Journal of Cost Analysis and Parametrics, 4:127–147, 2011

ISSN: 1941-658X print
DOI: 10.1080/1941658X.2011.629494

A Probabilistic Method for Predicting Software Code Growth

MICHAEL A. ROSS

Tecolote Research, Inc., Manhattan Beach, California

A significant challenge that many cost analysts and project managers face is predicting by how much their initial estimates of software development cost and schedule will change over the lifecycle of the project. Examination of currently-accepted software cost, schedule, and defect estimation algorithms reveals a common acknowledgment that estimated software size is the single most influential independent variable. Unfortunately, the most important business decisions about a software project are made at its beginning, the time when most estimating is done, and coincidently the time of minimum knowledge, maximum uncertainty, and hysterical optimism. This article describes a model and methodology that provides probabilistic growth adjustment to single-point Technical Baseline Estimates of Delivered Source Lines of Code, for both new software and pre-existing reused software that is sensitive to the maturity of their single-point estimates. The model is based on Software Resources Data Report data collected by the U.S. Air Force and has been used as part of the basis for several USAF program office estimates and independent cost estimates. It provides an alternative to other software code growth methodologies, such as Holchin's and Jensen's code growth matrices.

Introduction

The Tecolote DSLOC Estimate Growth Model v06 (DEGM6) provides probabilistic growth adjustments to single-point Technical Baseline Estimates (TBEs) of Delivered Source Lines of Code (DSLOC), for both New software and Pre-Existing Reused (PER) software, that are sensitive to the *maturity* of the DSLOC TBEs; i.e., when, in the Software Development Life Cycle (SDLC), the DSLOC TBE is performed. It is a *data-driven* model and methodology that is based on Software Resources Data Report (SRDR) data collected by the U.S. Air Force Cost Analysis Agency (AFCAA) (Rosa, 2008). This model provides an alternative to other software code growth methodologies, such as Holchin's (2003) and Jensen's (2008) code growth matrices.

This article includes custom Cumulative Distribution Function (CDF) tables that can be copied into tools, such as ACEIT or Crystal Ball, in order to construct Custom CDFs[1] that are needed to model the baseline New DSLOC growth factor distribution and to model the baseline PER DSLOC growth factor distribution. This article also includes a set of DSLOC growth factor multipliers as a function of estimate maturity (EM) for each of New DSLOC and PER DSLOC such that appropriate application of these factors to a DSLOC TBE yields corresponding Least, Likely, and Most DSLOC values that, if input to SEER-SEM, will reasonably model growth and uncertainty consistent with SRDR historical data.

Address correspondence to Michael A. Ross, Tecolote Research, Inc., 3601 Aviation Blvd., Suite 1600, Manhattan Beach, CA 90266-3756. E-mail: mross@tecolote.com

Model Summary

The DEGM6 equations for applying growth and uncertainty to TBE New and PER DSLOC are[2]

$$\boldsymbol{S_{DAdjNew}} \equiv S_{DNew}\left(e^{-bt}\left(\boldsymbol{K_{GFNew}} - 1\right) + 1\right) \tag{1}$$

and

$$\boldsymbol{S_{DAdjPER}} \equiv S_{DNew}\left(e^{-bt}\left(\boldsymbol{K_{GFPER}} - 1\right) + 1\right), \tag{2}$$

where

$\boldsymbol{S_{DAdjNew}} \equiv$ Growth-adjusted New DSLOC estimate distribution;
$\boldsymbol{S_{DAdjPER}} \equiv$ Growth-adjusted PER DSLOC estimate distribution;
$S_{DNew} \equiv$ Technical Baseline Estimate (TBE) of New DSLOC;
$S_{DPER} \equiv$ Technical Baseline Estimate (TBE) of PER DSLOC;
$\boldsymbol{K_{GFNew}} \equiv$ Baseline (assuming Estimate Maturity = 0%) New DSLOC growth factor distribution (see Custom CDF in Table 3);
$\boldsymbol{K_{GFPER}} \equiv$ Baseline (assuming Estimate Maturity = 0%) PER DSLOC growth factor distribution (see Custom CDF in Table 3);
$b \equiv$ Decay constant; default is 3.466 based on Boehm's (1981 pp. 310–311) *Cone of Uncertainty*;
$t \equiv$ Estimate Maturity Parameter: (SDLCBegin = 0%; SyRR = 20%; SwRR = 40%; SwPDR = 60%; SwCDR = 80%; SwAccept = 100%).

The equations for providing the appropriate New and PER $\langle Least, Likely, Most\rangle$ DSLOC inputs to SEER-SEM are:

Growth-Adjusted New DSLOC	Growth-Adjusted PER DSLOC	
$S_{DAdjNewLeast} = S_{DNew}\left(-0.828071e^{-3.466t} + 1\right)$	$S_{DAdjPERLeast} = S_{DPER}\left(-0.687191e^{-3.466t} + 1\right)$.	(3)
$S_{DAdjNewLikely} = S_{DNew}\left(-0.828071e^{-3.466t} + 1\right)$	$S_{DAdjPERLikely} = S_{DPER}\left(-0.687192e^{-3.466t} + 1\right)$	
$S_{DAdjNewMost} = S_{DNew}\left(5.366128e^{-3.466t} + 1\right)$	$S_{DAdjPERMost} = S_{DPER}\left(3.658219e^{-3.466t} + 1\right)$	

The remainder of this article describes the basis of these equations.

Components of the Model

Normalized Estimate Maturity

The single parameter input to the DEGM6 is normalized EM t. By default, EM is quantified by the scale contained in Table 1. This scale is consistent with the model defaults for the baseline New and PER DSLOC growth factor distributions, which is based on SRDR data, and with the uncertainty decay factor, which is based on Boehm's *Cone of Uncertainty*. Tailored instances of the model can be created for different SDLCs as long as historical data exist where the projects followed that particular SDLC and where these data have been used to determine corresponding baseline growth factor distributions and uncertainty decay factor values or distributions.

TABLE 1 Default normalized estimate maturity scale

Estimate maturity scale	
t = 0%	Begin SDLC
t = 20%	System Requirements Review
t = 40%	System Design Review / Software Requirements Review
t = 60%	Software Preliminary Design Review
t = 80%	Software Critical Design Review
t = 100%	Software Acceptance

DSLOC Baseline Growth Factor Distributions

DSLOC estimate growth is modeled at the computer program (CSCI) level and is applied by multiplying the TBEs of New and PER DSLOC by the appropriate decay-adjusted growth factor distribution. The baseline (zero EM) growth factor distributions for New DSLOC and for Pre-Existing DSLOC have the following characteristics (Table 2) and Custom CDFs (Table 3).

TABLE 2 SRDR data set distribution statistics

ACE DSLOC baseline growth factor distribution statistics			
New DSLOC growth factor		Pre-existing DSLOC growth factor	
Number of Data Points (N)	56	*Number of Data Points (N)*	45
Data Set Mean (m)	1.75	*Data Set Mean (m)*	1.43
CDF Mean (m′)	1.75	*CDF Mean (m′)*	1.42
%ile @ Data Set Mean (P(m))	69%	*%ile @ Data Set Mean (P(m))*	71%
%ile @ CDF Mean (P(m′))	69%	*%ile @ CDF Mean (P(m′))*	71%
%ile @ Point (P(pt))	29%	*%ile @ Point (P(pt))*	29%
Data Set Median m[∼]	1.20	*Data Set Median m[∼]*	1.04
CDF Median m′[∼]	1.204296	*CDF Median m′[∼]*	1.037044
Define a baseline growth factor distribution in ACE by using this value as the "Equation / Throughput" field entry with a custom CDF containing corresponding median-normalized growth factor values.		*Define a baseline growth factor distribution in ACE by using this value as the "Equation / Throughput" field entry with a custom CDF containing corresponding median-normalized growth factor values.*	
Data Set Std Dev s	1.33	*Data Set Std Dev s*	0.91
CDF Std Dev s′	1.32	*CDF Std Dev s′*	0.90
Data Set CV (C[V])	0.76	*Data Set CV (C[V])*	0.64
CDF CV (C′[V])	0.75	*CDF CV (C′[V])*	0.63

TABLE 3 DSLOC estimate growth factor distribution CDFs

ACE DSLOC baseline growth factor distribution CDFs					
Copy shaded columns into ACEIT custom CDF dialog box			Copy shaded columns into ACEIT custom CDF dialog box		
New DSLOC growth factor CDF			Pre-existing DSLOC growth factor CDF		
%ile	Raw growth factor	Median-normalized growth factor	%ile	Raw growth factor	Median-normalized growth factor
0.0	0.547902	0.4549560272208	0.0	0.655131	0.6317293787416
1.0	0.551141	0.4576460806781	1.0	0.655131	0.6317293787416
2.0	0.581378	0.4827532462799	2.0	0.660043	0.6364662585323
3.0	0.608058	0.5049076955001	3.0	0.665570	0.6417952482967
4.0	0.627232	0.5208286323592	4.0	0.683037	0.6586381727100
5.0	0.636229	0.5282996372516	5.0	0.706474	0.6812380644476
6.0	0.636407	0.5284473677728	6.0	0.720040	0.6943196660333
7.0	0.642677	0.5336535369111	7.0	0.721267	0.6955034049290
8.0	0.650977	0.5405458522550	8.0	0.722519	0.6967099061808
9.0	0.664670	0.5519163885260	9.0	0.723852	0.6979960756790
10.0	0.676993	0.5621483902387	10.0	0.725186	0.6992822451771
11.0	0.682089	0.5663801768247	11.0	0.782870	0.7549050972897
12.0	0.689030	0.5721433207804	12.0	0.840553	0.8105279494022
13.0	0.698820	0.5802731079439	13.0	0.855829	0.8252587074461
14.0	0.747107	0.6203681800052	14.0	0.858990	0.8283060100418
15.0	0.820302	0.6811466717064	15.0	0.877333	0.8459940234183
16.0	0.834355	0.6928160959576	16.0	0.907822	0.8753946054196
17.0	0.837741	0.6956274919723	17.0	0.930109	0.8968845711522
18.0	0.900236	0.7475209997647	18.0	0.935987	0.9025533043474
19.0	0.951335	0.7899511413624	19.0	0.941866	0.9082220375426
20.0	0.968243	0.8039911843758	20.0	0.947745	0.9138907707378
21.0	0.980545	0.8142062798349	21.0	0.953623	0.9195595039331
22.0	0.987532	0.8200079742699	22.0	0.959502	0.9252282371283
23.0	0.990888	0.8227943734626	23.0	0.965381	0.9308969703235
24.0	0.992523	0.8241524749089	24.0	0.971260	0.9365657035187
25.0	0.994159	0.8255105763553	25.0	0.977138	0.9422344367139
26.0	0.995794	0.8268686778016	26.0	0.983017	0.9479031699091
27.0	0.997430	0.8282267792480	27.0	0.988896	0.9535719031044
28.0	0.999065	0.8295848806943	28.0	0.994774	0.9592406362996
29.0	1.000455	0.8307384829524	29.0	1.000001	0.9642804324725
30.0	1.001516	0.8316194196262	30.0	1.000010	0.9642887324668
31.0	1.002576	0.8325003563000	31.0	1.000018	0.9642970324612
32.0	1.003637	0.8333812929738	32.0	1.000027	0.9643053324555
33.0	1.004698	0.8342622296476	33.0	1.000035	0.9643136324499
34.0	1.005759	0.8351431663214	34.0	1.000044	0.9643219324442

(*Continued*)

TABLE 3 (Continued)

ACE DSLOC baseline growth factor distribution CDFs					
Copy shaded columns into ACEIT custom CDF dialog box			Copy shaded columns into ACEIT custom CDF dialog box		
New DSLOC growth factor CDF			Pre-existing DSLOC growth factor CDF		
%ile	Raw growth factor	Median-normalized growth factor	%ile	Raw growth factor	Median-normalized growth factor
35.0	1.006934	0.8361189505898	35.0	1.000053	0.9643302324386
36.0	1.008635	0.8375310337930	36.0	1.000061	0.9643385324329
37.0	1.019294	0.8463820771439	37.0	1.000070	0.9643468324273
38.0	1.043799	0.8667296952683	38.0	1.000078	0.9643551324216
39.0	1.056488	0.8772661203276	39.0	1.000087	0.9643634324160
40.0	1.061531	0.8814541263447	40.0	1.000096	0.9643717324103
41.0	1.077386	0.8946191944000	41.0	1.000104	0.9643800324047
42.0	1.095158	0.9093763078095	42.0	1.008341	0.9723224006821
43.0	1.101241	0.9144271946891	43.0	1.017606	0.9812565274949
44.0	1.110578	0.9221809075650	44.0	1.022750	0.9862163911111
45.0	1.129681	0.9380430984295	45.0	1.025832	0.9891891231291
46.0	1.146794	0.9522532535966	46.0	1.028802	0.9920524918041
47.0	1.161612	0.9645572137280	47.0	1.031629	0.9947791563005
48.0	1.175352	0.9759666924584	48.0	1.034150	0.9972099058180
49.0	1.188582	0.9869524694725	49.0	1.035597	0.9986049529090
50.0	1.204296	1.0000000000000	50.0	1.037044	1.0000000000000
51.0	1.220227	1.0132285108501	51.0	1.051076	1.0135314180882
52.0	1.235491	1.0259034375419	52.0	1.065109	1.0270628361763
53.0	1.253759	1.0410721288710	53.0	1.071722	1.0334396884843
54.0	1.278366	1.0615054344346	54.0	1.076215	1.0377723791408
55.0	1.310158	1.0879036710637	55.0	1.080648	1.0420468568251
56.0	1.348175	1.1194715146162	56.0	1.085033	1.0462747641318
57.0	1.362497	1.1313642195322	57.0	1.088700	1.0498114997334
58.0	1.368921	1.1366985449027	58.0	1.090935	1.0519658919248
59.0	1.385809	1.1507218356303	59.0	1.095459	1.0563282213600
60.0	1.403391	1.1653207912851	60.0	1.118300	1.0783540487449
61.0	1.422378	1.1810874633589	61.0	1.141142	1.1003798761299
62.0	1.435969	1.1923722637887	62.0	1.156588	1.1152741102021
63.0	1.441217	1.1967305353143	63.0	1.171110	1.1292768951102
64.0	1.466421	1.2176586141183	64.0	1.178534	1.1364363112958
65.0	1.504536	1.2493083329262	65.0	1.182410	1.1401740431203
66.0	1.569993	1.3036606799299	66.0	1.202303	1.1593561686135
67.0	1.641339	1.3629037558390	67.0	1.242216	1.1978437861926
68.0	1.711234	1.4209419553934	68.0	1.285361	1.2394469706890
69.0	1.769168	1.4690479512317	69.0	1.339813	1.2919546393959

(*Continued*)

TABLE 3 (Continued)

ACE DSLOC baseline growth factor distribution CDFs					
Copy shaded columns into ACEIT custom CDF dialog box			Copy shaded columns into ACEIT custom CDF dialog box		
New DSLOC growth factor CDF			Pre-existing DSLOC growth factor CDF		
%ile	Raw growth factor	Median-normalized growth factor	%ile	Raw growth factor	Median-normalized growth factor
70.0	1.791218	1.4873573359220	70.0	1.394266	1.3444623081028
71.0	1.810478	1.5033503939377	71.0	1.446846	1.3951643364272
72.0	1.826520	1.5166707673287	72.0	1.499427	1.4458663647515
73.0	1.833663	1.5226022189589	73.0	1.512964	1.4589197393336
74.0	1.836591	1.5250336550182	74.0	1.515345	1.4612163557037
75.0	2.001786	1.6622047665646	75.0	1.546372	1.4911344165383
76.0	2.176723	1.8074659984159	76.0	1.600314	1.5431496329445
77.0	2.270585	1.8854052239881	77.0	1.651218	1.5922352307263
78.0	2.358345	1.9582776404535	78.0	1.696045	1.6354615912594
79.0	2.433223	2.0204534599156	79.0	1.739863	1.6777139417320
80.0	2.516756	2.0898160858878	80.0	1.775599	1.7121742117209
81.0	2.607790	2.1654072775023	81.0	1.811336	1.7466344817099
82.0	2.690278	2.2339015178687	82.0	1.838907	1.7732199061744
83.0	2.769916	2.3000301078190	83.0	1.865456	1.7988209749484
84.0	2.893396	2.4025628420106	84.0	1.877871	1.8107927137641
85.0	2.997528	2.4890301913380	85.0	1.883219	1.8159497876006
86.0	3.005193	2.4953945827531	86.0	2.007928	1.9362040968520
87.0	3.055583	2.5372369555455	87.0	2.281838	2.2003299503721
88.0	3.172005	2.6339089359211	88.0	2.502560	2.4131677723693
89.0	3.403969	2.8265227894852	89.0	2.537125	2.4464974840362
90.0	3.710696	3.0812166786418	90.0	2.571689	2.4798271957032
91.0	4.007346	3.3275433624827	91.0	2.669888	2.5745180730522
92.0	4.295195	3.5665622442057	92.0	2.768086	2.6692089504012
93.0	4.404569	3.6573823260358	93.0	2.972917	2.8667228978367
94.0	4.555443	3.7826615156815	94.0	3.208214	3.0936148652969
95.0	4.830813	4.0113180287745	95.0	3.477960	3.3537251840393
96.0	5.272124	4.3777655963462	96.0	3.775264	3.6404101838074
97.0	5.904905	4.9032028421625	97.0	4.167301	4.0184431884618
98.0	6.163649	5.1180536566296	98.0	4.748802	4.5791722028886
99.0	6.245217	5.1857845825628	99.0	5.265691	5.0775979934902
100.0	6.253957	5.1930414674842	100.0	5.265691	5.0775979934902

The default DSLOC baseline growth factor distribution statistics and CDF tables are developed from historical data reported in SRDRs and collected by the AFCAA. This data were filtered first by eliminating all data points where the New or PER growth factor is zero or undefined (i.e., the estimated value cannot be zero and the final actual value cannot be zero):

$$Candidate_i = EstNew_i \neq 0 \wedge EstPER_i \neq 0 \wedge ActNew_i \neq 0 \wedge ActPER_i \neq 0, \qquad (4)$$

where

$Candidate_i \equiv$ Boolean indicator of the i^{th} project in the list of SRDR projects where TRUE indicates the element satisfies the filter criteria, FALSE indicates it does not;
$EstNew_i \equiv$ ARO/ATP estimated New DSLOC of the i^{th} project in the list of SRDR projects;
$EstPER_i \equiv$ ARO/ATP estimated PER DSLOC of the i^{th} project in the list of SRDR projects;
$ActNew_i \equiv$ Actual delivered New DSLOC of the i^{th} project in the list of SRDR projects;
$ActPER_i \equiv$ Actual delivered PER DSLOC of the i^{th} project in the list of SRDR projects;
$\wedge \equiv$ Symbolic logic "and" operator; if both operands evaluate to TRUE then the expression evaluates to TRUE, otherwise the expression evaluates to FALSE.

Ideally, the filtering described in Equation (4) would be the only filtering necessary; however, the SRDR data set contains several instances of extreme (considered unrealistic by the author) growth or shrinkage. Therefore, the resulting filtered data are filtered again to eliminate all data points that are outside above and below two multiplicative standard deviations of the filtered data set mean. This additional filtering served to remove three data points (SRDR instances) from the original 59 New software data points (5%) and the same three data points from the original 48 PER data points (6%).

The author recognizes that choosing to perform this additional filtering is subject to some criticism; however, the statistics from the resulting data set show virtually no change in the dataset median positions[3] while reducing the coefficients of variation (CV) to values the author considers somewhat more reasonable[4] at the risk of possibly being somewhat more optimistic. The author acknowledges the point of view that suggests a no-pruning strategy might have been more appropriate since it would have "completely" captured the inherent uncertainty.

$$CandidateNew_i = K_{GFNew_i} \in \left((\%SEE_{GFNew} + 1)^{-2}\,\overline{K}_{GFNew}, (\%SEE_{GFNew} + 1)^{2}\,\overline{K}_{GFNew}\right)$$

where

$$K_{GFNew_i} \equiv ActNew_i / EstNew_i$$

and where

$$\%SEE_{GFNew} \equiv \sqrt{\frac{1}{(N-1)} \sum_{i=1}^{N} \left(\frac{\left(K_{GFNew_i} - \overline{K}_{GFNew}\right)}{\overline{K}_{GFNew}} \right)^2}$$

$$CandidatePER_i = K_{GFPER_i} \in \left((\%SEE_{GFPER} + 1)^{-2}\,\overline{K}_{GFPER}, (\%SEE_{GFPER} + 1)^{2}\,\overline{K}_{GFPER}\right)$$

where

$$K_{GFPER_i} \equiv ActPER_i / EstPER_i$$

and where

$$\%SEE_{GFPER} \equiv \sqrt{\frac{1}{(N-1)} \sum_{i=1}^{N} \left(\frac{\left(K_{GFPER_i} - \overline{K}_{GFPER}\right)}{\overline{K}_{GFPER}} \right)^2}$$

where

$CandidateNew_i \equiv$ Boolean indicator of the i^{th} project in the list of SRDR projects where TRUE indicates the project data satisfies the filter criteria, FALSE indicates it does not;
$CandidatePER_i \equiv$ Boolean indicator of the i^{th} project in the list of SRDR projects where TRUE indicates the project data satisfies the filter criteria, FALSE indicates it does not;
$K_{GFNew_i} \equiv$ New DSLOC estimate growth factor of the i^{th} project in the list of SRDR projects;
$K_{GFPER_i} \equiv$ PER DSLOC estimate growth factor of the i^{th} project in the list of SRDR projects;
$\%SEE_{GFNew} \equiv$ Percentage Standard Error of Estimate of the list of New DSLOC estimate growth factors belonging to those projects in the list of projects defined by $Candidate_i =$ TRUE;
$\%SEE_{GFPER} \equiv$ Percentage Standard Error of Estimate of the list of PER DSLOC estimate growth factors belonging to those projects in the list of projects defined by $Candidate_i =$ TRUE.

DSLOC Estimate Uncertainty Decay

Decrease (decay) of the uncertainty implied by DSLOC estimate growth-factor distributions as a project progresses from start to finish and is modeled by the general form:

$$\boldsymbol{K_{GFAdj}} = e^{-bt}\left(\boldsymbol{K_{GF}} - 1\right) + 1, \tag{5}$$

where

$t \equiv$ Normalized EM (percentage of the development process *duration* at which the estimate is performed); $t_{start} \equiv t_0 \equiv 0\%$ and $t_{finish} \equiv 100\%$;
$b \equiv$ Decay parameter; by default is set to a value of 3.466, which emulates the decay behavior of Boehm's *Cone of Uncertainty*;[5]
$\boldsymbol{K_{GF}} \equiv$ Growth factor distribution at time t_0;
$\boldsymbol{K_{GFAdj}} \equiv$ Decay-adjusted growth factor distribution at EM t.

The practical effect of applying this model is time-progressive compression of the DSLOC estimate distribution about the TBE position approaching no uncertainty at process completion.

In order to render Equation (5) useful in a particular estimating situation, we need to assume some value (or distribution) for the uncertainty decay function proportionality constant b. Two methods for accomplishing this are: (1) to perform a regression analysis of relevant historical data to determine an expected value or distribution for b and (2) to assume uncertainty decay consistent with Boehm's *Cone of Uncertainty*. The latter is considered to be the model's default and can be accomplished by assuming $b = 3.466$ (see Figure 1) and time t to be normalized according to the SDLC EM scale in Table 1.

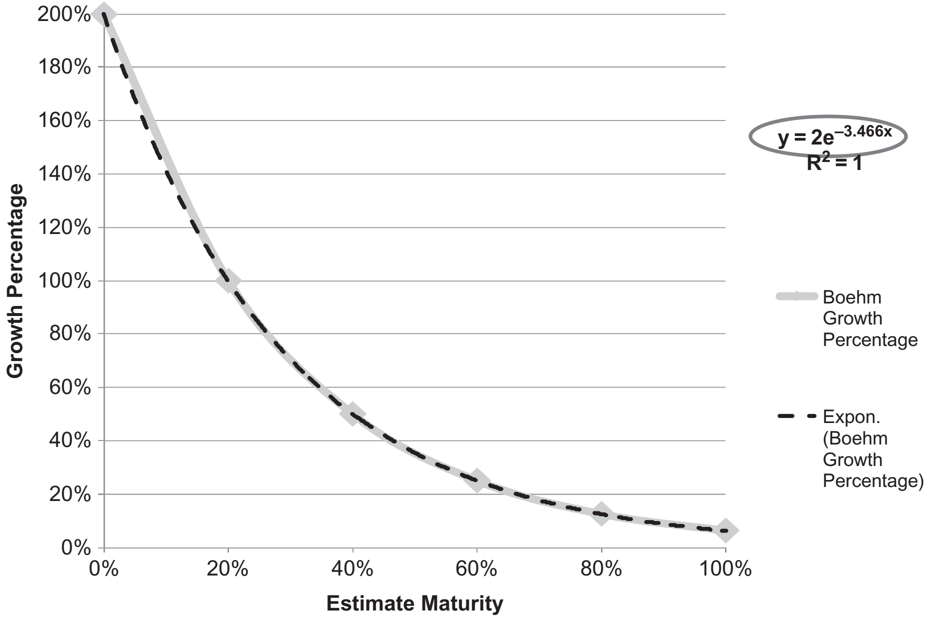

FIGURE 1 Curve fit of Boehm *Cone of Uncertainty*—top half.

Decay-Adjusted DSLOC Growth-Factor Distributions

We assume some normalized uncertainty scale factor function $\boldsymbol{K_U}$ of time t where $\boldsymbol{K_U}(t) \in [0, 1]$, where $\boldsymbol{K_U}(t|t = 0) = 1$ represents maximum (full scale) uncertainty, and hypothesize that $\boldsymbol{K}_U(t)$ decreases (decays) at a rate proportional to its value (i.e., uncertainty tends to decay faster during the early stages of a process when experience is low and tends to decay slower during the later stages of a process when experience is high). We model this hypothetical behavior mathematically as:

$$\frac{d\ \boldsymbol{K_U}(t)}{dt} \propto -\boldsymbol{K_U}(t) \quad \therefore \frac{d\ \boldsymbol{K_U}(t)}{dt} = -b\ \boldsymbol{K_U}(t), \tag{6}$$

where b is the constant of proportionality.[6] Solving the ordinary differential Equation (6) yields:

$$\frac{d\boldsymbol{K_U}(t)}{\boldsymbol{K_U}(t)} = -bdt \quad \rightarrow \int \frac{d\boldsymbol{K_U}(t)}{\boldsymbol{K_U}(t)} = \int -bdt \quad \rightarrow \quad \ln(\boldsymbol{K_U}(t)) = -bt + c$$

$$\therefore \boldsymbol{K_U}(t) = e^{-bt}e^{c}. \tag{7}$$

Since we have already posited the constraint $\boldsymbol{K_U}(t|t = 0) = 1$, we can solve Equation (7) for the constant of integration c:

$$\boldsymbol{K_U}(0) = e^{-b(0)}e^{c} = 1 \quad \rightarrow \quad e^{c} = 1 \quad \therefore c = 0. \tag{8}$$

Substituting the equivalent of c in Equation (8) for c in Equation (7) yields:

$$\boldsymbol{K_U}(t) = e^{-bt}e^{(0)} \quad \therefore \boldsymbol{K_U}(t) = e^{-bt}. \tag{9}$$

Applying Uncertainty Decay to Growth-Factor Distributions

Suppose we have a baseline DSLOC estimate growth factor distribution $\boldsymbol{K_{GF}}$, which has been developed from historical data and which models the amount of uncertainty that exists about the TBE of DSLOC, assuming that this estimate is done at the beginning of a software development process; i.e., EM is zero, consistent with the processes from which the historical data were collected. Suppose this baseline distribution is represented as a CDF; i.e., a mapping of growth factor values to percentiles. We would like to model what happens to the uncertainty modeled by this baseline distribution as activities in the process progress to completion. We have already hypothesized that uncertainty decays over time and have developed a model for this decay in Equation (9). Since the function $\boldsymbol{K_U}(t)$ in Equation (9) is normalized (i.e., yields uncertainty factors that are percentages of full scale), we can scale our baseline DSLOC estimate growth factor distribution by the transformation:

$$\boldsymbol{K_{GFAdj}} = \boldsymbol{K_U}(t)\,(\boldsymbol{K_{GF}} - 1) + 1, \tag{10}$$

where

$\boldsymbol{K_{GF}} \equiv$ baseline growth factor distribution at $t = 0$ (0% EM) which is given as a custom CDF (see Table 3);
$\boldsymbol{K_{GFAdj}} \equiv$ decay-adjusted growth factor distribution at some EM t.

This transformation effectively scales the percentage differences between the growth factors in the baseline growth factor distribution and no growth (a growth factor of 1).

Substituting the value of $\boldsymbol{K_U}(t)$ in Equation (9) for $\boldsymbol{K_U}(t)$ in Equation (10) yields:

$$\boldsymbol{K_{GFAdj}} = e^{-bt}\,(\boldsymbol{K_{GF}} - 1) + 1. \tag{11}$$

As stated earlier, in order to render Equation (11) useful in a particular estimating situation, we need to assume some value (or distribution) for the uncertainty decay function proportionality constant b; either by assuming $b = 3.466$ (Boehm's *Cone of Uncertainty*) or by analyzing relevant historical data to model decay as a single value b or as a distribution $\boldsymbol{B}$. Figures 2 and 3 illustrate the behavior of Equation (11) with decay constant $b = 3.466$ over the range of possible EM values $t \in [0, 1]$.

Applying Growth Factor Distributions to TBEs of New and PER DSLOC

We can now transform single-point TBEs of New S_{DNew} and PER S_{DPER} DSLOC into growth-adjusted distributions of New $\boldsymbol{S_{DAdjNew}}$ and PER $\boldsymbol{S_{DAdjPER}}$ DSLOC by scaling the appropriate instantiation of Equation (11) (a distribution) by the corresponding single-point TBE:

$$\boldsymbol{S_{DAdjNew}} \equiv S_{DNew}\left(e^{-bt}\,(\boldsymbol{K_{GFNew}} - 1) + 1\right) \tag{12}$$

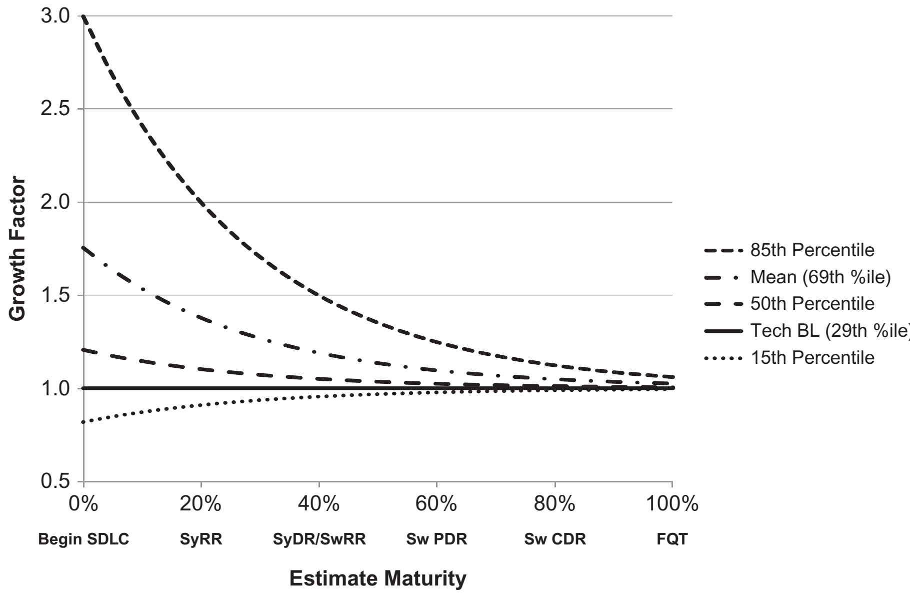

FIGURE 2 New DSLOC growth-factor decay.

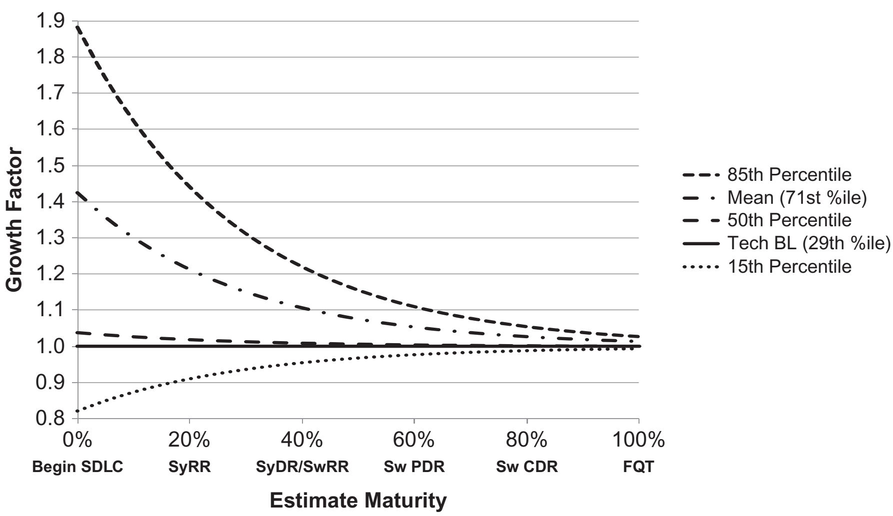

FIGURE 3 PER DSLOC growth-factor decay.

and

$$\boldsymbol{S_{DAdjPER}} \equiv S_{D\_New}\left(e^{-bt}\left(\boldsymbol{K_{GFPER}} - 1\right) + 1\right). \tag{13}$$

Figures 4 and 5 illustrate the behaviors of the growth-adjusted New DSLOC estimate distribution as described in Equation (12) and the growth-adjusted PER DSLOC estimate distribution as described in Equation (13) for given New and PER TBEs and a given EM.

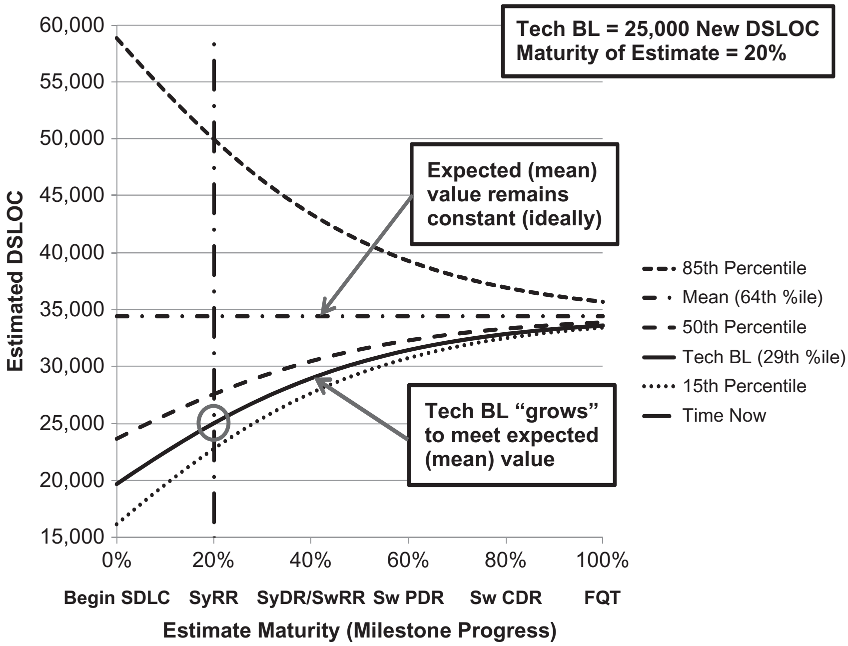

FIGURE 4 Example growth-adjusted New DSLOC distribution vs. estimate maturity.

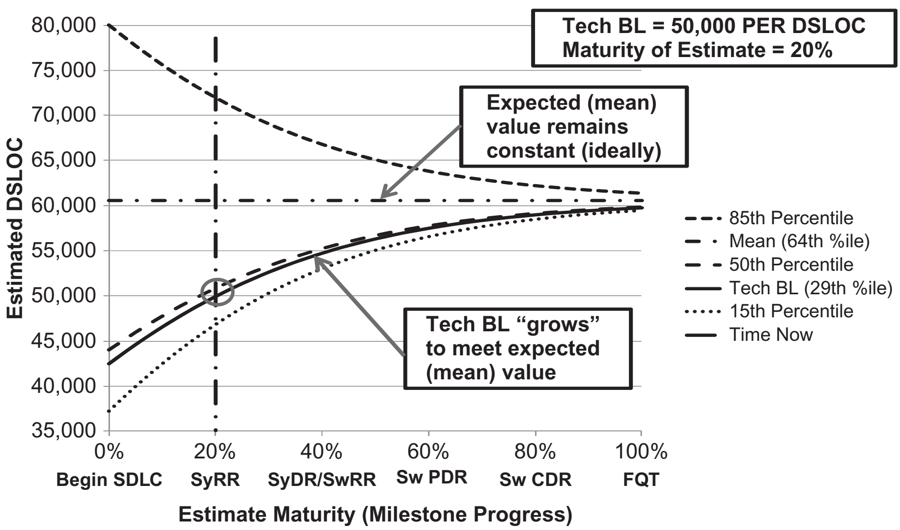

FIGURE 5 Example growth-adjusted PER DSLOC distribution vs. estimate maturity.

Modeling DSLOC Growth in ACEIT

The process for using DEGM6 within ACEIT for each of a particular set of computer programs (CSCIs) is (see Table 4):

- Define a variable for each CSCI for each of New and PER to represent the particular CSCI's New and PER DSLOC baseline growth factor distributions; e.g., SI010101_New_BL_GF and SI010101_PER_BL_GF. These will represent the

TABLE 4 Example ACEIT application of new DSLOC estimate growth

WBS/CES Description	Unique ID	Equation / Throughput
New Growth-Adjusted DSLOC	SI010101_New_Adj_Sd	SI010101_New_Adj_GUF * SI010101_New_Sd
Technical Baseline DSLOC Point Estimate	SI010101_New_Sd	25000 [Given]
Maturity at DSLOC Estimate	SI010101_New_Sd_Est_Mat	0.20 [Sys Req Rev Complete = 20% Estimate Maturity]
Baseline Growth Factor	SI010101_New_BL_GF	1.204296 [Tecolote DSLOC Estimate Growth Model v06 Median of SRDR New DSLOC Data Set]
Decay Constant	SI010101_New_GF_Decay	3.466 [Tecolote DSLOC Estimate Growth Model v06 Default]
Adjusted Growth Factor	SI010101_New_Adj_GUF	exp(−SI010101_New_GF_Decay * SI010101_New_Sd_Est_Mat) * (SI010101_New_BL_GF - 1) + 1 [Tecolote DSLOC Estimate Growth Model v06]

random variables (distributions) $\boldsymbol{K_{GFNew}}$ and $\boldsymbol{K_{GFPER}}$ in Equations (1) and (2), respectively. These variables must be described as distributions using ACEIT's custom CDF feature. The model default position CDFs are shown in Table 3. Note that when using ACEIT's custom CDF feature, it is best to normalize the growth factor values about the median growth factor value in the right-most (shaded) columns of Table 3 and set the point estimate to the median (50th percentile) growth factor value in order to see reasonable point estimate values and percentages that are calculated from the CDF.

- Define a variable for each CSCI for each of New and PER; e.g., SI010101_New_Sd_Est_Mat and SI010101_PER_Sd_Est_Mat for each of New and PER; that will represent the EM variable t in Equations (1) and (2). For example, if the current TBE of New DSLOC for SI010101 was performed at successful completion of a System Requirements Review (System Requirements Analysis is complete) then the variable SI010101_New_Sd_Est_Mat would, from Table 1, be entered as 0.2 (20%).
- Define a new variable for each CSCI for each of New and PER; e.g., SI010101_New_GF_Decay and SI010101_PER_GF_Decay; that will represent the decay parameter variable b in Equations (1) and (2). Note that these variables could alternatively be described as random variables (distributions) $\boldsymbol{B}$ using ACE's custom CDF feature based on some program-specific historical data. The model default is a constant value for b of 3.466.
- Define a variable for each CSCI for each of New and PER; e.g., SI010101_New_Adj_GUF and SI010101_PER_Adj_GUF; that will represent the uncertainty-decay-adjusted version of the New DSLOC and PER DSLOC growth factor distributions for that CSCI. The equation field for each of these

variables implements Equation (11); e.g., exp(–SI010101_New_GF_Decay * SI010101_New_Sd_Est_Mat) * (SI010101_New_BL_GF – 1) + 1.

- If the decay constant is being described as a random variable ***B*** (distribution) then, because each decay constant random variable is inversely related to its corresponding growth factor random variable, as can be seen in Equation (11), we would need to negatively correlate each growth factor/decay constant pair in order for the convolution of these two variables to work properly in ACEIT.[7] For example, we would group SI010101_New_GF_Decay and SI010101_New_BL_GF and call the group SI010101_Growth_Decay_Group. We would then set the Group Strength of SI010101_New_GF_Decay to "–1" and set the Group Strength of SI010101_New_BL_GF to "D." Note that none of this step is necessary if using the model defaults based on SRDR data and assuming the decay to be constant with a value of 3.466.

Modeling DSLOC Growth in SEER-SEM

Bi-Normal Distribution

The Galorath, Inc. SEER family of estimating tools incorporates a rather unorthodox probability distribution to model input uncertainty. This officially unnamed distribution here is referred to as a bi-normal distribution because it combines the left half of the Probability Density Function (PDF) of one normal (Gaussian) distribution that has a particular mean and standard deviation with the right half of the PDF of another normal distribution that has the same mean but possibly a different standard deviation. Figures 6 and 7 show, respectively, the PDF and CDF of an example bi-normal distribution where the mean of each component distribution is zero, where the standard deviation of the left (low) distribution equals 1, and where the standard deviation of the right (high) distribution equals 4.

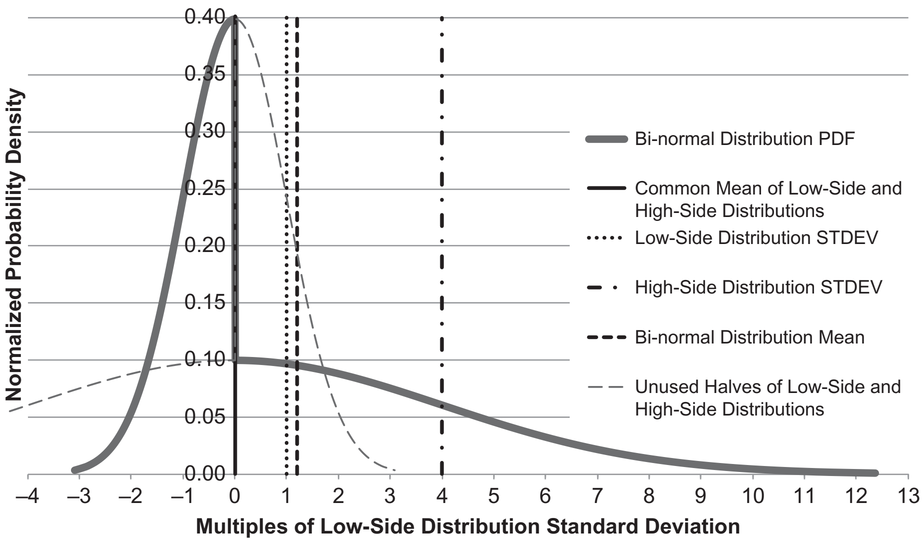

FIGURE 6 PDF of an example bi-normal distribution.

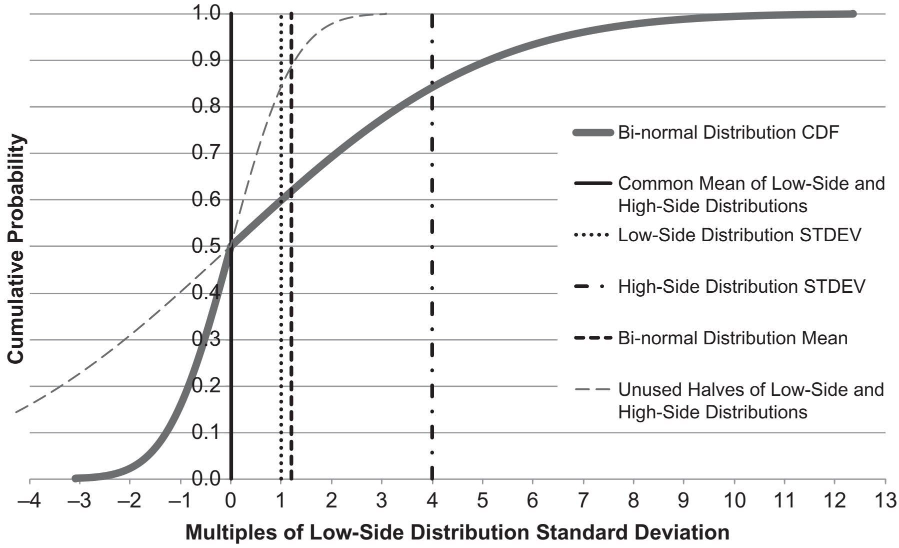

FIGURE 7 CDF of an example bi-normal distribution.

The PDF of the bi-normal distribution can thus be described as:

$$\boldsymbol{f}_{\textbf{\textit{bi-normal}}}\left(x \mid \tilde{\mu}, \sigma_L^2, \sigma_H^2\right) = \begin{cases} \boldsymbol{f}_{\textbf{\textit{normal}}}\left(x \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_L^2\right) & x \leq \tilde{\mu} \\ \boldsymbol{g}_{\textbf{\textit{normal}}}\left(x \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_H^2\right) & x \geq \tilde{\mu} \end{cases}, \tag{14}$$

the CDF can be described as:

$$\boldsymbol{F}_{\textbf{\textit{bi-normal}}}\left(x \mid \tilde{\mu}, \sigma_L^2, \sigma_H^2\right) = \begin{cases} \boldsymbol{F}_{\textbf{\textit{normal}}}\left(x \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_L^2\right) & x \leq \tilde{\mu} \\ \boldsymbol{G}_{\textbf{\textit{normal}}}\left(x \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_H^2\right) & x \geq \tilde{\mu} \end{cases}, \tag{15}$$

and the inverse CDF (i.e., the quantile function) can be described as:

$$\boldsymbol{F}^{-1}_{\textbf{\textit{bi-normal}}}\left(p \mid \tilde{\mu}, \sigma_L^2, \sigma_H^2\right) = \begin{cases} \boldsymbol{F}^{-1}_{\textbf{\textit{normal}}}\left(p \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_L^2\right) & p <= 0.5 \\ \boldsymbol{G}^{-1}_{\textbf{\textit{normal}}}\left(p \mid \mu = \tilde{\mu}, \sigma^2 = \sigma_H^2\right) & p >= 0.5. \end{cases} \tag{16}$$

$p \in (0, 1)$

We have already stated that, for bi-normal distributions, the mean of the left (low-side) distribution μ_L is always equal to the mean of the right (high-side) distribution μ_H. However, the low-side distribution standard deviation σ_L need not equal the high-side distribution standard deviation σ_H. When $\sigma_H > \sigma_L$ the overall bi-normal distribution is skewed to the right, when $\sigma_L > \sigma_H$ the overall distribution is skewed to the left, and when $\sigma_L = \sigma_H$ the overall distribution is symmetrical and classically normal. Because, for bi-normal distributions, the low-side and high-side distributions are normal, because the low-side distribution mean equals the high-side distribution mean, and because the mean of any normal distribution is always its median (50th percentile) value, it follows that the low-side distribution contributes half of the overall distribution's probability density and the high-side distribution contributes the other half of the overall distributions probability density. Therefore, μ_L and μ_H are always equal to the overall bi-normal distribution's median value $\boldsymbol{m}$. Note, however, that $\mathscr{m}$ is not necessarily equal to the overall bi-normal distribution's mean μ; this is only true for the special case of a symmetric bi-normal distribution, i.e., one where $\sigma_L = \sigma_H$.

SEER-SEM uses bi-normal distributions to model the uncertainty about its DSLOC inputs. For each DSLOC input, SEER-SEM expects the user to have elicited each of a least (*L*), likely (*M*), and most (*H*) DSLOC value for that input. Together these values describe the range of possible DSLOC outcomes for that input and the particular DSLOC outcome within that range that is estimated to be the "most likely" to occur. SEER-SEM turns each least, likely, and most triple $\langle L, M, H \rangle$ into a bi-normal distribution according to the following assignments:

$$\mu_L = \mu_H = \mathscr{m} = \frac{(L + 4M + H)}{6}, \tag{17}$$

$$\sigma_L = \frac{\mathscr{m} - L}{3}, \tag{18}$$

and

$$\sigma_H = \frac{H - \mathscr{m}}{3}. \tag{19}$$

It is important to note here that the relationship in Equation (17) constrains the amount of skew that can be modeled by the bi-normal distribution. Maximum right (high-side) skew occurs when $M = L$ and maximum left (low-side) skew occurs when $M = H$.

SEER-SEM requires that the uncertainty about a DSLOC estimate be characterized as a (Least, Likely, Most) triple. Since SEER-SEM provides no facility for specifying DSLOC growth and growth-uncertainty decay, DSLOC inputs to SEER-SEM must already be growth- and uncertainty-adjusted. Therefore, in order to model DSLOC growth in SEER-SEM according to DEGM6, DSLOC Least, Likely, and Most values must be chosen to force SEER-SEM's bi-normal distribution to match, as closely as possible, the distributions described in Table 3 and adjusted for uncertainty decay as a function of EM.

Growth-adjusted Least L_{Adj}, Likely M_{Adj}, and Most H_{Adj} DSLOC inputs to SEER-SEM can be calculated for each of New and PER as functions of the given New and PER DSLOC TBEs S_{DNew} and S_{DPER} with given EM t. For each of New and PER we define a set of three DSLOC estimate growth multipliers K_{LAdj}, K_{MAdj}, and K_{HAdj} using Equation (11):

$$K_{LAdj} = e^{-3.466t}\,(K_L - 1) + 1 \tag{20}$$

and

$$K_{MAdj} = e^{-3.466t}\,(K_M - 1) + 1 \tag{21}$$

and

$$K_{HAdj} = e^{-3.466t}\,(K_H - 1) + 1, \tag{22}$$

such that

$$L_{Adj} = K_{LAdj} S_D \quad \text{and} \quad M_{Adj} = K_{MAdj} S_D \quad \text{and} \quad H_{Adj} = K_{HAdj} S_D\,. \tag{23}$$

We first instantiate Equations (18), (19), and (17) with K_L, K_H, and K_M, respectively:

$$\sigma_L = \frac{\mathscr{m} - K_L}{3} \quad \therefore K_L = \mathscr{m} - 3\sigma_L \tag{24}$$

and

$$\sigma_H = \frac{K_H - \mathscr{m}}{3} \quad \therefore K_H = \mathscr{m} + 3\sigma_H \tag{25}$$

and

$$\mathscr{m} = \frac{(K_L + 4K_M + K_H)}{6} \quad \rightarrow \quad K_M = \frac{6\mathscr{m} - K_L - K_H}{4}$$

$$\rightarrow \quad K_M = \frac{6\mathscr{m} - (\mathscr{m} - 3\sigma_L) - (\mathscr{m} + 3\sigma_H)}{4}$$

$$\therefore K_M = \frac{4\mathscr{m} + 3\sigma_L - 3\sigma_H}{4}. \tag{26}$$

Recall that $\mathscr{m}$ is always equal to the overall bi-normal distribution median. We wish to force this value to be equal to the SRDR data set median $\mathscr{m}_{dataset}$; therefore,

$$K_L = \mathscr{m}_{dataset} - 3\sigma_L \text{ and } K_H = \mathscr{m}_{dataset} + 3\sigma_H \text{ and } K_M = \frac{4\mathscr{m}_{dataset} + 3\sigma_L - 3\sigma_H}{4}. \tag{27}$$

Substituting the equivalents of K_L, K_H, and K_M in Equations (27) for K_L, K_H, and K_M in Equations (20), (22), and (21), respectively, yields:

$$K_{LAdj} = e^{-3.466t}\left(\mathscr{m}_{dataset} - 3\sigma_L - 1\right) + 1 \tag{28}$$

and

$$K_{MAdj} = e^{-3.466t}\left(\left(\frac{4\mathscr{m}_{dataset} + 3\sigma_L - 3\sigma_H}{4}\right) - 1\right) + 1 \tag{29}$$

and

$$K_{HAdj} = e^{-3.466t}\left(\mathscr{m}_{dataset} + 3\sigma_H - 1\right) + 1. \tag{30}$$

Appropriate values for $\mathscr{m}_{dataset}$ can be found in Table 2. Appropriate values for σ_L and σ_H have been determined by using the Microsoft Excel Solver add-in to minimize the difference between each SRDR data set mean value and its corresponding bi-normal distribution mean value by varying its associated σ_L and σ_H values. The results from running Solver and then calculating K_L, K_M, and K_H are shown in Table 5.

Substituting the computed values of K_L, K_M, and K_H in Table 5 for K_L, K_M, and K_H in Equations (20), (21), and (22) for each New and PER DSLOC yields:

- For New DSLOC:

$$K_{LAdj} = -0.828071e^{-3.466t} + 1$$

$$K_{MAdj} = -0.828071e^{-3.466t} + 1$$

$$K_{HAdj} = 5.366128e^{-3.466t} + 1. \tag{31}$$

TABLE 5 Bi-normal distribution parameters and resulting multiplier values

DSLOC Type	Solver change values (results) $\sigma[L]$	$\sigma[H]$	Solver target (objective) $\lvert\mu$[SRDR Data Set] − μ[binormal approx]$\rvert$	SEER-SEM multiplier expression scale factors *K[L]*	*K[M]*	*K[H]*
New	0.344122	1.720611	0.000001	−0.828071	−0.828071	5.366128
Pre-Existing	0.241411	1.207058	0.000000	−0.687191	−0.687192	3.658219

- For Pre-Existing DSLOC:

$$\begin{aligned} K_{LAdj} &= -0.687191e^{-3.466t} + 1 \\ K_{MAdj} &= -0.687192e^{-3.466t} + 1 \\ K_{HAdj} &= 3.658219e^{-3.466t} + 1. \end{aligned} \tag{32}$$

Substituting the multiplier expressions in the sets of Equations (31) and (32) for the multiplier variables in Equations (23), yields the sets of equations for determining the appropriate Least, Likely, and Most DSLOC values to input into SEER-SEM such that growth, growth uncertainty, and growth uncertainty decay are modeled consistent with DEGM6 and with the SRDR data upon which it is based.

- For New DSLOC:

$$\begin{aligned} Least &= S_D\left(-0.828071e^{-3.466t} + 1\right) \\ Likely &= S_D\left(-0.828071e^{-3.466t} + 1\right) \\ Most &= S_D\left(5.366128e^{-3.466t} + 1\right). \end{aligned} \tag{33}$$

- For PER DSLOC:

$$\begin{aligned} Least &= S_D\left(-0.687191e^{-3.466t} + 1\right) \\ Likely &= S_D\left(-0.687192e^{-3.466t} + 1\right) \\ Most &= S_D\left(3.658219e^{-3.466t} + 1\right). \end{aligned} \tag{34}$$

Figures 8 and 9 show comparisons between the resulting bi-normal CDFs and the corresponding SRDR data set CDFs.

Conclusions

It is this author's opinion that the DEGM6 model as described in this article represents a quantum improvement over the field of available software code growth methodologies. Specifically, among the advantages of this model over the Holchin (2003) and Jensen (2008) code growth matrices are the following:

- DEGM6 is based on AFCAA-collected SRDR data versus Holchin's Delphi survey of *experts* approach and Jensen's data from multiple proprietary sources.

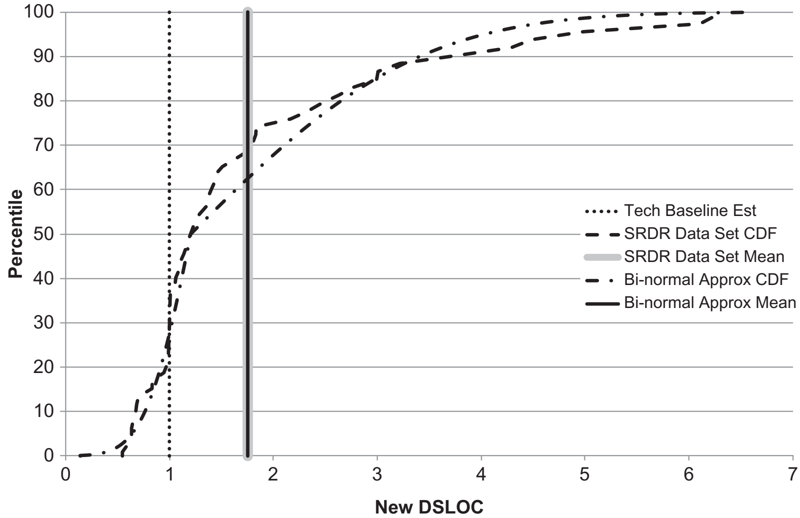

FIGURE 8 Comparison of New DSLOC growth factor CDFs—bi-normal approx vs. SRDR data set. Maturity of estimate = 0%. Mean growth factor values: bi-normal approx = 1.75; SRDR data set = 1.75. Confidence %s @ mean: bi-normal approx = 63%; SRDR data set = 69%.

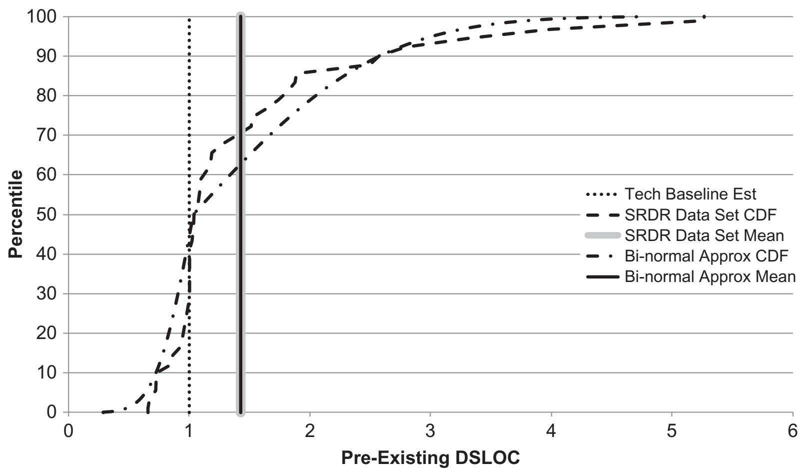

FIGURE 9 Comparison of PER DSLOC growth factor CDFs—bi-normal approx vs. SRDR data set. Maturity of estimate = 0%. Mean growth factor values: bi-normal approx = 1.42; SRDR data set = 1.42. Confidence %s @ mean: bi-normal approx = 63%; SRDR data set = 71%.

- DEGM6 requires only one parameter, EM, which is reasonably objective versus Holchin's and Jensen's rather subjective and vaguely-defined Complexity and Maturity parameters.
- DEGM6 produces a growth-factor distribution result (embodies uncertainty) versus Holchin's single-point growth-factor result. (Jensen uses the lognormal distribution as a model.)

- DEGM6 provides growth-factor distribution decay based on updated EM parameter versus Holchin's single-point growth factor reduction based on updated Complexity and Maturity parameters. (Jensen defines EM in terms of defined program phases.) DEGM6 differentiates between New and PER DSLOC growth versus Holchin's and Jensen's one-growth-factor-fits-all approach.

This model has been used as part of the basis for several USAF program office estimates and independent cost estimates. Planned enhancements to this model include rerunning the data analysis using a recently-updated version of the AFCAA SRDR data set. The number of programs and possible stratifications in this new data set may lead to unique baseline growth factor distributions for particular software types and/or characteristics.

Acronyms

ACEIT	Automated Cost Estimating Integrated Tools
AFCAA	Air Force Cost Analysis Agency
ARO	Announcement of Research Opportunity
ATP	Authority to Proceed
BL	Baseline
CDF	Cumulative Distribution Function
CES	Cost Estimating Structure
CSCI	Computer Software Configuration Item (i.e., a computer program)
CV	Coefficient of Variation (= standard deviation divided by mean)
DEGM6	DSLOC Estimate Growth Model v06
DSLOC	Delivered Source Lines of Code
EM	Estimate Maturity
GF	Growth Factor
PER	Pre-Existing Reused
SDLC	Software Development Life Cycle
SEE	Standard Error of the Estimate
SEER	System Evaluation and Estimation of Resources
SEM	Software Estimating Model
SRDR	Software Resources Data Report
STDEV	Standard Deviation
TBE	Technical Baseline Estimate
USAF	United States Air Force
WBS	Work Breakdown Structure

Notes

1. The term "Custom CDF" refers to a feature in ACEIT that allows distributions to be specified as a discrete range-value-to-percentile mapping as opposed to a mapping described by some mathematical distribution function such as "lognormal."
2. We use the ***Arial bold italic*** font to denote a random variable; i.e., a variable that can take on values according to some probability distribution, the ***Times New Roman bold italic*** font to denote a function, the **Times New Roman bold font** to denote a vector or matrix or array, the *Times New Roman italic font* to denote a simple variable, and the Times New Roman normal font to denote a number.
3. The median New DSLOC growth factor changed from 1.19 to 1.20 and the median PER DSLOC growth factor changed from 1.02 to 1.04.

4. The New DSLOC growth factor distribution CV changed from 0.98 to 0.75 and the median PER DSLOC growth factor distribution CV changed from 1.77 to 0.63.
5. Note that the model uses only the rate of uncertainty decay implied by Boehm's *Cone of Uncertainty*. The model does not use Boehm's growth factors but instead uses growth factors derived from the SRDR data.
6. The symbol $\propto$ indicates that the left operand is proportional (related by some factor) to the right operand.
7. Note that e^{-bt} is equivalent to $1/e^{bt}$.

References

Boehm, B. W. (1981). *Software engineering economics*. Englewood Cliffs, NJ: Prentice-Hall, Inc. ISBN 0-13-822122-7.

Holchin, B. (2003). Code Growth Study. Goleta, CA: Tecolote Research, Inc.

Jensen, R. (2008, October). Estimating software growth. *Space Systems Cost Analysis Group (SSCAG)*, Reston, VA.

Rosa, W. (2008). Software resources data report database. Arlington, VA: USAF Cost Analysis Agency (AFCAA).

About the Author

Michael A. Ross has over 35 years of experience in software engineering as a developer, manager, process expert, consultant, instructor, and award-winning international speaker. Mr. Ross is currently a Technical Expert for Tecolote Research, Inc. Mr. Ross's previous experience includes three years as President and CEO of r2Estimating, LLC (makers of the r2Estimator software estimation tool), three years as Chief Scientist of Galorath Inc. (makers of the SEER suite of estimation tools), seven years with Quantitative Software Management, Inc. (makers of the SLIM suite of software estimating tools) where he was a senior consultant and Vice President of Education Services, and 17 years with Honeywell Air Transport Systems (formerly Sperry Flight Systems) and two years with Tracor Aerospace where he developed and/or managed the development of real-time embedded software for various military and commercial avionics systems. Mr. Ross is a Life Member of ISPA, is currently on the Board of Directors of its Southern California chapter, and regularly presents papers at ISPA/SCEA annual conferences (four of which have been recognized with Best Paper Awards). Mr. Ross did his undergraduate work at the United States Air Force Academy and Arizona State University, receiving a Bachelor of Science in Computer Engineering.