NORTHROP GRUMMAN
DEFINING THE FUTURE

Software Development Schedule Realism: Prediction Band Tool

The Society of Cost Estimating and Analysis

June 2007

Allison Converse, Jeff Jaekle, Eric Druker
Intelligence Group (TASC)
Northrop Grumman Corporation

Outline

- Purpose
- Data Collection & Normalization
  - Code Language Adjustment
- Regression Analysis
- Prediction Bands
- Schedule Realism Prediction Band Tool
- Alternate Data Application
- Examples
- Conclusions
- Future Research

Allison.converse@ngc.com June, 2007

NORTHROP GRUMMAN

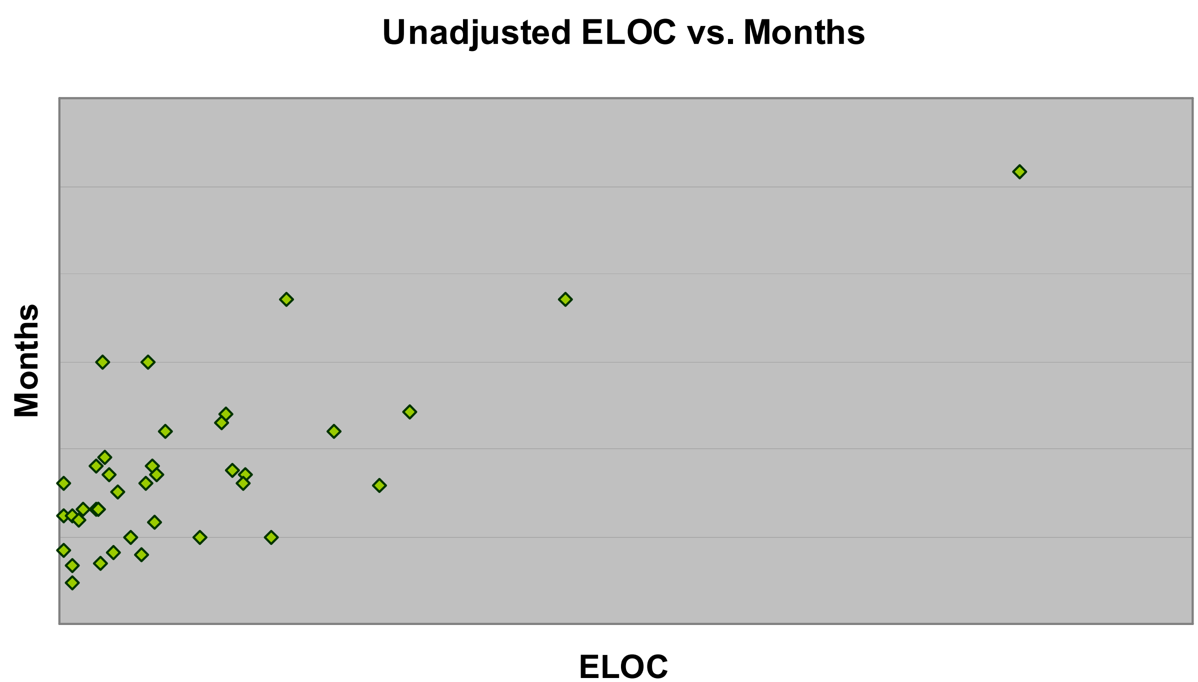
Purpose

- **The schedule realism tool provides:**
  - A methodology for producing schedule distributions based on historical programs for similar programs
    - The tool can be easily updated to incorporate additional data as it becomes available
  - A schedule prediction band for a proposed software development program based on ELOC
  - Having this prediction band allows the user to determine:
    - A suggested schedule length
      - This length can be chosen to reflect a level of risk acceptable to decision makers
    - The probability that a proposed software development schedule will be met
    - Upside/Most Likely/Downside scenarios for the final schedule
    - A new schedule prediction as ELOC changes
- **Application:**
  - Knowing both the probability of achieving a proposed schedule and the schedule length for an associated level of risk is invaluable when decision makers consider schedule changes, risk mitigation plans, funding and other decisions

Allison.converse@ngc.com

Data Collected

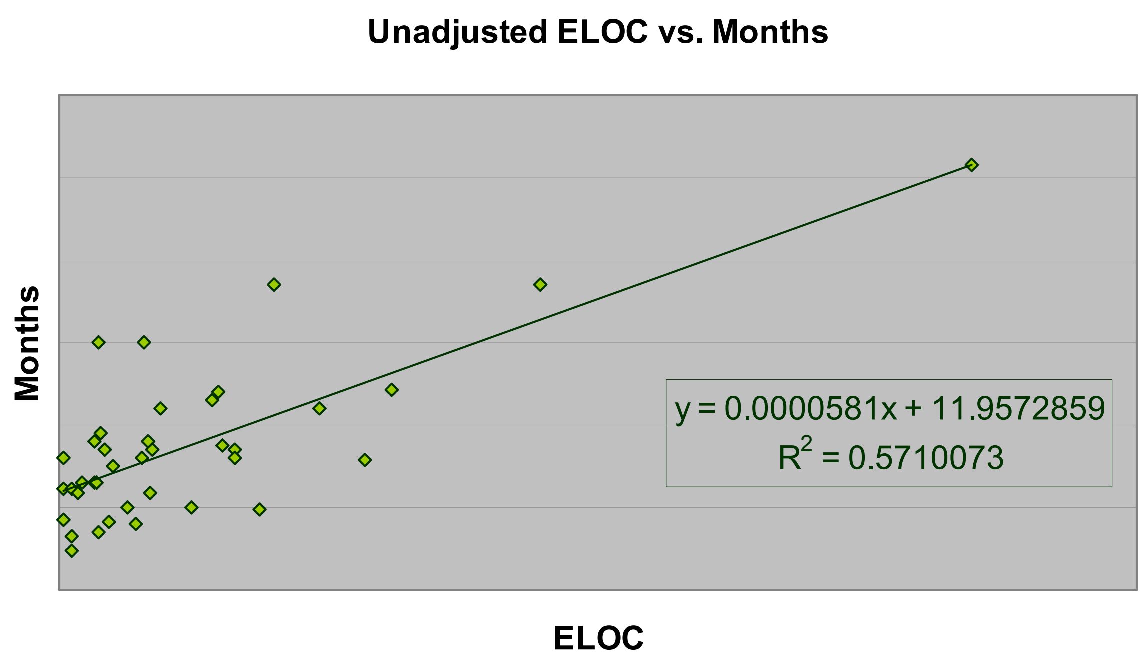
- **Collected 39 data points from completed Automated Information System (AIS) Segment software development releases**
  - Final Schedule was scatter-plotted against ELOC at Complete
  - Final Schedule is defined in Months
    - Start date: Requirements Review
    - End date: Pre-Ship Review (PSR)
  - ELOC at Complete – break outs by code language are known



Allison.converse@ngc.com

Code Language Adjustments

- **The data graphed below is the raw ELOC data**
  - A linear, statistically significant regression exists
  - Regression line equation: y = 5.81E-05*x + 11.96
- **The data shown includes bias due to code language**
  - Time per line of code differs based on programming language
    - For example, it takes longer to code one line of SQL than it does to code one line of C++
  - The steps by which the data was adjusted and the results of the normalization are shown on the following slides



Allison.converse@ngc.com June, 2007

Code Language Adjustments (cont’d)

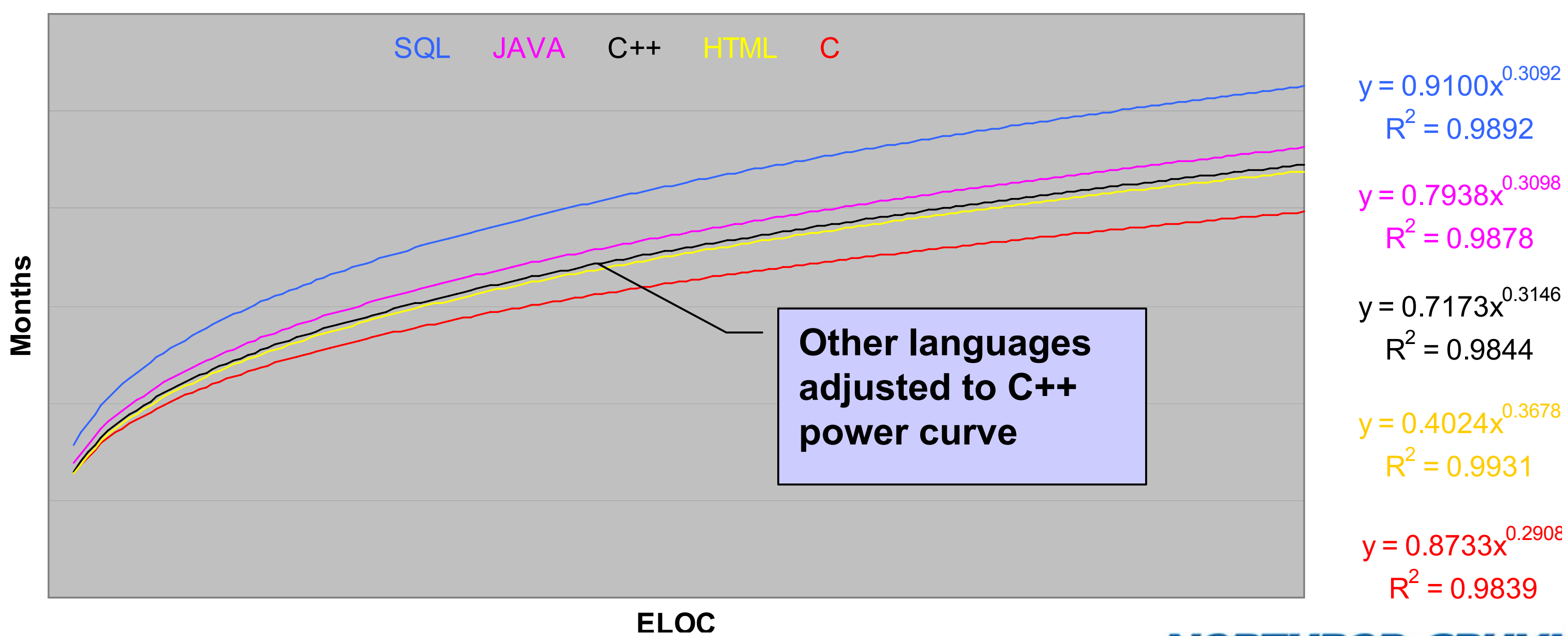
- **A commercially available software cost estimating tool was used to determine the power curves for the five most common languages in the data set (SQL, Java, C++, HTML, C)**
  - Graphs of the power curves for C++, C, HTML, Java and SQL can be seen on the following slide
- **C++ was chosen as the baseline language**
  - The majority of the code within the data set was C++
  - As seen on the following graph, the C++ power curve is the middle curve.
    - Normalizing to the middle curve minimizes the effect of potential errors in the curves as adjustments are minimized
- **For the languages that were present in the data set but not adjusted (D, IDL, JSP and Scripts), the amount of code in the data set was negligible**

Allison.converse@ngc.com

Code Language Adjustments (cont'd)

- **The commercially available software cost estimating tool was used with all settings at notional except language**
- **For all five language, 15 incremental values of ELOC were entered**
- **The resulting output for each language was fit with a power curve**
  - The corresponding equations are then use to determine a correction for language (detailed on the following slide)
- **The resultant power curves are shown below:**

Code Language Power Curves: Months vs ELOC



NORTHROP GRUMMAN

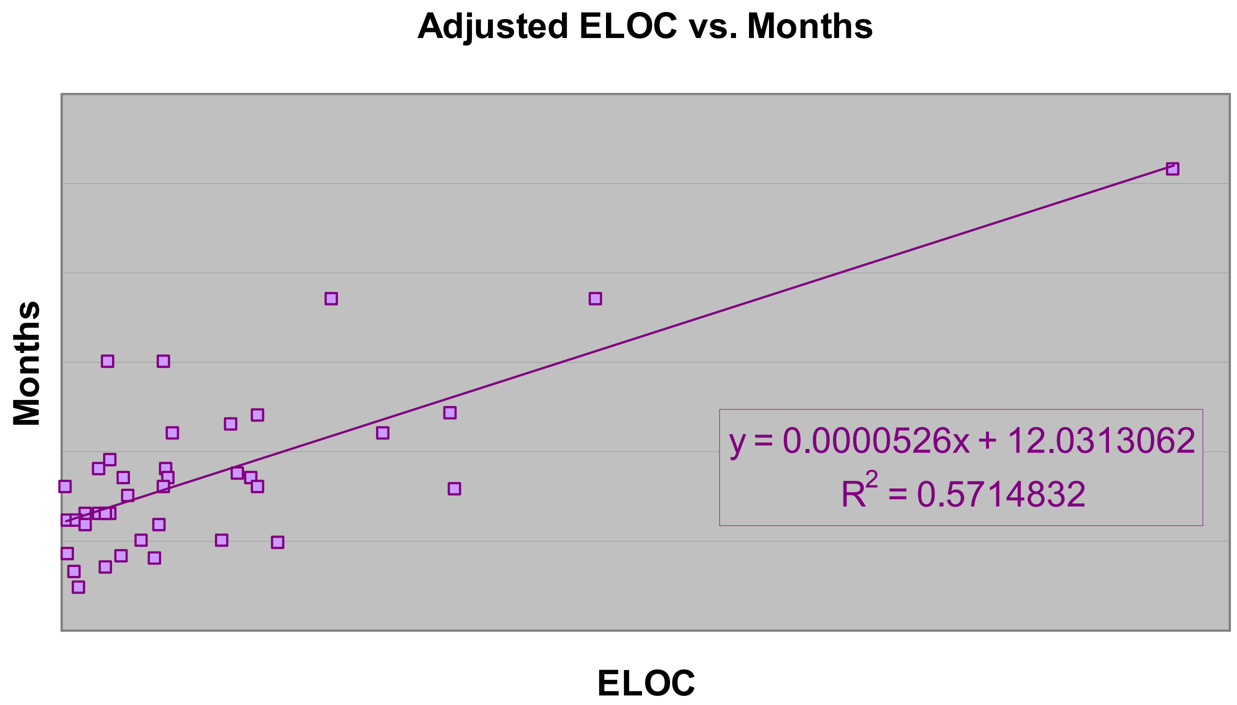
Allison.converse@ngc.com June, 2007

Code Language Adjustments (cont'd)

- **Use the power curves from the graph on the previous slide to convert the other code languages to the equivalent in C++**
  - Find the Power curve for C++
    - Number of months = $a*(C{+}{+}\ ELOC)^b$
  - Find the Power curve for other language
    - Number of months = $c*(Other\ ELOC)^d$
  - So at a set number of months n
    - $n = a*(C{+}{+}\ ELOC)^b$, $n = c*(Other\ ELOC)^d$
    - $a*(C{+}{+}\ ELOC)^b = c*(Other\ ELOC)^d$
    - **$C{+}{+}\ ELOC = (c/a)^{(1/b)} *(Other\ ELOC)^{(d/b)}$**
- **The results were as follows:**
  - $C{+}{+}\ ELOC = 2.13 * (SQL\ ELOC)^{0.98}$
  - $C{+}{+}\ ELOC = 1.38 * (Java\ ELOC)^{0.98}$
  - $C{+}{+}\ ELOC = 1.87 * (HTML\ ELOC)^{0.92}$
  - $C{+}{+}\ ELOC = 0.16 * (C\ ELOC)^{1.17}$

Regression of Adjusted Data

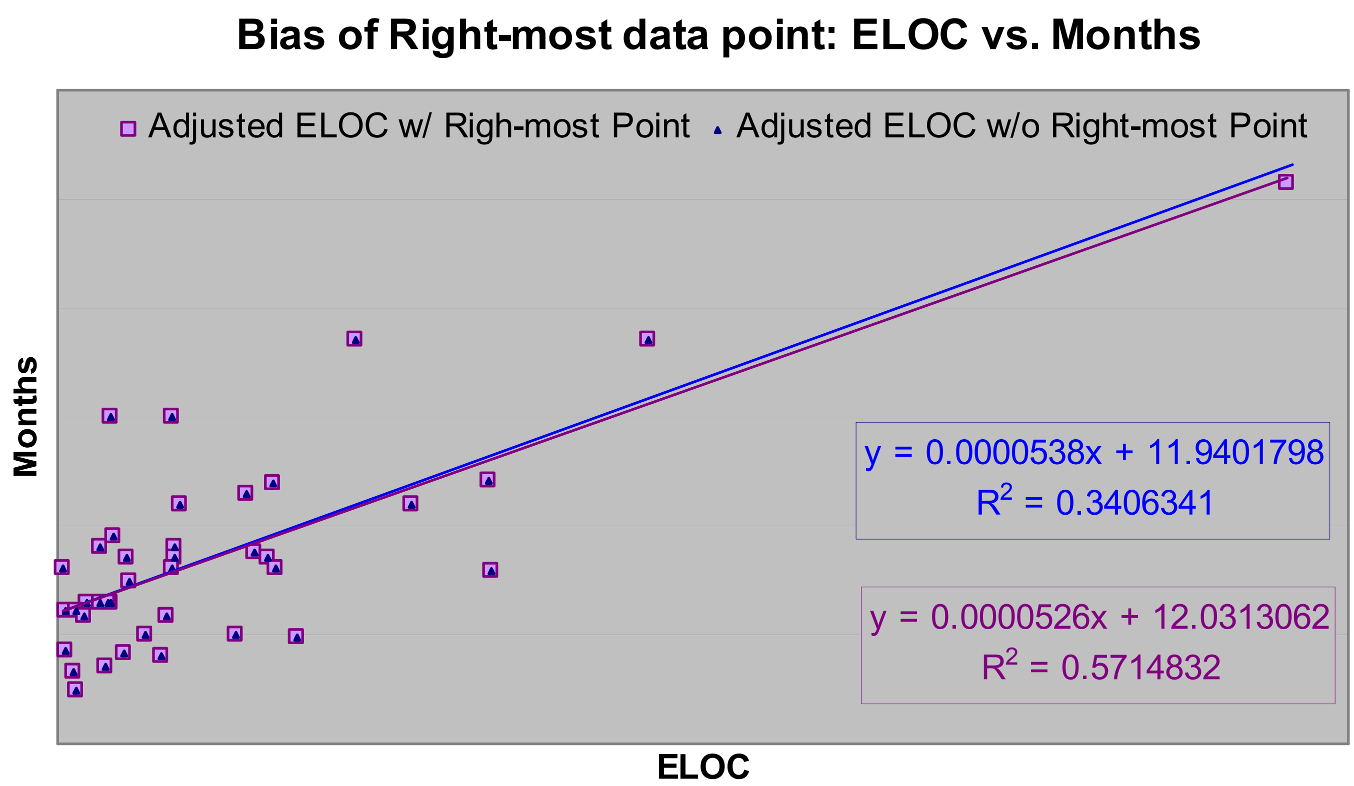
- **All C, HTML, Java & SQL ELOC were converted into equivalent C++ ELOC**
- **A linear relationship was found between ELOC and the Final Schedule**
- **Regression line equation: y = 5.26E-05*x + 12.03**
  - p-Value = 2.63E-08, statistically significant
  - Checked for bias from the right-most data point, no bias exists (Shown on the following slide)



Allison.converse@ngc.com June, 2007

Regression of Adjusted Data (cont'd)

- As expected, the R^2 decreases
- Without the right-most point the regression is still statistically significant
- The difference between the two equations is minimal

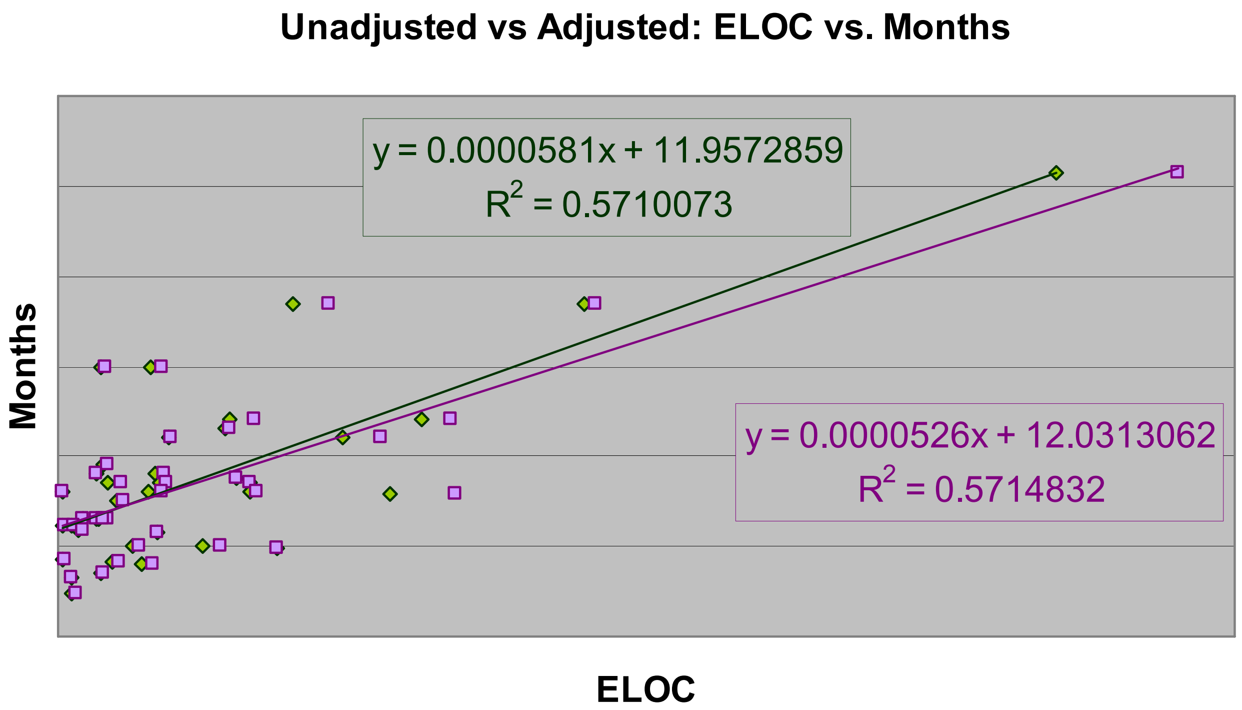


Allison.converse@ngc.com June, 2007

NORTHROP GRUMMAN

Unadjusted vs. Adjusted

- **Normalizing the data to a common language did not significantly alter the regression, but did improve it slightly**
  - Unadjusted R^2 of 0.5710 vs. Adjusted R^2 of 0.5715
- **A comparison of the two regressions can be seen below:**



Allison.converse@ngc.com June, 2007

NORTHROP GRUMMAN

Prediction Bands

- **A Confidence Interval determines the probability that a true value is within a certain range**
- **A Prediction Interval is a Confidence Interval for Y at a fixed X**
  - Since the Prediction Interval is for Y at a fixed X, the probability of the value being within a certain range includes the error in the coefficients of the regression equation in addition to the error of the equation itself
  - As a Prediction Interval accounts for more error than a Confidence Interval, the range of a Prediction Interval will be larger than the range of a Confidence Interval
- **A series of Prediction Intervals forms a Prediction Band about the Regression line**
  - The smallest range for a Prediction Interval will occur at the mean X value
  - This will be reflected in the Prediction Bands which will be arced about the regression
- **Used in the Schedule Realism Tool, Prediction Bands:**
  - Calculate the probability that the Proposed Schedule will be less than or equal to the Actual Schedule
  - Allow decision makers to propose a schedule with a desired confidence

Allison.converse@ngc.com

Prediction Band Calculation

$$\hat{Y} \pm t_{\alpha,df} * SEE * \sqrt{(n+1)/n + (X-X_{bar})^{2}/(\sum X^{2} - n * X_{bar}^{2})}$$

Where:

- $\hat{Y}$ = The y-value calculated from the regression line at the given x-value
- $t_{\alpha,df}$ = The Student t distribution
- SEE = Standard Error of the Estimate
- n = The number of observations
- X = The observed x-values
- X_{bar} = The mean of the observed x-values

Allison.converse@ngc.com

NORTHROP GRUMMAN

Schedule Realism Prediction Band Tool

- **The Inputs and Outputs of the tool can be seen in a screen shot of the tool on the following slide**
  - Inputs
    - Adjusted ELOC
      - It should be noted that using the tool with ELOC outside the range of data tends to provide very wide prediction bands and should be avoided
    - Proposed Start Date (Requirements Review)
    - Proposed End Date (PSR)
  - Outputs
    - Probability of achieving the proposed schedule
      - Probability that actual schedule is less than or equal to proposed schedule
    - The associated schedules (in months) for predetermined probabilities
      - The schedules are predicted by the tool based on the ELOC input
- **Methodology of the tool**
  - For the given ELOC, prediction intervals are calculated at every 0.005 α-level
  - The proposed schedule is then matched to an end point of a prediction interval and the corresponding α-level is determined
    - Below the regression line, the probability is 0.5 - (1-α)/2
    - Above the regression line, the probability is 0.5 + (1-α)/2

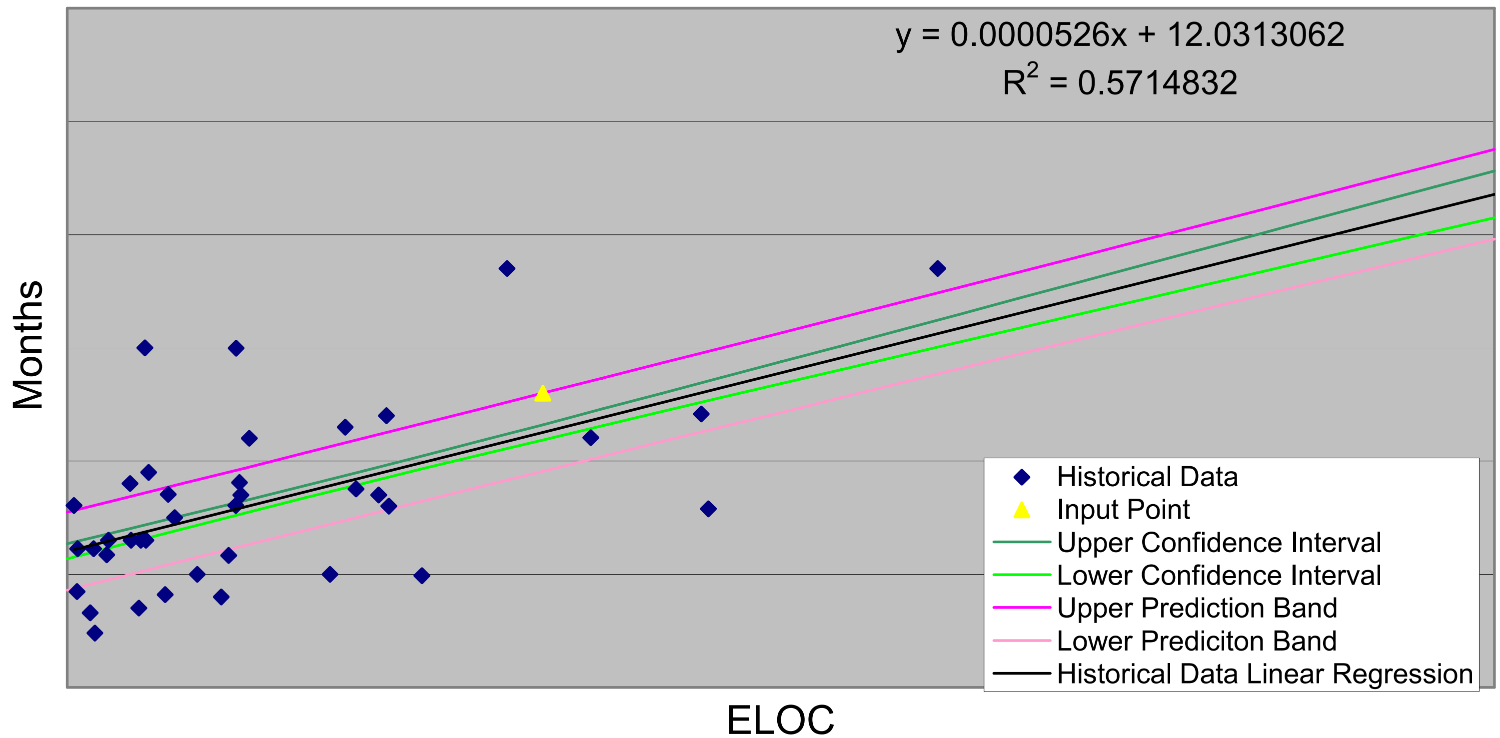
Allison.converse@ngc.com

NORTHROP GRUMMAN

Schedule Realism Prediction Band Tool Screenshot

Inputs				Outputs	
Release	**ELOC**	**Start Date**	**End Date (PSR)**	**Probability of Schedule**	**Proposed Schedule (Months)**
Release 6	200,000	3/1/2008	12/1/2010	70.25%	26.00

Outputs	
Probability	**Needed Schedule**
25%	18.17
40%	20.90
50%	22.54
60%	24.18
75%	26.92
90%	30.84



Allison.converse@ngc.com June, 2007

NORTHROP GRUMMAN

Application for Alternate Data Sets

- **The methodology used to create tool can be applied to any program if the right information is available**
- **Data required:**
  - Schedule Duration
  - A statistically significant schedule driver
  - The regression statistics from the regression

Allison.converse@ngc.com

NORTHROP GRUMMAN

Example 1

- **Example 1 – No Code Growth Applied**
  - Proposed Schedule = 18 Months
  - 90,000 C++ ELOC New Code
  - 10,000 Java ELOC Reused Code
  - 60% New Code Growth factor
- **Prediction Band Tool Results**
  - Convert ELOC to C++ equivalent
    - 90,000 + 10,000*Code adjustment function
      - 90,000 + 11,991
        - 101,991 ELOC
  - Input 101,991 ELOC and 18 Months into the tool
  - From the regression line 101,991 ELOC would have a 19.45 Month schedule
  - Calculate the intersecting Prediction Band
    - Probability of 39.75% of completing the release within 18 months

Allison.converse@ngc.com

Example 2

- **Example 2 – Code Growth Applied**
  - Proposed Schedule = 18 Months
  - 90,000 C++ ELOC New Code
  - 10,000 Java ELOC Reused Code
  - 60% New Code Growth factor
- **Prediction Band Tool Results**
  - Convert ELOC to C++ equivalent
    - (90,000*1.6) +10,000*Code adjustment function
      - 144,000 + 11,991
        - 155,991 ELOC
  - Input 155,991 ELOC and 18 Months into the tool
  - From the regression line 155,991 ELOC would have a 23.08 Month schedule
  - Calculate the intersecting Prediction Band
    - Probability of 18.75% of completing the release within 18 months.

Allison.converse@ngc.com

NORTHROP GRUMMAN

Future Research

- **Instead of length of schedule (months), use effort (hours) to measure the development time**
  - Calculate the prediction bands using hours from Contract Performance Report (CPR) data to develop the historical data set
- **Improve the range of the data to make the tool applicable to a wider range of release sizes**
  - Develop a separate tools for small releases
    - The historical data currently available does not include many segments with short schedule lengths
  - Tool is only as good as the scope of the historical data, so larger data points are needed to make the tool applicable to larger efforts
- **Consolidate the Schedule Realism Prediction Band Tool from this paper with the tool developed in the 2007 SCEA Paper: “Software Estimation Through the Use of Earned Value Data” (Jaekle, Greene, et al.) to produce a statistically based distribution of cost and schedule based on ELOC**

Allison.converse@ngc.com

NORTHROP GRUMMAN

Conclusions

- **For the AIS data set, a linear relationship exists between ELOC and Schedule**
  - Regression line equation: y = 5.81E-05*x + 11.96
- **Normalizing for code language does not impact the regression significantly**
  - Normalized regression line equation: y = 5.26E-05*x + 12.03
  - Improves the regression slightly but remains a linear relationship
- **The linear relationship of ELOC and Schedule is unexpected**
  - The commonly used equation for predicting software development effort (in this case schedule) is a power equation in the form of: effort = a*size^{b}
- **A tool was created which:**
  - Can be used for any data set that has a statistically significant schedule driver
  - Produces schedule distributions based on historical programs for similar programs
  - Calculates a prediction band for a proposed software development schedule based on ELOC, which allows the user to determine:
    - A suggested schedule length for a level of risk acceptable to decision makers
    - The probability that a proposed schedule will be met
    - Upside/Most Likely/Downside scenarios for the final schedule
    - A new schedule prediction as ELOC changes

Allison.converse@ngc.com

References

- ***"Schedule Realism Analysis,"*** **Blackburn, Chelson and Eng, AMC Study, April, 2002**
- ***"Schedule and Cost Growth,"*** **Coleman, Summerville and Dameron, SCEA, 2002**
- ***"Software Estimation Through the Use of Earned Value Data,"*** **Jaekle, Greene, et al., SCEA, 2007**