Homeland Security

Domain-Driven Software Cost, Schedule, and Phase Distribution Models: Using Software Resource Data Reports

Wilson Rosa (DHS)
Barry Boehm (USC)
Brad Clark (SEI-CMU)
Ray Madachy (NPS)
Joseph P Dean (AFCAA)

2013 ICEAA Professional Development & Training Workshop
June 19, 2013 (New Orleans, LA)

Introduction

- Instead of developing software cost and schedule estimation models with many parameters, this paper describes an analysis approach based on grouping similar software applications together called Productivity Types.
- Productivity types are groups of application domains that are environment independent, technology driven, and are characterized by 13 COCOMO product attributes.
- Consideration is also given to the operating environment it operates within.
- Over 200 actual software projects from DoD's Software Resource Data Reports (SRDRs) were fully inspected and analyzed to produce a comprehensive set of Cost Estimation Relationships, Schedule Estimation Relationships, and Software Productivity Benchmarks.

✓ Analysis results will be discussed in this presentation.

OUTLINE

- Research Method
- Data Demographics
- Software Productivity Benchmarks
- Effort and Schedule Estimation Models
- Conclusion
- Backup

Research Method

Instrumentation

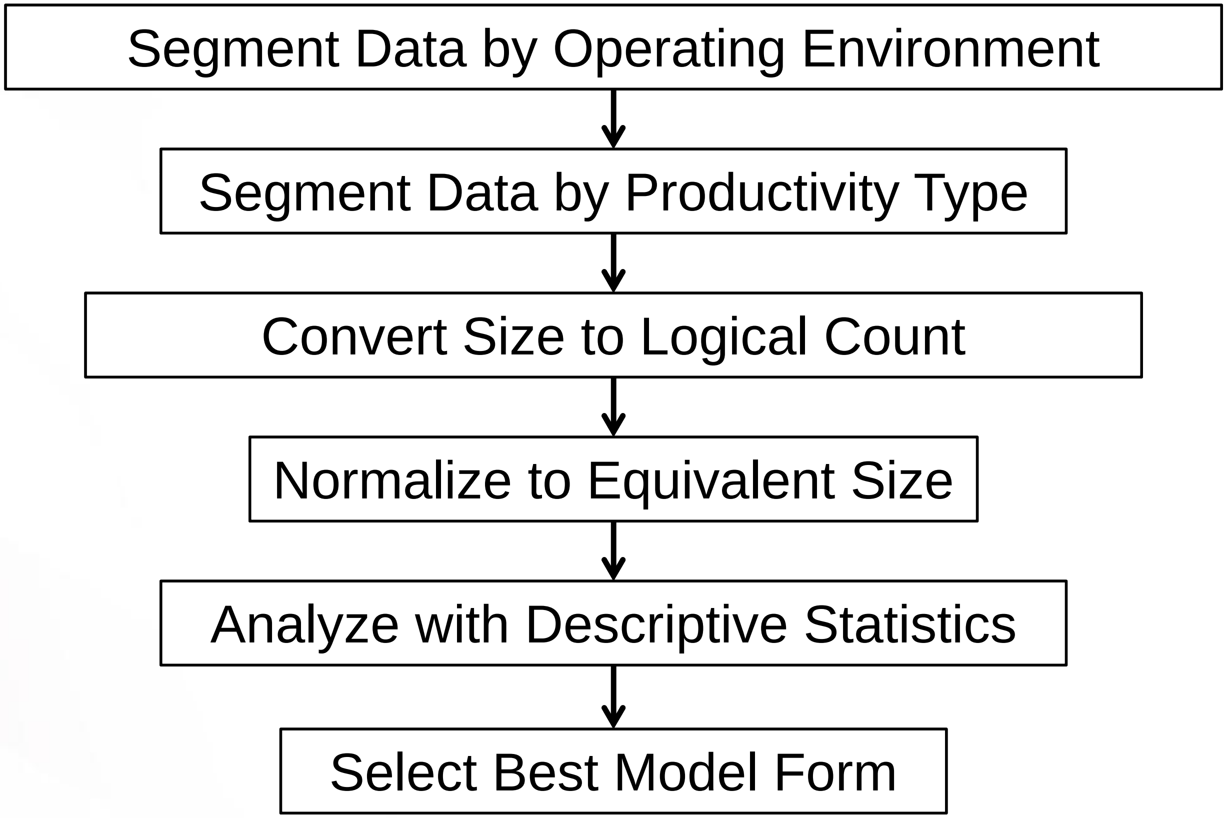
- Questionnaire:
  - Software Resource Data Report” (SRDR) (DD Form 2630)
- Source:
  - Defense Cost Analysis Resource Center (DCARC) website: http://dcarc.pae.osd.mil/Policy/csdrReporting.aspx
- Content:
  - Allows for the collection of project context, company information, requirements, product size, effort, schedule, and quality

Data Collection and Validation

- Initial Dataset
  - 800 completed software projects were collected from DCARC
  - Of the 800 projects, 345 were fully reviewed using GAO Best Practices
  - Of the 345 reviewed, 141 were excluded based on the following limitations:
    - Inadequate information on reused and modified code
    - Projects cancelled or terminated before delivery
    - Missing/Inaccurate effort and schedule data
    - Same duration (start and end dates) across software projects/components
    - Missing Adaptation and Adjustment Factors (DM, CM, IM)
    - Duplicate records or submissions
    - Estimates At Completion vice Actual Data
- Final Dataset
  - 204 of 345 projects included in the analysis as these passed quality inspection

Data Normalization and Analysis Workflow

- Data was normalized to "account for cost and sizing units, mission or application, technology maturity, and content so they are consistent for comparisons" (source: GAO)



Segment Data by Operating Environment

Segment Data by Productivity Type

Convert Size to Logical Count

Normalize to Equivalent Size

Analyze with Descriptive Statistics

Select Best Model Form

Segment Data by Operating Environment (OE)

Operating Environment		Acronym	Examples
Ground Site	Fixed	GSF	Command Post, Ground Operations Center, Ground Terminal, Test Faculties
	Mobile	GSM	Intelligence gathering stations mounted on vehicles, Mobile missile launcher
Ground Vehicle	Manned	GVM	Tanks, Howitzers, Personnel carrier
	Unmanned	GVU	Robots
Maritime Vessel	Manned	MVM	Aircraft carriers, destroyers, supply ships, submarines
	Unmanned	MVU	Mine hunting systems, Towed sonar array
Aerial Vehicle	Manned	AMV	Fixed-wing aircraft, Helicopters
	Unmanned	AVU	Remotely piloted air vehicles
Ordinance Vehicle	Unmanned	OVU	Air-to-air missiles, Air-to-ground missiles, Smart bombs, Strategic missiles
Space Vehicle	Manned	SVM	Passenger vehicle, Cargo vehicle, Space station
	Unmanned	SVU	Orbiting satellites (weather, communications), Exploratory space vehicles





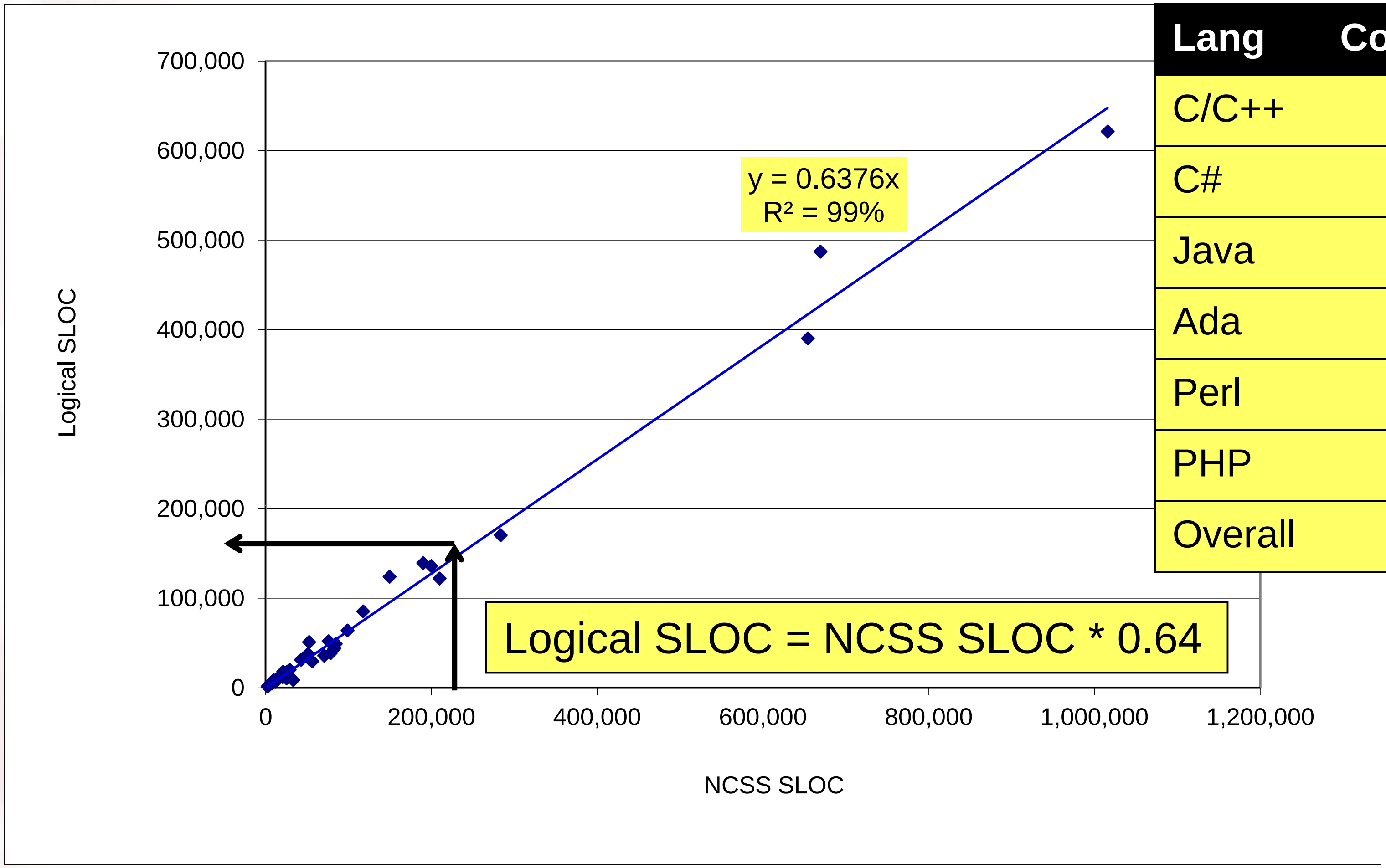
Segment Data by Productivity Type (PT)

- Different productivities have been observed for different software application types.
- SRDR dataset was segmented into 14 productivity types to increase the accuracy of estimating cost and schedule

1. Sensor Control and Signal Processing (SCP)
2. Vehicle Control (VC)
3. Real Time Embedded (RTE)
4. Vehicle Payload (VP)
5. Mission Processing (MP)
6. System Software (SS)
7. Telecommunications (TEL)
8. Process Control (PC)
9. Scientific Systems (SCI)
10. Planning Systems (PLN)
11. Training (TRN)
12. Test Software (TST)
13. Software Tools (TUL)
14. Intelligence & Information Systems (IIS)

See Slides 35 & 36 for productivity type definitions and examples

Convert Size (NCSS) to Logical Count

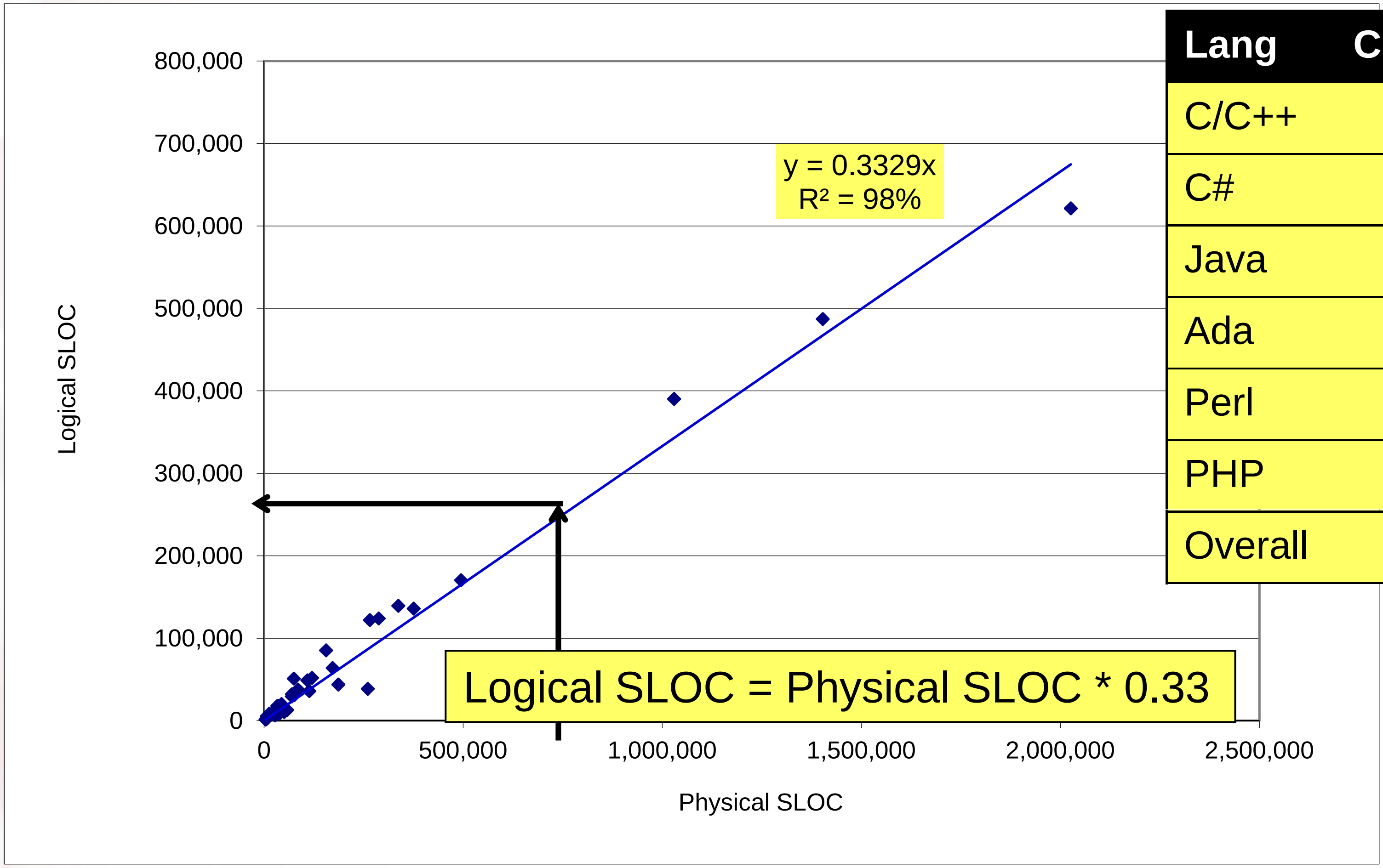


Logical SLOC = NCSS SLOC * 0.64

Lang	Conversion
C/C++	0.61
C#	0.61
Java	0.72
Ada	0.52
Perl	0.70
PHP	0.66
Overall	0.64

Rosa, Boehm, Clark and Madachy, “ Domain-Driven Software Cost Estimation”, 27th International Forum on COCOMO® and Systems/Software Cost Modeling, University of Southern California, Oct 2012 (USA)

Convert Size (Physical) to Logical Count



Logical SLOC = Physical SLOC * 0.33

Lang	**Conversion**
C/C++	0.32
C#	0.35
Java	0.52
Ada	0.25
Perl	0.25
PHP	0.44
Overall	0.33

Rosa, Boehm, Clark and Madachy, “ Domain-Driven Software Cost Estimation”, 27th International Forum on COCOMO® and Systems/Software Cost Modeling, University of Southern California, Oct 2012 (USA)

Normalize to Equivalent Size

- **Logical SLOC normalized to Equivalent SLOC (ESLOC) to reflect the actual degree of work involved:**

Formula:

$$\text{ESLOC} = \text{New SLOC} + \text{Modified SLOC}*\mathbf{AAF_M} + \text{Reused SLOC}*\mathbf{AAF_R} + \text{Generated SLOC}*\mathbf{AAF_G} + \text{Converted SLOC}*\mathbf{AAF_C}$$

Where:

$$\text{AAF}_i = 0.4*\text{DM} + 0.3*\text{CM} + 0.3*\text{IM}$$

And:

AAF	=	Adaptation Adjustment Factor
i	=	Refers to the size type: Modified (M), Reuse (N), Generated (R), Converted (C)
DM	=	Design Modified (DM), also known as re-design
CM	=	Code Modified (CM), also known as re-code
IM	=	Integration Modified (IM), also known as re-test

- ✓ Formula adapted from COCOMO II Reuse Model
- ✓ Model Input Parameters (DM, CM, IM) provided by Data Sources (System Developers)

Analyze the Data

- Data is analyzed using the following taxonomy

Operating Environment

Productivity Type	GSF	GVM	MVM	AVM	AVU	OVU	**Total**
SCP	13		3	9	8	3	36
VC		12		9		3	24
RTE	16	**10**	6	17		3	52
MP	16			12			28
SYS	10		1	2			13
SCI	13			6			19
PLN	11						11
TEL	3						3
IIS	8		2				10
Total	98	24	10	55	8	9	204

When the dataset is segmented by Productivity Type and Operating Environment, the impact accounted for by many COCOMO II model drivers are considered

Select Best Model Form (Measures)

- Accuracy of the Models verified using seven different measures:

Measure	Symbol	Description
Standard Error	SEE	Standard Error of the Estimate is a measure of the difference between the observed and CER estimated effort. The SEE is to linear models as the standard deviation is to a sample mean.
Coefficient of Variation	CV	Percentage expression of the standard error compared to the mean of dependent variable. A relative measure allowing direct comparison among models.
Mean Absolute Deviation	MAD	Measures the average percentage by which the regression overestimates or underestimates the observed actual value. Mitigates against the “cancellation” effect from the sign and magnitude of a single % error.
Prediction Accuracy	PRED (L)	PRED (L): Prediction accuracy is the percentage of CER estimates that are within L percentage of the actual effort observations. L is commonly set to 30%.
Variance Inflation Factor	VIF	Indicates whether multicollinearity (correlation among predictors) was present in a multi-regression analysis. Multicollinearity is problematic because it can increase the variance of the regression coefficients, making them unstable and difficult to interpret.
Coefficient of Determination	R^2	The Coefficient of Determination shows how much variation in dependent variable is explained by the regression equation. Not applicable for Non-Linear regression.
F-test	F-test	The value of the F test is the square of the equivalent t test; the bigger it is, the smaller the probability that the difference could occur by chance. Not applicable for Non-Linear regression.

Select Best Effort Model Form (Rules of Thumb)

- Three effort model forms were examined for each dataset

$$PM = A*Size^{B}$$

Log-Linear Model

$$PM = C + A*Size^{B}$$

Non-Linear Model 1

$$PM = C + Size^{B}$$

Non-Linear Model 2

Where

PM	=	Software development effort (in Person-months)
Size	=	Size in Thousand Equivalent Source Lines of Code (KESLOC)
A	=	Calibrated Productivity constant (ESLOC/PM)
B	=	B-exponent (Normally greater than one, indicating diseconomies of scale)
C	=	Fixed level of effort support activities (in Person-Months)

- Rules of Thumb for Selecting Best Model

Measure	Rules of Thumb
# Observations	**> 7**
B-Exponent	**> 1.0**
CV	< 50%
MAD	< 50%
PRED (30)	> 40%
R^2	> 60%

Select Best Schedule Model Form (Rules of Thumb)

- Two schedule model forms were examined for each dataset

$$TDEV = A * PM^{F}$$

COCOMO 81 Model

$$TDEV = A * Size^{B} * FTE^{C}$$

Non-Linear Model

Where

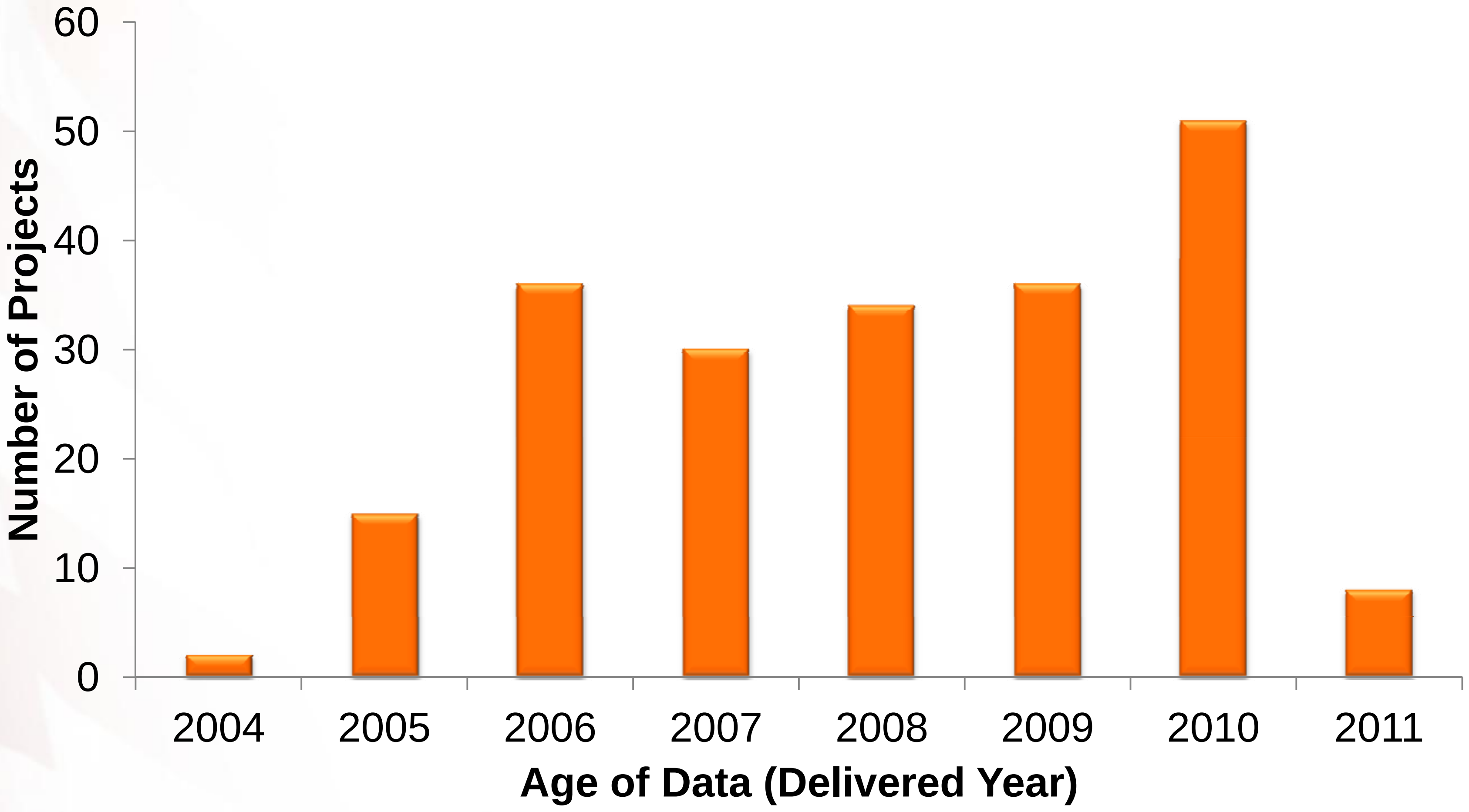
TDEV	=	Time (in months) to develop the Software Product
Size	=	Software Size in Equivalent Source Lines of Code (ESLOC)
FTE	=	Full Time Equivalent (FTE) Staffing Levels
PM	=	Total Estimated Effort in Person-Months (PM)
A	=	is a duration constant
B	=	Scaling factor to account for changing productivity as size increases,
C	=	C-Scaling Factor accounts for the non-linear relationship between increasing staffing levels and shortening development time, TDEV
F	=	Scaling factor for effort changes

- Rules of Thumb for Selecting Best Model

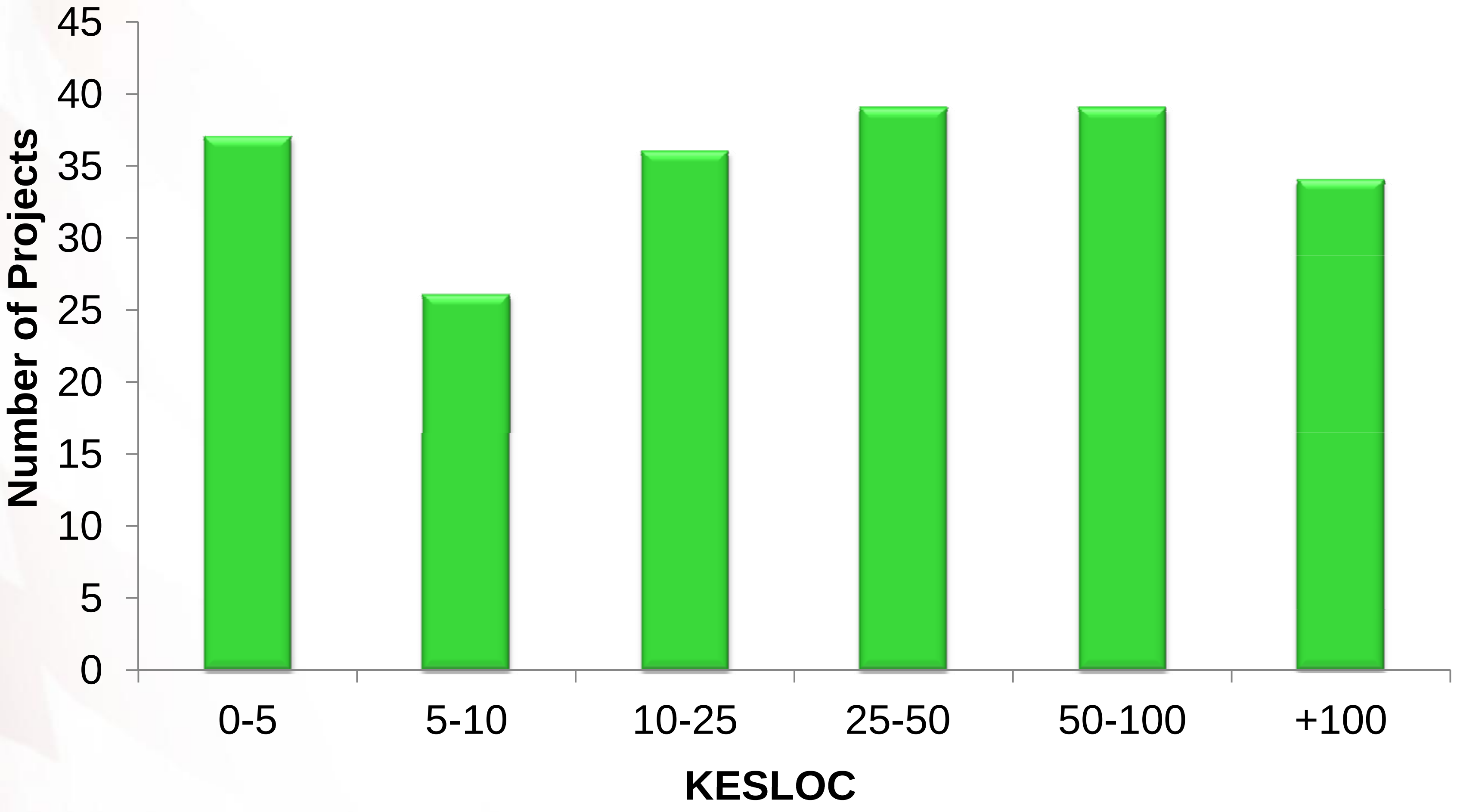
Measure	**Rules of Thumb**
# Observations	**> 10**
C-Scaling Factor	**< 0.0**
MAD	< 50%
PRED (30)	> 40%

DATA DEMOGRAPHICS

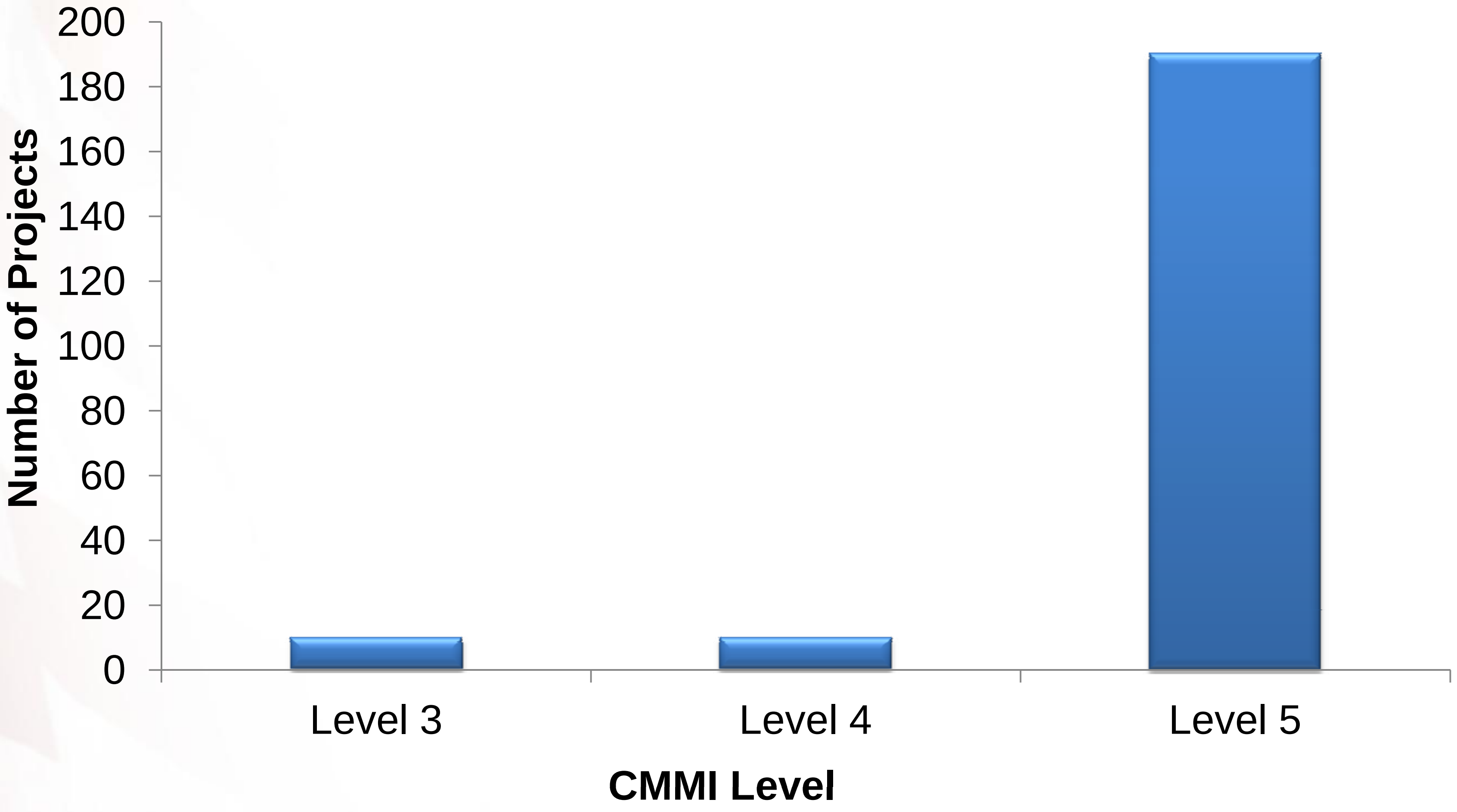
Age of Data


Number of Projects
60
50
40
30
20
10
0
2004
2005
2006
2007
2008
2009
2010
2011
Age of Data (Delivered Year)

Software Size

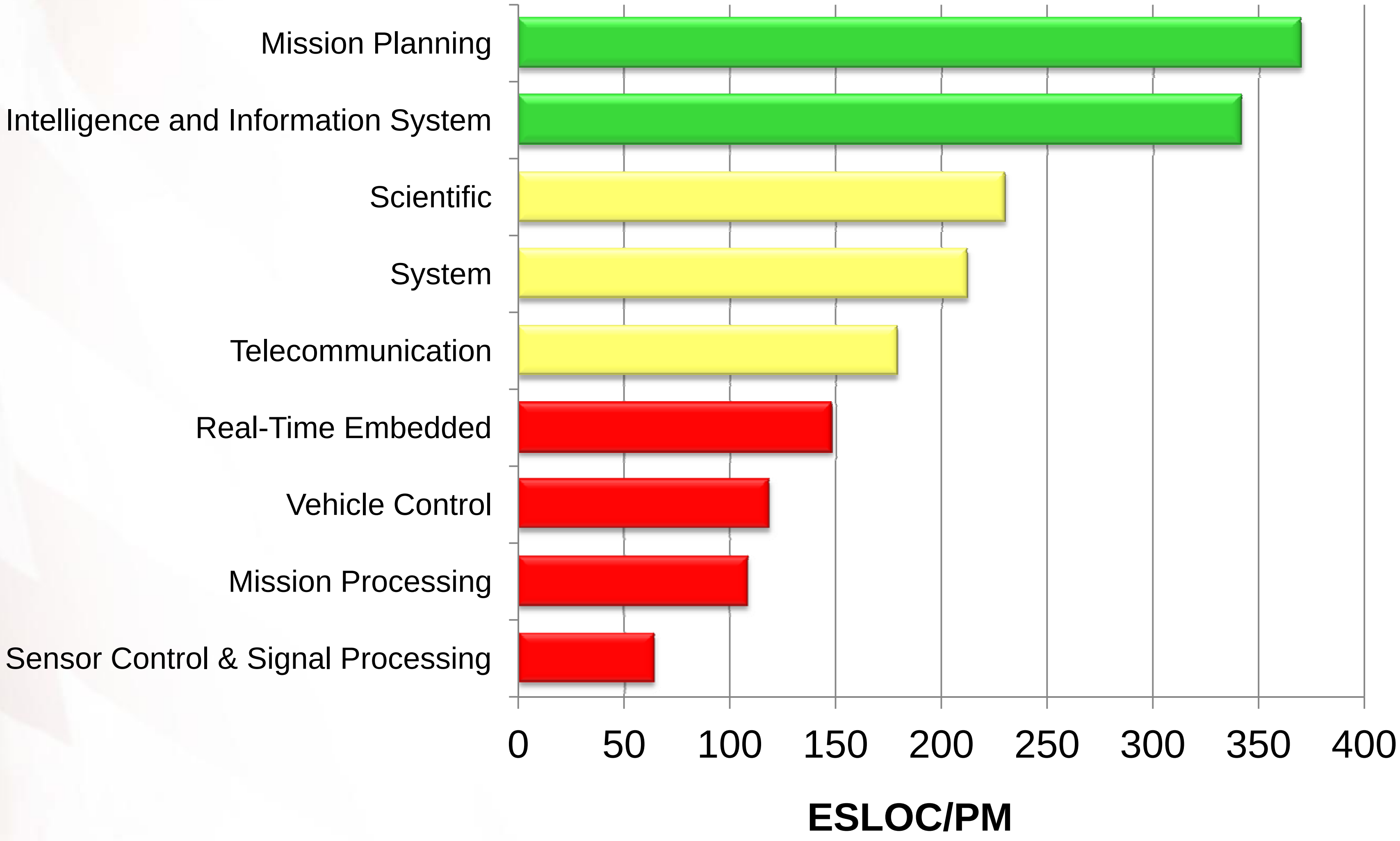


Capability Maturity Model Integration (CMMI) Level



Homeland Security

Productivity Range (Median)



SOFTWARE PRODUCTIVITY BENCHMARKS

Software Productivity Benchmarks

- Software productivity refers to the ability of an organization to generate outputs using the resources that it currently has as inputs. Inputs typically include facilities, people, experience, processes, equipment, and tools. Outputs generated include software applications and documentation used to describe them.

$$\mathrm{PROD} = \frac{ESLOC}{PM}$$

- Metric used to express software productivity is equivalent source lines of code (ESLOC) per person-month (PM) of effort. While many other measures exist, ESLOC/PM will be used because most of the data collected by the Department of Defense (DoD) on past projects is captured using these two measures. While controversy exists over whether or not ESLOC/PM is a good measure, consistent use of this metric provides for meaningful comparisons of productivity.

Software Productivity Benchmarks (ALL)

Productivity Benchmarks by PT, across Operating Environments

PT	OE	ESLOC/PM				Obs.	Std. Dev.	CV	KESLOC	
		MEAN	1st Quartile	Median	3rd Quartile				MIN	MAX
SCP	ALL	71	34	64	83	36	53	74%	0.8	221
MP	ALL	143	86	109	187	28	84	59%	1	225
VC	ALL	147	94	119	184	25	91	62%	1.4	189
RTE	ALL	141	59	148	185	52	87	61%	1	449
TEL	ALL	196	132	179	265	11	110	56%	1	312
SYS	ALL	325	130	212	310	13	347	**107%**	6.8	475
SCI	ALL	267	94	230	351	15	218	82%	1.8	218
IIS	ALL	365	306	342	419	10	72	20%	2	417
PLN	ALL	419	324	370	546	11	164	39%	29	310

Software Productivity Benchmarks (GSF)

Productivity Benchmarks by PT, Ground Site Fixed (GSF)

PT	OE	ESLOC/PM				Obs.	Std. Dev.	CV	KESLOC	
		MEAN	1st Quartile	Median	3rd Quartile				MIN	MAX
SCP	GSF	64	30	67	82	13	1	48%	1	72
MP	GSF	163	98	133	223	14	86	53%	5	225
SCI	GSF	192	129	164	246	12	102	53%	5	125
RTE	GSF	174	140	168	199	16	61	35%	2	87
TEL	GSF	196	132	179	265	11	110	56%	1	312
SYS	GSF	325	155	237	304	10	98	**105%**	1	475
PLN	GSF	419	324	370	546	11	164	39%	29	310
IIS	GSF	379	313	372	433	8	74	20%	2	417

Software Productivity Benchmarks (GSF)

Productivity Benchmarks by PT, Aerial Vehicle Manned

PT	OE	ESLOC/PM				Obs.	Std. Dev.	CV	KESLOC	
		MEAN	1st Quartile	Median	3rd Quartile				MIN	MAX
SCP	AVM	95	46	74	115	9	80	85%	2	107
MP	AVM	128	67	106	173	12	84	66%	1	201
VC	AVM	173	118	157	214	9	71	41%	1	87
RTE	AVM	180	135	170	188	17	63	35%	5	132

Effort and Schedule Estimation Models

Effort Estimation Models, ALL OE

Effort Estimation Models by PT, All Operating Environment (OE)

PT	OE	Model Form	Obs.	R^2 (%)	MAD (%)	CV (%)	PRED (30)	KESLOC	
								MIN	MAX
IIS	ALL	PM= 3.102 * KESLOC ^ **0.9713**	10	98	16	19	80%	2	417
MP	ALL	PM= 9.229 * KESLOC ^ **1.019**	28	68	45	44	43%	1	225
PLN	ALL	PM = 30.61 + KESLOC ^ **1.165**	11	***	35	41	60%	29	310
SYS	ALL	PM= 61.13 + 2.306 * KESLOC ^ **1.089**	13	***	47	36	54%	6.8	475
VC	ALL	PM= 7.836 * KESLOC ^ **1.002**	25	86	48	**57**	52%	1.4	189

Effort Estimation Models, GSF

Effort Estimation Models by PT, Ground Site Fixed

PT	OE	Model Form	Obs.	R^2 (%)	MAD (%)	CV (%)	PRED (30)	KESLOC	
								MIN	MAX
IIS	GSF	PM = 3.081 * KESLOC ^ **0.9619**	8	99	15	17	88%	2	417
MP	GSF	PM = 5.617 * KESLOC ^ **1.085**	14	77	39	37	50%	5	225
PLN	GSF	PM = 30.61 + KESLOC ^ **1.165**	11	***	35	41	60%	29	310
RTE	GSF	PM = 23.62 + KESLOC ^ **1.433**	16	**	35	26	56%	2	87
SCI	GSF	PM = 128.7 + KESLOC ^ **1.223**	12	**	36	30	67%	5	125
SYS	GSF	PM = 49.54 + KESLOC ^ **1.264**	10	**	39	45	50%	1	475

Homeland Security

Effort Estimation Models, AVM

Effort Estimation Models by PT, Aerial Vehicle Manned (AVM)

PT	OE	Model Form	Obs.	R^2 (%)	MAD (%)	CV (%)	PRED (30)	KESLOC	
								MIN	MAX
MP	AVM	PM = 4.364 + 5.398 * KESLOC ^ **1.194**	12	**	46	49	42%	1	201
RTE	AVM	PM = 29.19 + KESLOC ^ **1.439**	17	**	29	39	48%	5	132
SCP	AVM	PM = 23.85 + 3.89 * KESLOC ^ **1.402**	9	**	39	38	40%	2	107
VC	AVM	PM = 4.613 * KESLOC ^ **1.111**	9	97	28	23	78%	1	87

Effort Estimation Models, GVM

Effort Estimation Models by PT, Ground Vehicle Manned (GVM)

PT	OE	Model Form	Obs.	R^2 (%)	MAD (%)	CV (%)	PRED (30)	KESLOC MIN	KESLOC MAX
VC	GVM	PM = 6.751 * KESLOC ^ **1.153**	12	76	26	29	67%	1.4	39

Schedule Estimation Models, ALL OE

Schedule Estimation Models by PT, All Operating Environment (OE)

PT	OE	Model Form	Obs.	SE (%)	MAD (%)	CV (%)	PRED (30)	KESLOC	
								MIN	MAX
IIS	ALL	TDEV = 4.122 * KESLOC ^ **0.8447** * FTE ^ **(-0.8992)**	10	20	15	17	90%	2	417
PLN	ALL	TDEV = 5.468 * KESLOC ^ **0.5876** * FTE ^ **(-0.3771)**	10	37	24	29	60%	36	310
SCI	ALL	TDEV = 19.11 * KESLOC ^ **0.2123** * FTE ^ **(-0.2572)**	16	36	25	20	63%	1.8	218
MP	ALL	TDEV = 12.83 * KESLOC ^ **0.5471** * FTE ^ **(-0.4314)**	24	39	30	29	58%	1	225
VC	ALL	TDEV = 7.059 * KESLOC ^ **0.8331** * FTE ^ **(-0.7281)**	15	34	26	25	69%	1.4	189
RTE	ALL	TDEV = 31.28 * KESLOC ^ **0.06865** * FTE ^ **(-0.0761)**	35	31	23	20	80%	.3	100

Conclusion

- Regression analyses in this presentation indicate that Size and Productivity Type are valid predictors of software development effort for the time period beginning with program initiation up to the point of development test & evaluation.
- Variation in software development effort becomes more significant when dataset is grouped by Operating Environment. Thus, the effect of size on software development effort shall be interpreted along with Productivity Type and Operating Environment.
- Extending the analysis to schedule estimation, Productivity Type again was shown as a valid predictor of duration when used in combination with staffing levels (full time equivalents) and software size (ESLOC)
- Schedule analysis also shows that software development duration can be shortened by decreasing scope (Size) and/or increasing staffing levels. In contrast, duration can be lengthened by increasing scope and/or decreasing staffing levels.

BACKUP

Productivity Type Definitions (1 of 2)

TYPE	DESCRIPTION
Sensor Control and Signal Processing (SCP)	Software that requires timing-dependent device coding to enhance, transform, filter, convert, or compress data signals. Examples: Beam steering controller, sensor receiver/transmitter control, sensor signal processing, sensor receiver/transmitter test. Examples. of sensors: antennas, lasers, radar, sonar, acoustic, electromagnetic.
Vehicle Control (VC)	Hardware & software necessary for the control of vehicle primary and secondary mechanical devices and surfaces. Examples: Digital Flight Control, Operational Flight Programs, Fly-By-Wire Flight Control System, Flight Software, Executive.
Vehicle Payload (VP)	Hardware & software which controls and monitors vehicle payloads and provides communications to other vehicle subsystems and payloads. Examples: Weapons delivery and control, Fire Control, Airborne Electronic Attack subsystem controller, Stores and Self-Defense program, Mine Warfare Mission Package.
Real Time Embedded (RTE)	Real-time data processing unit responsible for directing and processing sensor input/output. Examples: Devices such as Radio, Navigation, Guidance, Identification, Communication, Controls And Displays, Data Links, Safety, Target Data Extractor, Digital Measurement Receiver, Sensor Analysis, Flight Termination, Surveillance, Electronic Countermeasures, Terrain Awareness And Warning, Telemetry, Remote Control.
Mission Processing (MP)	Vehicle onboard master data processing unit(s) responsible for coordinating and directing the major mission systems. Examples: Mission Computer Processing, Avionics, Data Formatting, Air Vehicle Software, Launcher Software, Tactical Data Systems, Data Control And Distribution, Mission Processing, Emergency Systems, Launch and Recovery System, Environmental Control System, Anchoring, Mooring and Towing.
Process Control (PC)	Software that manages the planning, scheduling and execution of a system based on inputs, generally sensor driven.
System Software (SYS)	Layers of software that sit between the computing platform and applications. Examples: Health Management, Link 16, Information Assurance, Framework, Operating System Augmentation, Middleware, Operating Systems

Productivity Type Definitions (2 of 2)

TYPE	DESCRIPTION
Training (TRN)	Hardware and software that are used for educational and training purposes. Examples: Onboard or Deliverable Training Equipment & Software, Computer-Based Training
Telecommunications (TEL)	The transmission of information, e.g. voice, data, commands, images, and video across different mediums and distances. Primarily software systems that control or manage transmitters, receivers and communications channels. Examples: switches, routers, integrated circuits, multiplexing, encryption, broadcasting, protocols, transfer modes, etc.
Software Tools (TOOL)	Software that is used for analysis, design, construction, or testing of computer programs. Examples: Integrated collection of tools for most development phases of the life cycle, e.g. Rational development environment
Test Software (TST)	Hardware & Software necessary to operate and maintain systems and subsystems which are not consumed during the testing phase and are not allocated to a specific phase of testing. Examples: Onboard or Deliverable Test Equipment & Software
Intelligence & Information Systems (IIS)	An assembly of software applications that allows a properly designated authority to exercise control over the accomplishment of the mission. Humans manage a dynamic situation and respond to user-input in real time to facilitate coordination and cooperation. Software that manipulates, transports and stores information. Examples: Database, Data Distribution, Information Processing, Internet, Entertainment, Enterprise Services*, Enterprise Information**
Scientific Systems (SCI)	Non real time software that involves significant computations and scientific analysis. Examples: Environment Simulations, Offline Data Analysis, Vehicle Control Simulators
Training (TRN)	Hardware and software that are used for educational and training purposes. Examples: Onboard or Deliverable Training Equipment & Software, Computer-Based Training