months for productivity analyses
  - Although some data providers gave the cost to produce the code, such information was not used
    - Conversions would produce uncertainties
- 186 records contain data useful for productivity analysis
- 27 records contain data useful for code growth analysis
- 22 records contain both code growth and productivity information

Topics of Presentation

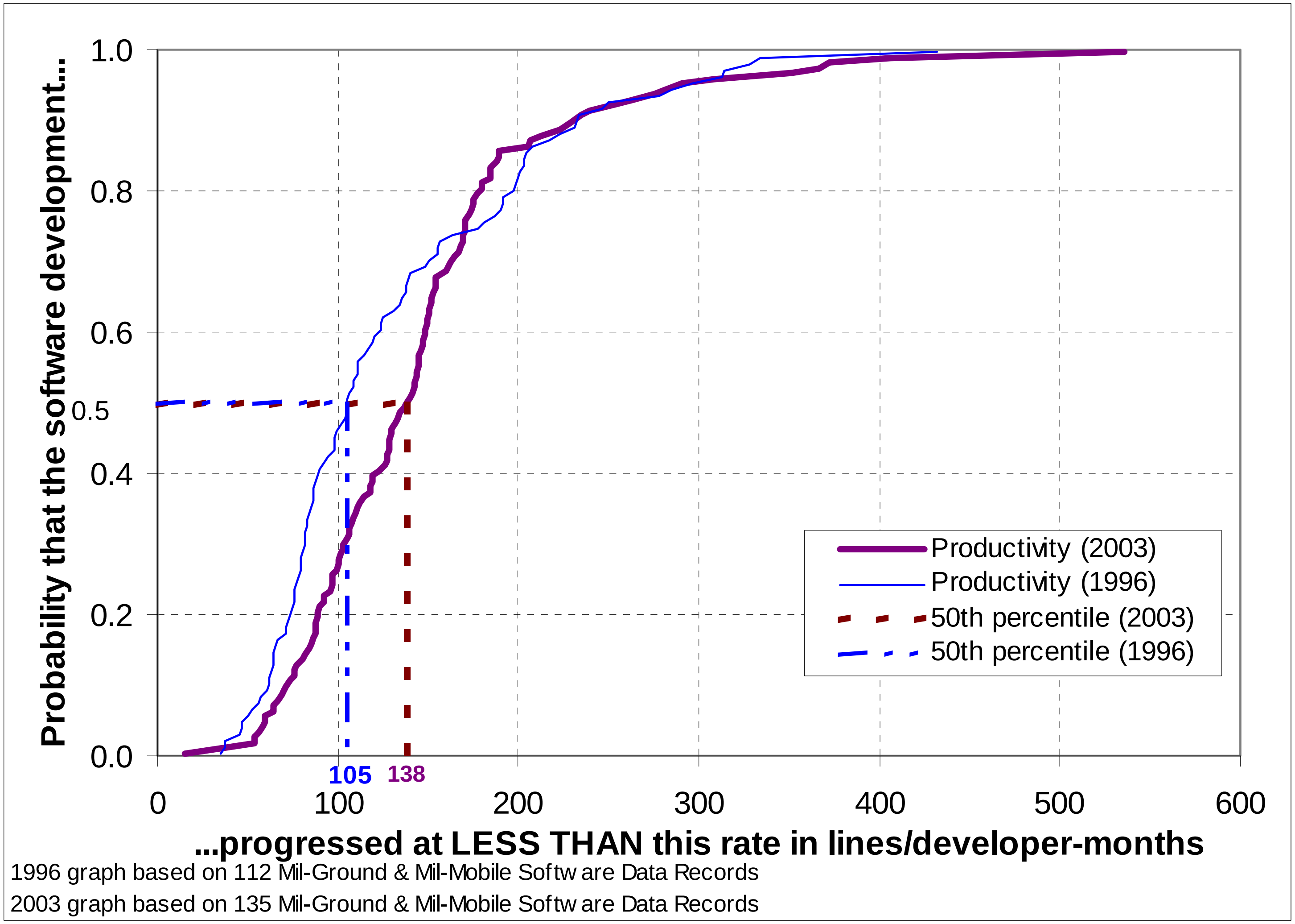
- Background
- Building the Analysis Database
- Data Characterization
- Software Metrics
- Conclusion

Summary of Productivity Analysis

- 186 records contain productivity information
- C, C++, and Ada are the languages predominantly reported
  - In 1996 study, most languages reported were Fortran, Jovial, and other “higher order languages”
- Comparing productivity distributions with 1996 data indicates an increase in median productivity levels
  - 31% increase in combined military ground & military mobile environments (2003 study median: 138 SLOC/DM)
  - 33% increase in combined military-spec avionics & unmanned space (2003 study median: 64 SLOC/DM)
- Median productivity of C/C++ in the combined military ground & military mobile environments is 148 SLOC/DM

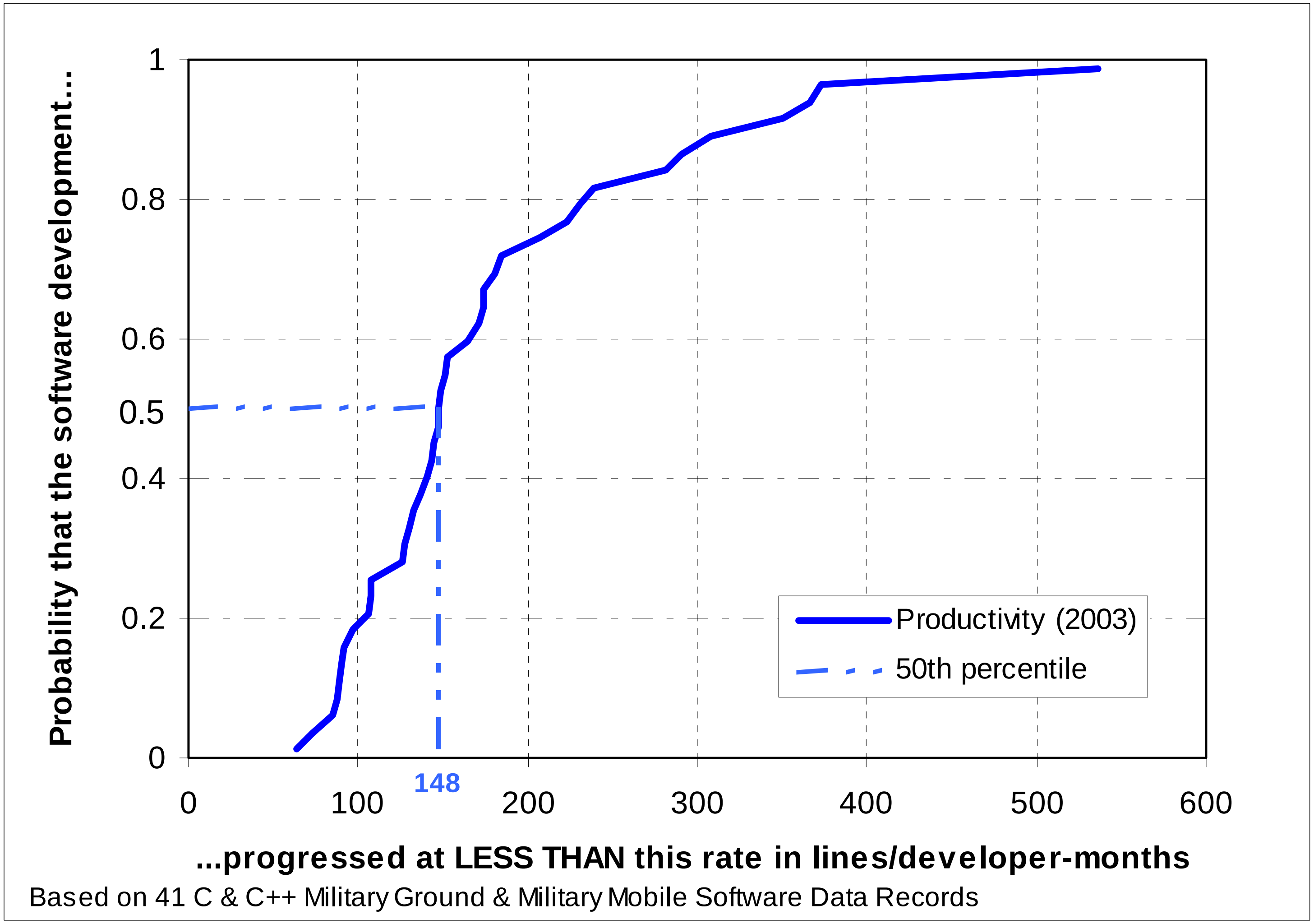
Productivity

(Military Ground & Military Mobile)



Productivity

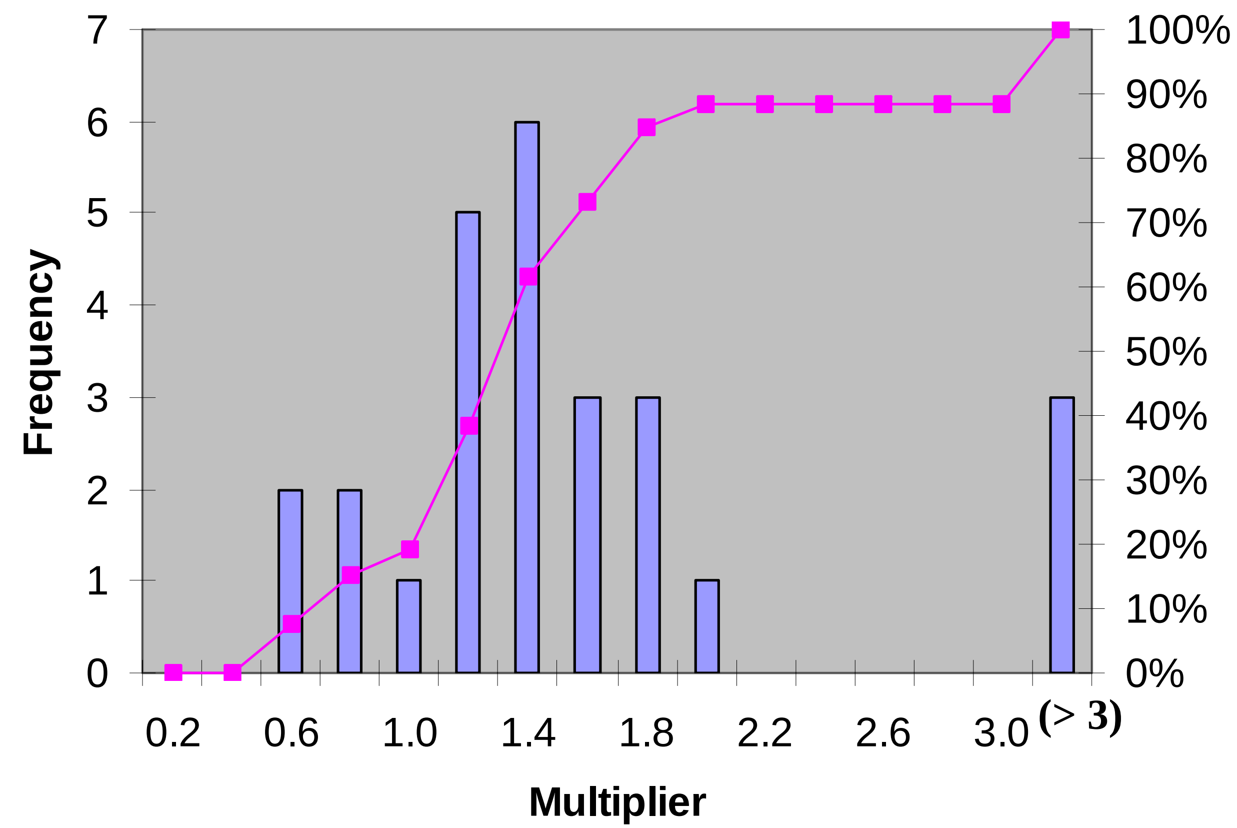
(C and C++)



Summary of Code Growth Analysis

- 27 data records contain code growth information
  - All associated with military mobile environment
  - None of these records had reused code
- Analysis based on code growth multiplier
  - Multiplier = SLOC(New/Actuals) / SLOC(Original Estimate)
- 80% of these records show code growth
  - 60% of the records have multipliers of at least 1.2
- Summary statistics on code growth multipliers:
  - Low: 0.52
  - Average: 1.49
  - High: 5.01

Distribution of Code Growth Multipliers



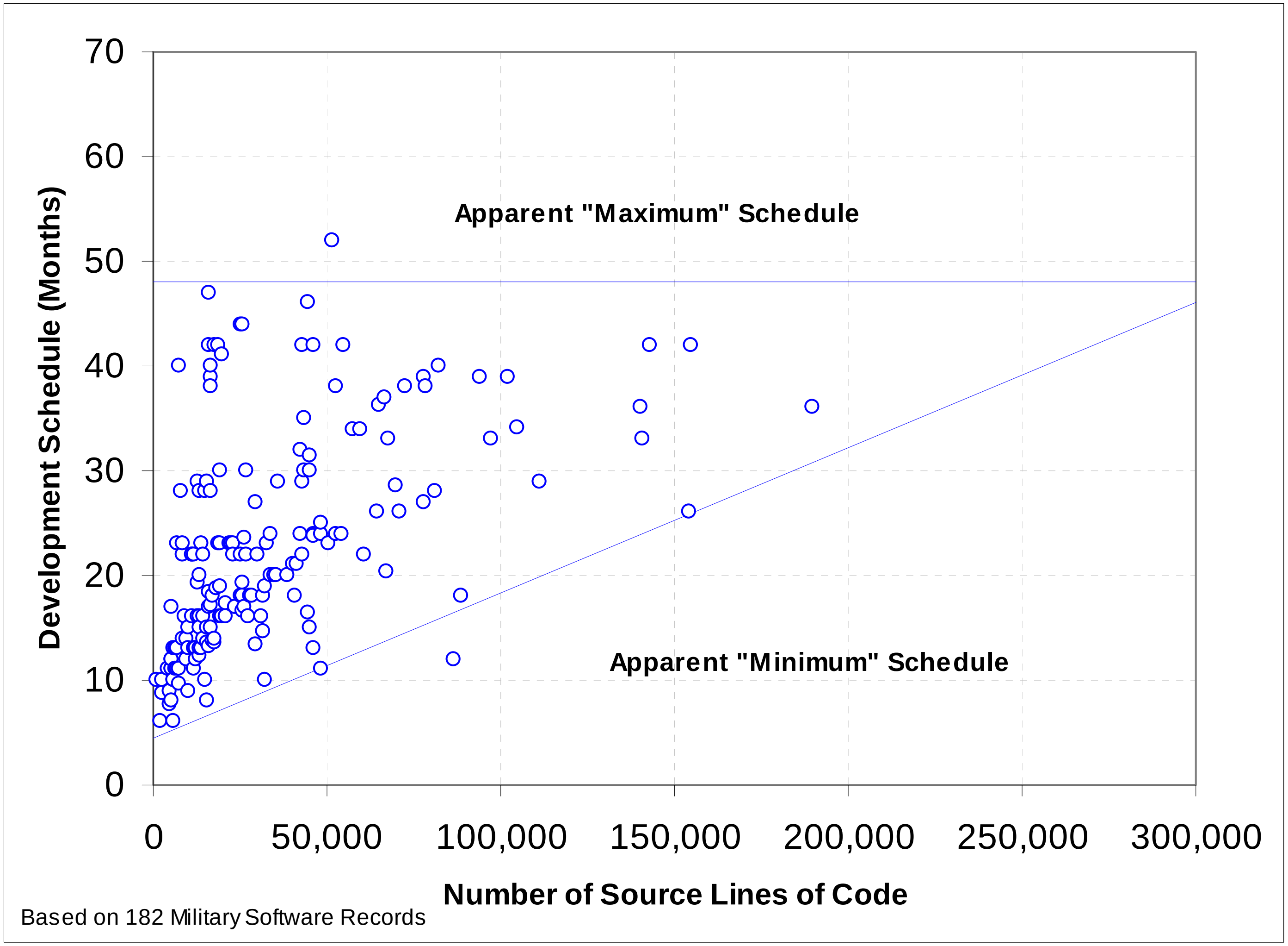
Bin intervals: $0.2(k\text{-}1) \leq m \leq 0.2k$, $k = 1, 2, \ldots, 15$, and $m > 3.0$

Low: 0.52 Ave: 1.49 High: 5.01

Summary of Development Schedule

- Data indicates that it is possible to complete moderately sized records (i.e., scores of KSLOC) in the same time to complete small records (i.e., thousands of SLOC)
  - Larger activities can have increased staffing, working in parallel
- There are apparent minimum and maximum schedules
  - No records lasted more than 52 months
    - Activities projected to last more than 48 months (the apparent maximum) might be cancelled, renamed, redirected or restarted and do not appear in the database
  - The apparent minimum appear to be a linear function of SLOC
    - MinSched[months] = 4.5 + 0.000139*SLOC
  - 99% (180 out of 182) of the applicable records fall within the apparent minimum and maximum schedule bounds

Distribution of Project Schedule



THE AEROSPACE CORPORATION

Cost Estimating Relationship Analysis

- Nonlinear regression analysis (i.e., curve fitting) conducted to find relationship between SLOC and the associated number of DM
  - Executed on environment / application pairs with at least 5 data points
  - The resulting functional relationship can be called a cost estimating relationships (CER) if the number of DM is equated with cost.

- Results in 11 CERs (down from 12 CERs in 1996)

Gains in 2003	Losses in 2003
Military Mobile/Mission Planning	Military Ground/Database
Military mobile/Signal Processing	Military Ground/Mission Planning
Mil-Spec Avionics/Oper. Sys/Executive	Unmanned Space/Database
	Unmanned Space/Test

- None of the environments has a database CER using 2003 study data

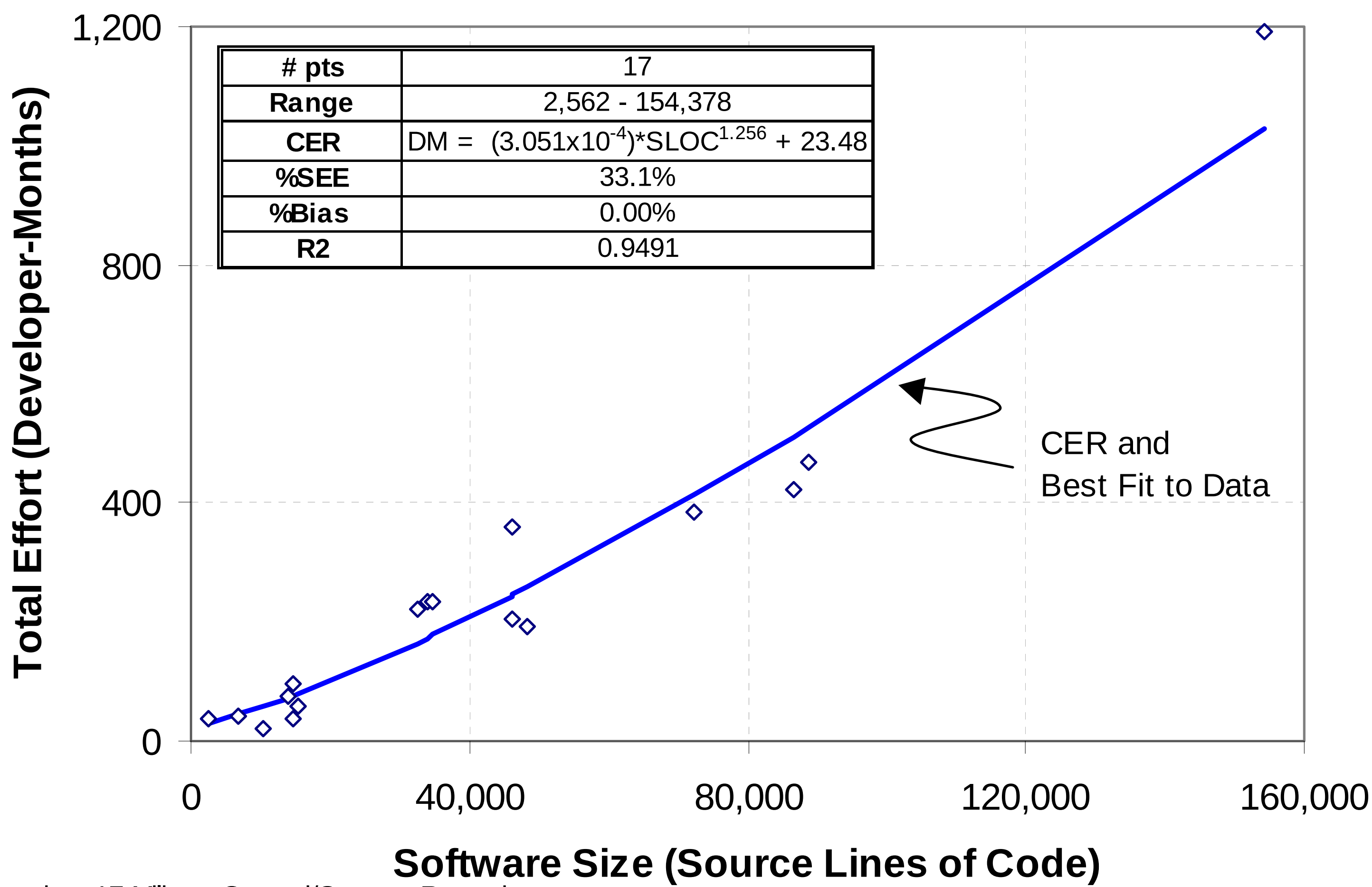
CER Results

Application Domain		Operating Environment			
		Military Ground	Military Mobile	Mil-Spec Avionics	Unmanned Space
Command & Control	# pts	9	30	2	10
	Range	6,000 - 54,400	5,000 - 111,256		13,298 - 155,077
	CER	$DM = (8.900x10^{-3})*SLOC - 12.21$	$DM = (1.351x10^{-3})*SLOC^{1.171}$		$DM = (1.687x10^{-2})*SLOC$
	%SEE	42.2%	28.2%		32.3%
	%Bias	0.00%	0.00%		0.0%
	R^2	0.7611	0.8261		0.9817
Database	# pts	3	5	4	0
	Range		11,123 - 38,519		
	CER		(Not provided; $R^2 < 0.1$)		
	%SEE		----		
	%Bias		----		
	R^2		< 0.1		
Mission Planning	# pts	4	13	1	1
	Range		5,169 - 77,693		
	CER		$DM = (6.990x10^{-3})*SLOC + 41.39$		
	%SEE		27.0%		
	%Bias		0.00%		
	R^2		0.8796		
O/S Exec	# pts	4	3	9	2
	Range			2,500 - 94,809	
	CER			$DM = (5.115x10^{-3})*SLOC + 172.9$	
	%SEE			51.0%	
	%Bias			0.00%	
	R^2			0.5597	
Signal Processing	# pts	9	21	2	8
	Range	13,530 - 143,026	4,270 - 140,925		2,200 - 66,700
	CER	$DM = (5.561x10^{-3})*SLOC + 21.91$	$DM = (9.354x10^{-3})*SLOC + 33.99$		$DM = (2.105x10^{-2})*SLOC - 38.65$
	%SEE	53.8%	34.3%		79.0%
	%Bias	0.00%	0.00%		0.00%
	R^2	0.6580	0.9375		0.8723
Simulation	# pts	5	0	0	3
	Range	11,100 - 140,200			
	CER	$DM = (3.766x10^{-6})*SLOC^{1.641} + 48.42$			
	%SEE	31.3%			
	%Bias	0.00%			
	R^2	0.9807			
Support	# pts	17	1	1	2
	Range	2,562 - 154,378			
	CER	$DM = (3.051x10^{-4})*SLOC^{1.256} + 23.48$			
	%SEE	33.1%			
	%Bias	0.00%			
	R2	0.9491			
Test	# pts	9	2	4	1
	Range	6,113 - 34,000			
	CER	$DM = (3.065x10^{-4})*SLOC^{1.326}$			
	%SEE	48.7%			
	%Bias	0.00%			
	R2	0.477			

CER & Associated Data

(Military Ground/Support)

# pts	17
Range	2,562 - 154,378
CER	DM = $(3.051\times10^{-4})*SLOC^{1.256} + 23.48$
%SEE	33.1%
%Bias	0.00%
R2	0.9491



Based on 17 Military Ground/Support Records

Topics of Presentation

- Background
- Data Building the Analysis Database
- Data Characterization
- Software Metrics
- Conclusion

Conclusions

- Median software productivity appears to have increased since 1996 study
  - Use of newer languages (e.g., C, C++) may be major factor
- Code growth continues to be an issue
  - Majority of records with this data show growth, one as much as 5x
- Data not available in several areas
  - COTS
    - Reason unknown
  - Labor effort in hours
    - Could be obtained if earned value management (EVM) data collected in dollars <u>and</u> hours
- Need to make sure SPOs are acquiring the right data now so future analyses can be performed