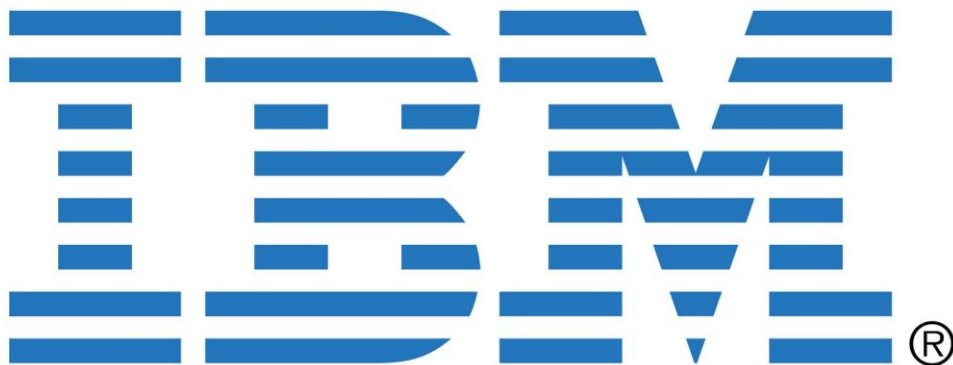


Research Paper WebEst

*Portierung des IBM Software Estimation Tool
als interaktive Webanwendung*



Ansprechpartner:

Andre Munzinger (a.munzinger@stud.tu-darmstadt.de)
Florian Friedrichs (friedrichsflorian@googlemail.com)
Dr. Holger Kremmin (holger.kremmin@de.ibm.com)

Stand: Montag, 28. Februar 2011

Inhaltsverzeichnis

1	Ziel des Dokumentes.....	1
2	Das COCOMO Modell.....	2
2.1	Komplexitätsklassen	2
3	Putnam Modell.....	3
3.1	PNR - Putnam Norden Rayleigh	4
3.2	Annahme – Operational Hours.....	5
3.3	Annahme – Special Skills Factor.....	5
3.4	Annahme – Gearing Factor.....	6
3.5	Annahme – PI - PP Approximation	6
3.6	Annahme - Constraints	7
3.7	COCOMO Modus.....	7
4	Berechnung.....	8
4.1	Berechnung der Source Lines Of Code (SLOC)	9
4.2	Berechnung des COCOMO Aufwandes.....	10
4.3	Berechnung der COCOMO Zeit.....	11
4.4	Berechnung des PUTNAM Aufwandes.....	12
4.5	Generierung der Rayleigh Curve	14
5	Glossar	16
6	Quellen	17

Abbildungsverzeichnis

Abbildung 1: PNR Curve.....	4
Abbildung 2: WebEst calculationmodel	8
Abbildung 3: Rayleigh Curve.....	14

Tabellenverzeichnis

Tabelle 1: Komplexitätsklassen COCOMO I Modell	2
Tabelle 2: Annahmen - Operational Hours	5
Tabelle 3: Annahmen - Special Skills Factor	5
Tabelle 4: Annahmen - Gearing Factor	6
Tabelle 5: Annahmen - PI PP Approximation	6
Tabelle 6: Annahmen - Constraints	7
Tabelle 7: MBI – MBP Zusammenhang	12

Formelverzeichnis

Formel 1: Putnam Effort.....	3
Formel 2: Berechnung Source Lines of Code	9
Formel 3: PI PP Approximation	9
Formel 4: Berechnung COCOMO Effort.....	10
Formel 5: Berechnung COCOMO Effort WebEst	10
Formel 6: Berechnung Estimated COCOMO Time.....	11
Formel 7: Berechnung Estimated COCOMO Time WebEst.....	11
Formel 8: Berechnung Putnam Effort mit Estimated COCOMO Time als Initialwert	12
Formel 9: Berechnung Manpower Buildup Parameter.....	12
Formel 10: Umrechnung der Estimated COCOMO Time (Years to Month)	15
Formel 11: Current-time Zusammenhang	15
Formel 12: Berechnung der Effort Time.....	15

1 Ziel des Dokumentes

Das nachfolgende Dokument beschreibt einen potentiellen Lösungsansatz zur Schätzung von Softwareprojekten, welcher in der zu entwickelnden WebEst Applikation als zu implementieren gilt.

Dazu werden zwei Schätzansätze zusammengeführt:

- Schätzung nach dem Constructive Cost Model
- Schätzung nach dem PUTNAM Model

Nach einer Erläuterung der beiden Schätzverfahren wird die neu entwickelte Schätzmethodik dargestellt und eruiert. Die Umsetzung erfolgte eins zu eins in beigefügter Excel File (WebEst_calculationmodel.xls).

2 Das COCOMO Modell

Das COCOMO Modell (Constructive Cost Model) wurde 1981 durch Barry W. Boehm, seinerzeit Softwareingenieur bei BOEING, entwickelt. Es fällt unter die Kategorie der modellbasierten Softwareschätzungsverfahren. Die Weiterentwicklung des COCOMO I Modells zeigte sich durch das ADA COCOMO Model (1987) sowie das COCOMO II Modell (1997).¹

Das im Folgenden verwendete Modell basiert auf COCOMO I. Dieses gliedert sich in drei Komplexitätsklassen:

2.1 Komplexitätsklassen

Die Komplexität des COCOMO I Modells gliedert sich in:

- Organic (einfache),
- Semi-detached (mittelschwere) und
- Embedded (komplexe) Projekttypen

Die folgend dargestellte Grafik illustriert die Projektkomplexität mit deren zugehörigen Parametern. Diese Werte basieren auf empirischen Untersuchungen und werden im Folgenden als gegeben angenommen.

Software Project	A	B	C	D
Organic / Simple	2,4	1,05	2,5	0,38
Semi-detached / Moderate	3,0	1,12	2,5	0,35
Embedded	3,6	1,20	2,5	0,32

Tabelle 1: Komplexitätsklassen COCOMO I Modell

¹ <http://www.wikipedia.de/COCOMO>

3 Putnam Modell

Das Putnam Modell stellt neben dem COCOMO Modell ein weiteres Modell-basiertes Softwareschätzungsverfahren dar. Seinerzeit 1978 von Larry W. Putnam publiziert, beschreibt es das Verhältnis zwischen Zeit (Time) und Aufwand (Effort) einer Software-Aufwands-Schätzung (Estimation). Das Putnam Modell ist unter anderem, neben einer großen Anzahl an statistischen Daten verschiedener IT-Projekte, die Basis für SLIM (Software Lifecycle Management) von QSM.²

Dieses Software Lifecycle Management von QSM findet unter anderem bei unserem Kunden IBM im Bereich Software Estimation Verwendung.

Nachfolgende Gleichung beschreibt den Zusammenhang der Parameter (siehe Abbildung 1).

$$\text{Effort} = \left[\frac{\text{Size}}{\text{Productivity} \cdot \text{Time}^{4/3}} \right]^3 \cdot B$$

Formel 1: Putnam Effort

Erläuterung der Parameter:

Effort:

Beschreibt den Aufwand in Personenjahren

Size:

Das Sizing beschreibt die Projektgröße in SLOC (Source Lines of Code)

Productivity:

Der Productivity Parameter gibt sich aus dem PI (Productivity Index). Dieser PI beschreibt die Produktivität des Umfeldes, in welchem das Projekt durchgeführt wird. Die konkrete Zuordnung ist unter Punkt 3.5 „PI-PP Approximation“ zu finden.

Time:

Beschreibt die Projektlaufzeit in Jahren

² ESI International – „Accurate Estimates Critical for Software Dev. PP III“

3.1 PNR - Putnam Norden Rayleigh

Das Putnam Norden Rayleigh, respektive die PNR Curve, stellt auf anschauliche Weise den Zusammenhang zwischen kumulierten Putnam Aufwand und der aktuellen Projektlaufzeit dar (siehe Abbildung 2).³

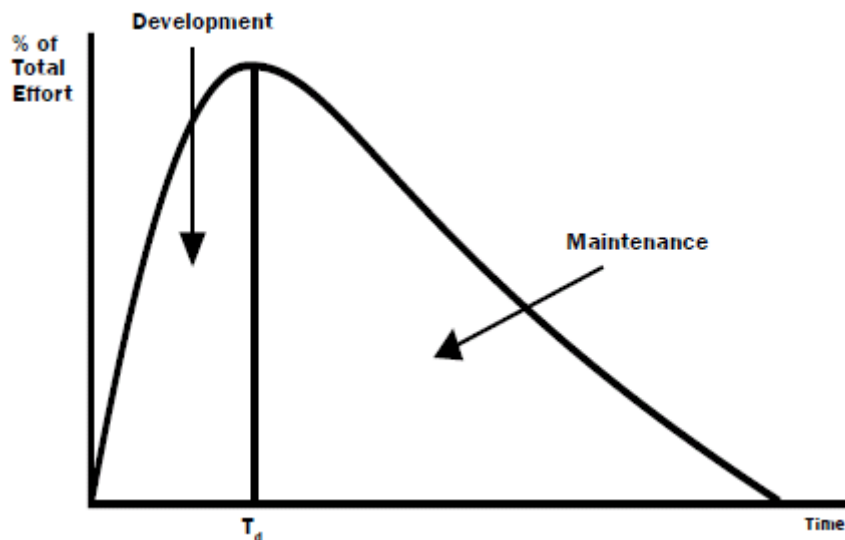


Abbildung 1: PNR Curve

Für die Beschreibung der Staffing Curve, respektive PNR Curve, stellt T_d den zentralen Parameter dar. T_d ist die mittels des COCOMO Modells ermittelte voraussichtliche Projektlaufzeit. Wie man aus der Grafik entnehmen kann, wird zu dem Zeitpunkt T_d die Maximal verfügbare Menge von Arbeitskräften benötigt.

T_d kann unter anderem auch als Estimated-Development-Time beschrieben werden. Die eigentliche Projektlaufzeit geht deutlich über die geschätzte Entwicklungszeit für das Projekt hinaus. Diese Phase wird als Maintenance, beziehungsweise Wartungs-/ Anpassungszeit bezeichnet.

³ http://www.wikipedia.de/Putnam_model

3.2 Annahme – Operational Hours

Ein COCOMO-Personenmonat oder auch Staff Month (SM) besteht aus 152 Arbeitsstunden (19 Arbeitstage * 8 Stunde). Dementsprechend kann eine Ressource 1824 Stunden pro Periode (12 Monate) tätig sein. Diese werden im Folgenden als gegeben angenommen, können aber in jeder Solution individuell angepasst werden.⁴

Employment	
working hours per day	8
working days per month	22
working months per year	3,6

Tabelle 2: Annahmen - Operational Hours

3.3 Annahme – Special Skills Factor

Die SLOC (Source Lines of Code) dienen als Ansatzpunkt zur Größeneinschätzung einer Softwareentwicklung. Der Wert E wird als Special Skills Factor bezeichnet und dient in Punkt 4.4 „Berechnung des Putnam Aufwandes“ zur korrekten Berechnung verwendet.⁴

Size(SLOC)	E
50000/20000	0,16
20000	0,18
30000	0,28
40000	0,34
50000	0,37
>50000	0,39

Tabelle 3: Annahmen - Special Skills Factor

⁴ Liebowitz, Jan (1999). "Knowledge Management Handbook"

3.4 Annahme – Gearing Factor

Der Gearing Factor dient als Berechnungsgrundlage zur Ermittlung der Size (SLOC – Source Lines of Code). Der Wert 125 wird im Folgenden als gegeben angenommen, kann jedoch in jeder Solution individuell angepasst werden.⁴

Gearing Factor
125

Tabelle 4: Annahmen - Gearing Factor

3.5 Annahme – PI - PP Approximation

Der PI (Productivity Index) bezeichnet den Produktivitätsindex. Der PI liegt im Bereich zwischen 1 und 40. Zur Veranschaulichung sind Teile der Approximation (lineare Regression) dargestellt. Die genaue Regressionsanalyse ist beigelegten Excel Sheet (WebEst_calculationmodel.xls) zu finden.⁴

PI	PP ~
1	764
5	2001
10	6667
15	22219
20	74045
25	246756
30	822314

Tabelle 5: Annahmen - PI PP Approximation

⁴ Liebowitz, Jan (1999). "Knowledge Management Handbook"

3.6 Annahme - Constraints

Die Constraints beziehen sich diesem Modell lediglich auf die Projektlaufzeit. Die minimale Projektlaufzeit (in Jahren) beträgt 0,75. Die maximale Projektlaufzeit beträgt 3 Jahre.⁴

Constraints	Years
Min Time (Estimated)	0,75
Max Time (Estimated)	3,00

Tabelle 6: Annahmen - Constraints

3.7 COCOMO Modus

Aus Komplexitätsgründen findet weiterhin nur der „Organic“ – COCOMO Modus Berücksichtigung.

Dieser beschreibt den COCOMO Aufwand bzw. die Zeit von kleinen bis mittelgroßen Projekten. Dies trifft unter anderem auf Projekte mit <50K SLOC zu. Darüber hinaus wird bei der Verwendung des Organic COCOMO Modus angenommen, dass das Projektumfeld mit der eingesetzten Hard-/ und Software vertraut ist und sich bereits zuvor erfolgreich mit ähnlichen Projekten auseinander gesetzt hat. Die Fixierung des Schätzmodells auf den COCOMO Organic Modus wird durchgeführt, um die Einführung von Justagefaktoren in dem zuvor beschriebenen Putnam Modell zu verhindern, da keine empirisch gesicherten Werte zur Verfügung stehen. Die Einführung von Justage Faktoren für die Berechnung des Putnam Aufwands für die drei verschiedenen COCOMO Modi ist notwendig, da das COCOMO Modell einen gestiegenen Aufwand durch eine längere Zeitdauer ausdrückt. Da jedoch die COCOMO Zeit als Initial Wert für die Putnam Aufwand Gleichung verwendet wird und dort die Zeit im Nenner der Gleichung steht, schrumpft der ausgewiesene Putnam Aufwand mit einer länger angenommenen Initial - Zeit. Somit wird die Einführung eines Anpassungs-/ Justagefaktors notwendig.

⁴ Liebowitz, Jan (1999). "Knowledge Management Handbook"

4 Berechnung

Die Berechnung beschreibt Schritt für Schritt den eigentlichen Schätzprozess unter Berücksichtigung der in Punkt 3 getroffenen Annahmen und Voraussetzungen.

Das WebEst Calculation Model (siehe Abb. 3) stellt den hier aufgebauten Berechnungsprozess zur Verfügung. In diesem werden die folgenden Berechnungsschritte berechnet.

- Berechnung der Source Lines Of Code (SLOC)
- Berechnung des COCOMO Aufwandes
- Berechnung der COCOMO Zeit
- Berechnung des PUTNAM Aufwandes
- Generierung der Rayleigh Curve

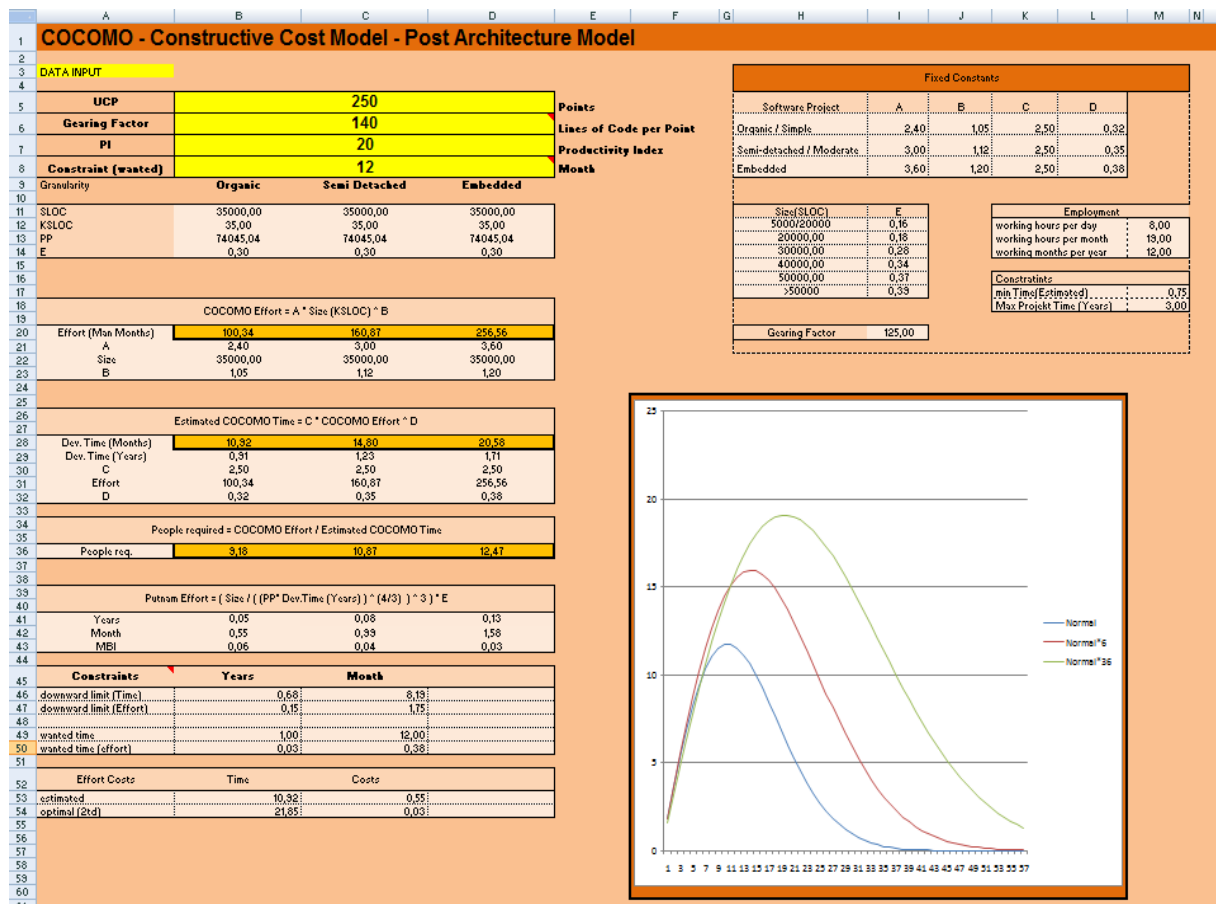


Abbildung 2: WebEst Calculationmodel

4.1 Berechnung der Source Lines Of Code (SLOC)

Als allgemeine Inputgrößen werden folgende Werte benötigt:

- **UCP** (Use Case Points)
- **Gearing Factor** (Default 125)
- **PI** (Productivity Index) [1;40]

$$SLOC = UCP * Gearing Factor (125)$$

Formel 2: Berechnung Source Lines of Code

Falls ein Productivity Parameter eingegeben wurde, wird dieser Wert direkt in das Schätzmodell übernommen. Falls ein Productivity Index ausgewählt wurde, findet automatisch eine entsprechend Approximation statt.

$$Productivity_{Parameter} = PI_{Value}(approximiert)$$

Formel 3: PI PP Approximation

4.2 Berechnung des COCOMO Aufwandes

$$COCOMO_{Effort} = A * Size(KSLOC)^B$$

Formel 4: Berechnung COCOMO Effort

- Die Parameter A und B sind abhängig von der Komplexität des jeweiligen Projektes
- (siehe Punkt 2 - Komplexitätsklassen)
- Für Size ist als Metrik KSLOC zu wählen (Codezeilen * 1000)

Hinweis:

Da unter Punkt 3.7 „COCOMO MODUS“ der Organic Modus als nachfolgend zu verwenden gilt, entsteht folgender Zusammenhang:

$$COCOMO_{Effort} = 2,4 * Size(KSLOC)^{1,05}$$

Formel 5: Berechnung COCOMO Effort WebEst

4.3 Berechnung der COCOMO Zeit

$$\textit{EstimatedCOCOMO}_{Time} = C * \textit{COCOMO_Effort}^D$$

Formel 6: Berechnung Estimated COCOMO Time

- Die Parameter C und D sind wiederum abhängig von der Komplexität (siehe Punkt 2.1 „Komplexitätsklassen“)
- Der COCOMO_Effort wurde bereits in Punkt 4.2 „Berechnung des COCOMO Aufwandes“ berechnet

Hinweis:

Da unter Punkt 3.7 „COCOMO MODUS“ der Organic Modus als nachfolgend zu verwenden gilt, entsteht folgender Zusammenhang:

$$\textit{EstimatedCOCOMO}_{Time} = 2,5 * \textit{COCOMO_Effort}^{0,32}$$

Formel 7: Berechnung Estimated COCOMO Time WebEst

4.4 Berechnung des PUTNAM Aufwandes

Die in 4.3 ermittelte Estimated COCOMO Time dient nun als Initialwert zur Berechnung des Putnam Aufwandes.

$$\text{Putnam}_{\text{Effort}} = \left(\frac{\text{Size}}{(PP * (\text{EstimatedCOCOMO}_{\text{Time}}^{4/3}))^3} \right) * E$$

Formel 8: Berechnung Putnam Effort mit Estimated COCOMO Time als Initialwert

- Size beschreibt die Source Lines of Code (SLOC)
- PP ermittelt sich aus dem Productivity Index (Punkt 3.5 „PI PP Approximation“)
- E bestimmt sich aus dem Special Skills Factor (Punkt 3.3 „Special Skills Factor“)

Weitere Putnam Implikationen:

$$MBP = \frac{\text{Putnam}_{\text{Effort}}}{\text{EstimatedCOCOMO}_{\text{Time}}^3}$$

Formel 9: Berechnung Manpower Buildup Parameter

Der MBP (Manpower Buildup Parameter) beschreibt die Personalaufbaurrate eines Projektes. Die von Putnam entwickelte Messgröße reicht für verschiedene Projekte von einem einstelligen Wert bis in die Hunderte. Um eine Auswertung des MBP für verschieden Projekte möglich zu machen, wurden von Putnam sechs MBP Level fixiert und einem MBI, dem Manpower Buildup Index zugeordnet. Diese sechs MBP Niveaus werden in der folgenden Tabelle aufgeführt.⁴

MBI	MBP	Rate of Buildup
1	7.3	Slow
2	14.7	Moderately slow
3	26.9	Moderate
4	55	Rapid
5	89	Very rapid
6	233	Extremely rapid

Tabelle 7: MBI – MBP Zusammenhang

⁴ Liebowitz, Jan (1999). "Knowledge Management Handbook"

Aus dem MBP lassen sich verschiedene Projekteigenschaften ableiten. So weisen große Werte darauf hin, dass versucht wird, die Projektlaufzeit durch den Mehreinsatz von Personal zu reduzieren.

Kleine Werte, wie zum Beispiel ein MBP von eins kann dementsprechend bedeuten, dass es sich um ein sehr komplexes Projekt handelt, bei welchem die Probleme sequentiell gelöst werden müssen und ein mehr von Personal diesen Prozess nicht beschleunigen kann. Jedoch kann ein niedriger MBP aber auch auf ein sehr knappes Budget oder aber auf Personalengpässe oder Limitierte Ressourcen hinweisen.

4.5 Generierung der Rayleigh Curve

Die $\text{Effort}_{\text{Time}}$ Gleichung beschreibt die Rayleigh Distribution und dient zur Berechnung der Staffingcurve für den Report der WebEst Applikation. Diese Kurve weist zur jedem Zeitpunkt des Projekts die voraussichtliche Personalstärke aus. Entwickelt wurde die Kurve von Norden, dem während der Analyse des Personalverlaufs einiger Forschungs und Entwicklungsprojekte die Ähnlichkeit des Verlaufs mit der aus der Statistik bekannten Rayleigh Kurve aufgefallen ist. Somit entwickelte er auf Basis der bereits erwähnten Rayleigh Kurve die folgende Personal-Verteilungsfunktion Approximation wie sie in Abbildung 4 dargestellt ist.⁷

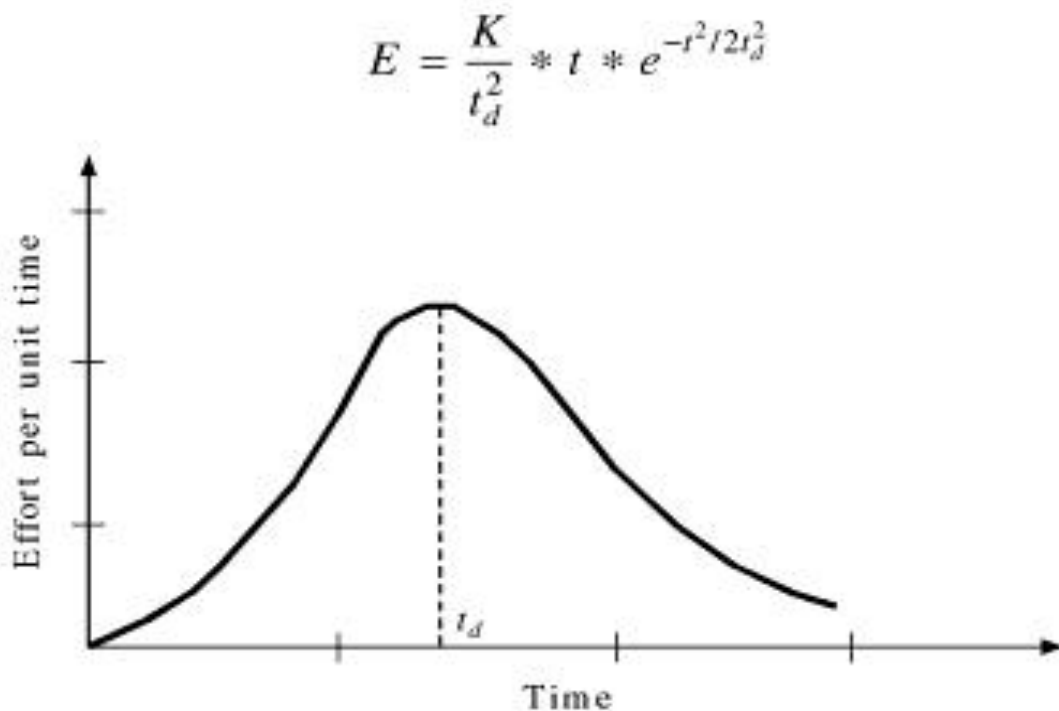


Abbildung 3: Rayleigh Curve

Erläuterung der Parameter:

E: Aufwand in Personen, zum Zeitpunkt t

K: Fläche unterhalb der Kurve (der Gesamte Aufwand)

t_d: Geschätzter Projektaufwand (Estimated COCOMO Time)

t: Bereits verstrichene Zeit

⁷ Mall, Rajib (2004), "Fundamentals of Software Engineering"

Diese von Norden entwickelte Approximation der Staffingcurve wurde nun an die uns zur Verfügung stehenden Parameter angepasst, womit sich die folgende Gleichung auf Basis von COCOMO und Putnam ergibt. Es sei noch zu erwähnen, dass sämtliche in der Gleichung verwendeten Zeiteinheiten auf Jahre umgerechnet wurden. Für die $EstimatedCOCOMO_{Time}$ oder $current_{Time}$ bedeutet dies beispielsweise:

$$EstimatedCocomoTime = \frac{EstimatedCocomoTime}{12}$$

Formel 10: Umrechnung der Estimated COCOMO Time (Years to Month)

$$currentTime = \frac{i}{12} \quad i \in \{1, 2, \dots, 3td\}$$

Formel 11: Current-time Zusammenhang

Diesen Zusammenhang erläutert nun folgende Gleichung:

$$Effort_{Time} = \left(\frac{Putnam_{Effort}}{EstimatedCOCOMO_{Time}^2} \right) * current_{Time} * e^{\left(\frac{current_{Time}^2}{(2 * EstimatedCOCOMO_{Time})^2} \right)}$$

Formel 12: Berechnung der Effort Time

Die von Norden entwickelte Rayleigh Kurve eignet sich darüber hinaus nicht nur zu der Schätzung des Personalverlaufs von Software Projekten, sondern für sämtliche Forschungs- und Entwicklungsprojekte.

5 Glossar

COCOMO	Constructive Cost Model
PI	Productivity Index
PP	Productivity Parameter
SLOC	Source Lines of Code
KSLOC	Thousand Source Lines of Code
UCP	Use Case Points
MBI	Manpower Buildup Index
MBP	Manpower Buildup Parameter
PNR	Putnam Norden Rayleigh

6 Quellen

Der aktuelle Stand der Verlinkungen von Webseiten bezieht sich auf den 27.02.2011.

- [1] <http://de.wikipedia.org/wiki/COCOMO>
- [2] ESI International – „Accurate Estimates Critical for Software Dev. PP III“ (2001/12 Volume3, Number8)
- [3] http://en.wikipedia.org/wiki/Putnam_model
- [4] Liebowitz, Jan (1999). "Knowledge Management Handbook"
- [5] WebEst Calculation Model Excel Sheet (WebEst_calculationmodel.xlsx)
- [6] Putnam, Lawrence H. (1978). "A General Empirical Solution to the Macro Software Sizing and Estimating Problem". IEEE transactions on software engineering, VOL. SE-4, NO. 4, pp 345-361.
- [7] Mall, Rajib (2004), "Fundamentals of Software Engineering"