

بسم الله الرحمن الرحيم

فصل ششم

مدیریت کنترل پروژه فناوری اطلاعات

حاجی صادق

- در این فصل به تعریف کارها یا فعالیتهایی که برای کامل کردن تمامی موارد تحویل دادنی مورد انتظار مربوط به محدوده باید انجام شوند می پردازیم.

# مدیریت زمان پروژه شامل موارد زیر است:

□ تعریف فعالیت

□ توالی فعالیت

□ تخمین منابع فعالیت ها

□ تخمین مدت زمان فعالیت ها

□ توسعه زمان بندی

□ کنترل زمان بندی

# Project Time Management

## PMBOK®



- **تعریف فعالیت ها**
  - Identifying what activities must be completed to produce the project scope deliverables
- **توالی فعالیت ها**
  - Determining whether activities can be completed sequentially or in parallel and any dependencies that may exist among them
- **تخمین منابع فعالیت ها**
  - Identifying the type of resources (people, technology, facilities, etc.) and the quantity of resources needed to carry out project activities
- **تخمین مدت زمان فعالیت ها**
  - Estimating the time to complete each activity
- **توسعه زمان بندی**
  - Based on the availability of resources, the activities, their sequence, and time estimates, a schedule for the entire budget can be developed
- **کنترل زمان بندی**
  - Ensuring that proper processes and procedures are in place in order to control changes to the project schedule

# Work Breakdown Structure (WBS)

پس از آنکه محدوده پروژه تعریف شد.

**قدم بعدی:**

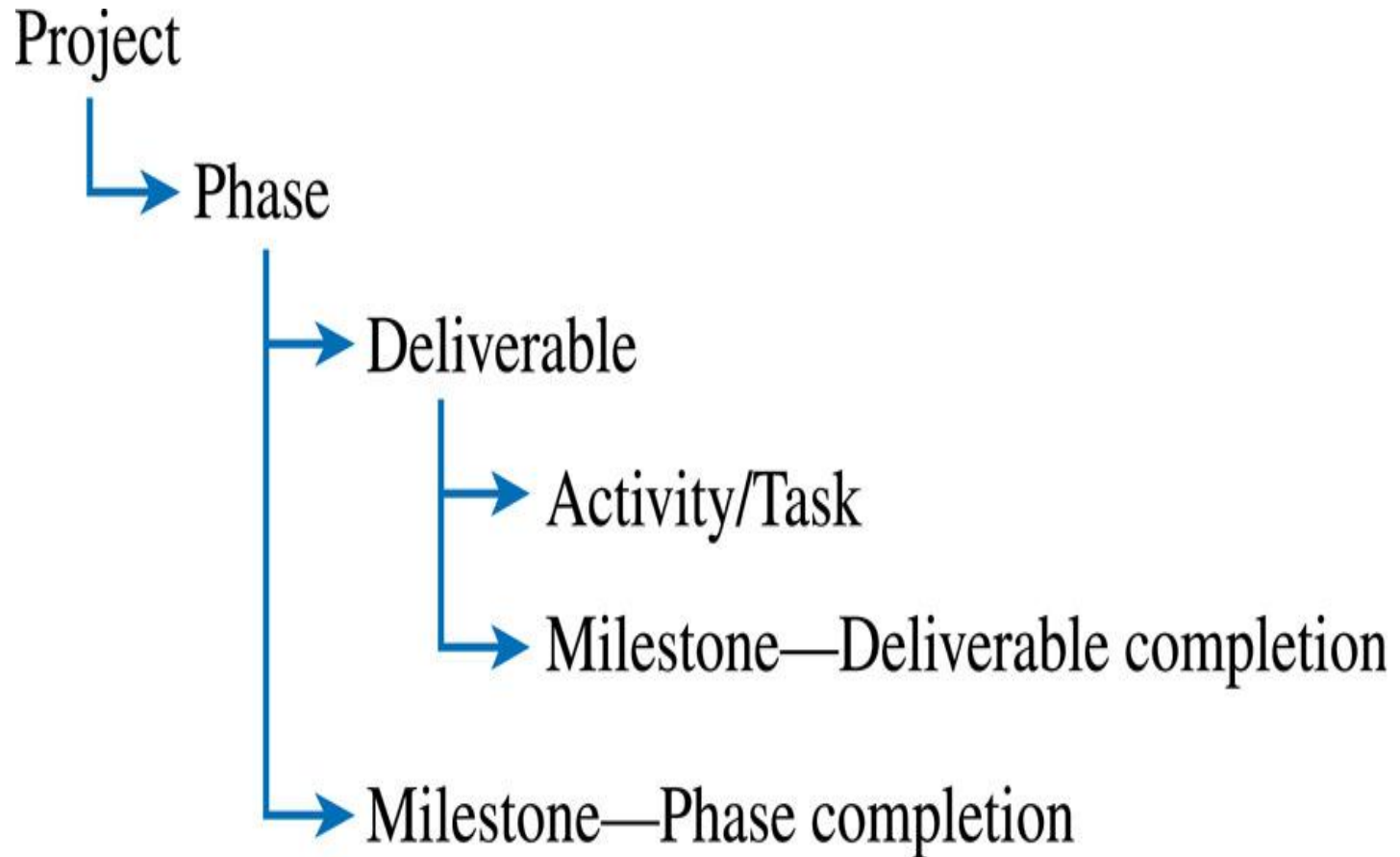
تعریف فعالیت ها و یا کارهایی است که اعضای تیم باید برای برآوردن نیازهای محدوده موارد تحویل دادنی انجام دهند.

ساختار شکست کار یک ابزار کاربردی برای ایجاد طرح پروژه است و محدوده پروژه را به زمان بندی و بودجه پیوند می دهد.

# Work Package

- ▶ Wbs پروژه را به اجزای کوچک تر و واحدهای کنترل پذیرتر کاری به نام بسته های کاری، تجزیه و یا تقسیم می کند.
- ▶ یک بسته کاری می تواند به عنوان یک سلسله مراتب نمایش داده شود که با خود پروژه شروع می شود.
- ▶ سپس پروژه به مراحل تجزیه می شود که هر فاز، همانطور که در جدول تعریف موارد تحویل دادنی و نمودار ساختار موارد تحویل دادنی تعریف شده، یک یا چند مورد تحویل دادنی دارد.
- ▶ هر فاز باید حداقل یک مورد تحویل دادنی خاص تولید کند، یعنی یک قطعه کاری محسوس و مشخص

# Work Package



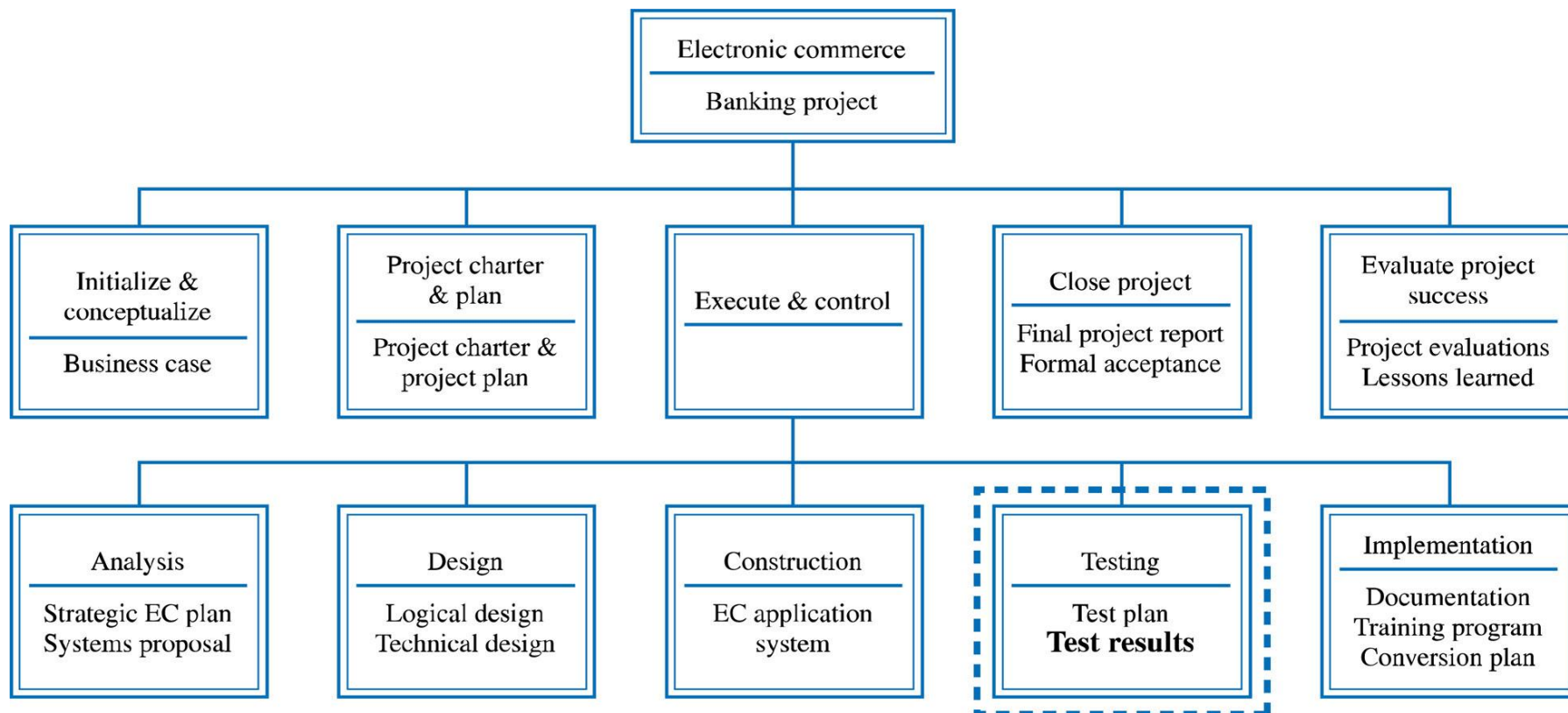
# موارد تحویل دادنی در مقابل نقاط عطف

- موارد تحویل دادنی
  - Tangible, verifiable work products
    - Reports, presentations, prototypes, etc.
- نقاط عطف
  - Significant events or achievements that provides evidence that the deliverable has been completed or that a phase is formally over
  - Focuses on the achievement, not the deliverable
    - Deliverable – prototype
    - Milestone – stakeholder's formal acceptance of the user interface
  - Smaller, shorter term deliverables keep the team focused
  - Cruxes (proof of concepts)
    - Successfully use a piece of software for the first time on small set of data, validates proof of concept for expanding to full blown database
  - Quality control
    - No user acceptance, no moving forward. The deliverable must be done right.



# Developing the WBS

- A work package is developed for each of the phases and deliverables defined in the Deliverable Structure Chart (DSC)
- Focus on Testing box with deliverable of Test Plan and Test Results



# مورد تحویل دادنی: گزارش نتایج تست

- فعالیتهای منطقی برای ایجاد سند نتایج آزمون:

۱. طرح آزمون را با کارآور مرور کنیم تا برای ذینفعان اصلی مشخص شود که چه موردی آزموده می شود. (چگونه و چه زمانی)
۲. آزمون های برنامه ریزی شده در طرح آزمون را انجام می دهیم.
۳. هنگامی که نتایج آزمون را گردآوری کردیم، باید آنها را تحلیل کنیم.
۴. خلاصه کردن نتایج تحلیل به صورت گزارش برای کار آور
۵. تایید نتایج آزمون توسط کارآور در صورتی که همه کارها خوب پیش رود.

**What are the deliverables? Milestones?**

# Example Work Breakdown Schedule

–0.0 EC Bank Project

+1.0 Conceptualize & initialize project

+2.0 Develop charter & plan

+3.0 Analysis

+4.0 Design

+5.0 Construction

–6.0 Testing

+6.1 Test plan

–6.2 Test results report

6.2.1 Review test plan with client

6.2.2 Carry out test plan

6.2.3 Analyze results

6.2.4 Prepare test results report and presentation

6.2.5 Present test results to client

6.2.6 Address any software issues or problems

6.2.7 **Milestone:** client signs off on test results

+6.3 **Milestone:** testing completed

+7.0 Implementation

+8.0 Close project

+9.0 Evaluate project success

# The WBS Should Follow the Work Package Concept

- 0.0 EC Bank Project
  - +1.0 Conceptualize & initialize project
  - +2.0 Develop charter & plan
  - +3.0 Analysis
  - +4.0 Design
  - +5.0 Construction

- 6.0 Testing

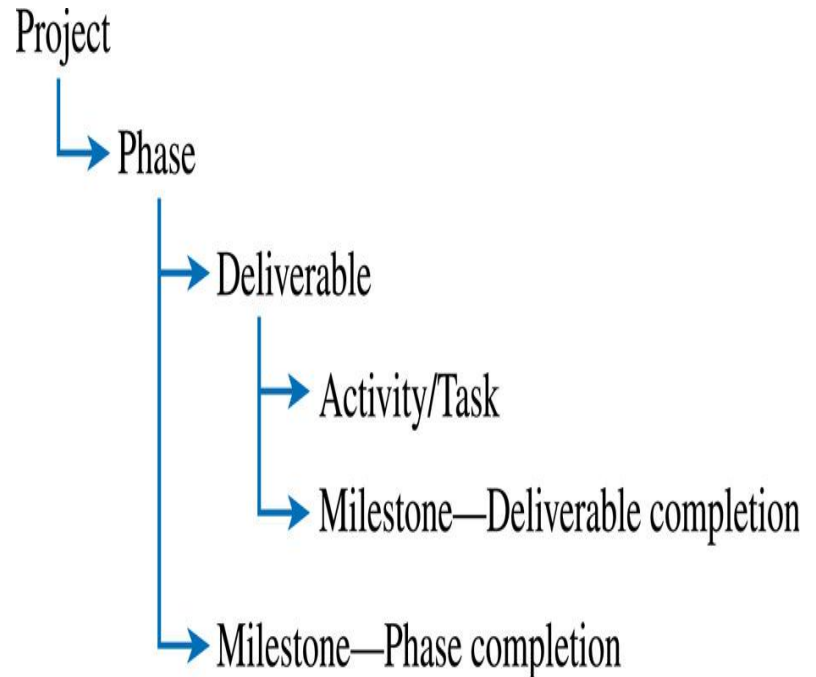
- +6.1 Test plan

- 6.2 Test results report

- 6.2.1 Review test plan with client
      - 6.2.2 Carry out test plan
      - 6.2.3 Analyze results
      - 6.2.4 Prepare test results report and presentation
      - 6.2.5 Present test results to client
      - 6.2.6 Address any software issues or problems
      - 6.2.7 **Milestone:** client signs off on test results

- +6.3 **Milestone:** testing completed

- +7.0 Implementation
  - +8.0 Close project
  - +9.0 Evaluate project success



# The WBS...

- ساختار شکست کار باید مبتنی بر مورد تحویل دادنی باشد.
- ساختار شکست کار باید از MOV پروژه پشتیبانی کند.
- سطح جزئیات باید از برنامه ریزی و کنترل پشتیبانی نماید.
- ساختار شکست کار باید افرادی را که کارها را انجام می دهند، دخالت دهد.
- چرخه های یادگیری و درس های آموخته شده، می تواند از ایجاد یک ساختار شکست کار پشتیبانی نماید.

# سوالات تخمین



**What are you going to estimate?**



**Where do you start?**



**How do you estimate?**

# تکنیک‌های تخمین زدن: روش‌های سنتی

یکی از کارهای حیاتی و مهم در مدیریت پروژه

- تخمین حدسی
- تکنیک دلفی
- محدودیت زمانی
- تخمین بالا به پایین
- تخمین پایین به بالا



# Guestimating

- ▶ Estimation by guessing or just picking numbers out of the air is not the best way to derive a project's schedule and budget.
- ▶ Unfortunately, many inexperienced project managers tend to **guesstimate**, or guess at the estimates, because it is quick and easy.
- ▶ When put on the spot to give an estimate, give a range of time and cost and say that more research will enable a more confident estimate



# Delphi Technique

## تکنیک دلفی

- Involves multiple, anonymous experts
- Each expert makes an estimate
- Estimates compared
  - If close, can be averaged
  - If not, do another iteration until consensus is reached
- Can take longer than most other estimation methods, but can be very effective and provide reasonable assurance

# Time Boxing

## محدودیت زمانی



- A “box” of time is allocated for a specific activity, task, or deliverable
  - May be based on a requirement that can’t be changed
- Can focus a team if used effectively
- Can demoralize a team if not used effectively
  - May result in long hours and pressure
  - May not succeed

# Top-Down

- Mandate from above has predetermined the time and cost of the project (competitor, satisfy client)
- Top & middle managers determine overall project schedule &/or cost
- Lower level managers are expected to breakdown schedule/budget estimates into specific activities (WBS)
  - Previous project experience can help in allocating time to activities



# Top-Down

- When top-down estimation is done by people independent of the project it may be overly optimistic or overly aggressive
  - Death March project
    - Project schedule has been compressed 50% or more
    - Staff has been reduced by 50% or more
    - Budget and resources have been reduced by 50% or more
    - Functionality, features or other requirements are twice what they should be under typical circumstances
  - Can force the PM to examine the project's risks more closely so that a specific budget or schedule target can be achieved

# Bottom-Up

- Schedules & budgets are constructed from WBS
- Starts with people who will be doing the work
- Schedules & budgets are the aggregate of detailed activities & costs
- **Analogous estimation** - use information from previous, similar projects as a basis for estimation
  - BUT – is everything else the same?



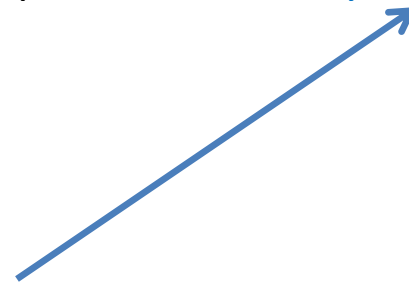
# Parametric Modeling

- Use project characteristics (parameters) in a mathematical model to estimate
- Example: \$50/ LOC based on:
  - Programming language
  - Level of expertise
  - Size & complexity

# Estimates are made for each activity in the WBS

## 6.2 Test Results Report

6.2.1 Review test plan with client	1 day
6.2.2 Carry out test plan	5 days
6.2.3 Analyze results	2 days
6.2.4 Prepare test results report and presentation	3 days
6.2.5 Present test results to client	1 day
6.2.6 Address any software issues or problems	5 days



How did we come up with these estimates? Using a technique, or combination of techniques, with the exception of guestimating!

## روش ها و متریک های مهندسی نرم افزار

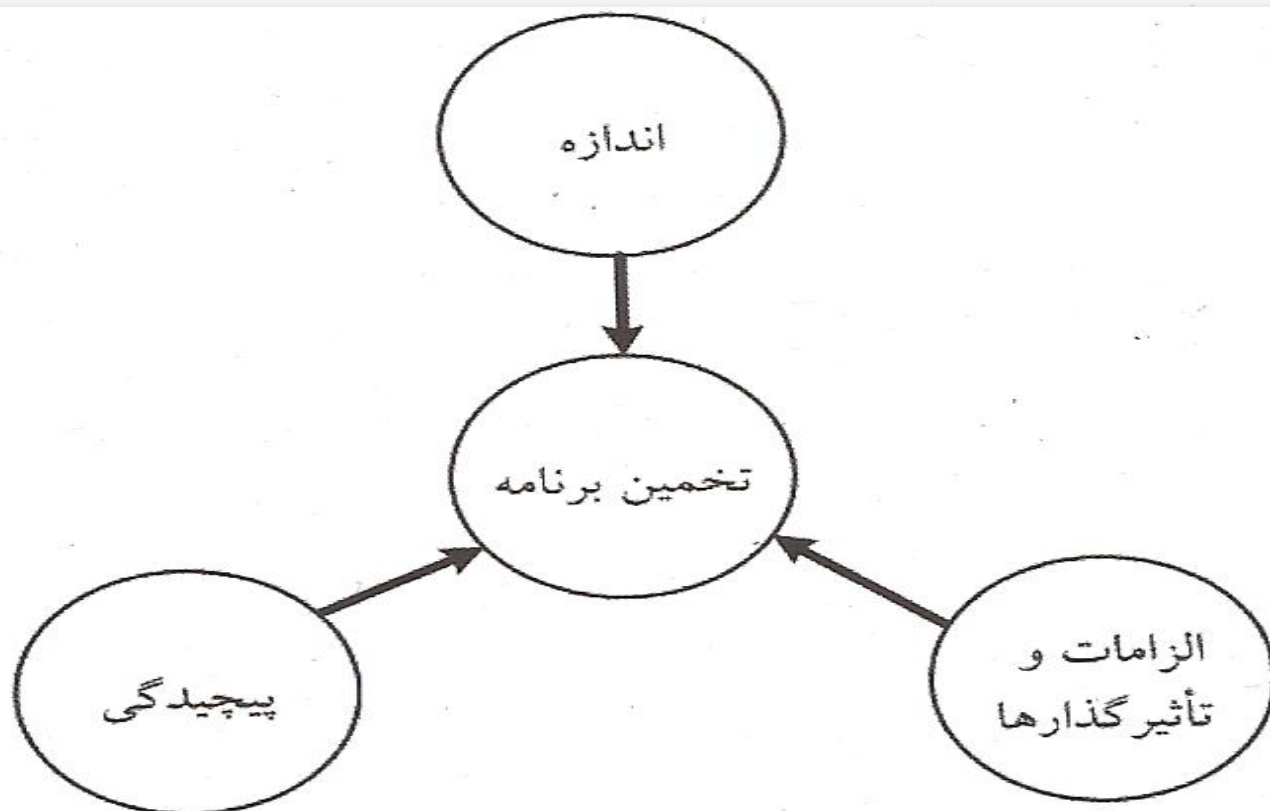
- Lines of Code (LOC) (خطوط کد)
- Function Points (نقاط تابعی)
- COCOMO (کوکومو)
- Heuristics (روش های مکاشفه ای)

Software engineering techniques focus on estimating the size of the system to be developed



## خطوط کد (LOC)

شمارش تعداد خطوط کد در زمان‌بندی‌های کامپیوتری، سنتی‌ترین، پرکاربردترین همچنین، بحث برانگیزترین متریک نرم‌افزاری، برای اندازه‌گیری محصول زمان‌بندی کاربردی است.



شکل ۴-۶ مدل تخمین مهندسی نرم‌افزار

یک زمان بندی جاوا با ۱۰۰۰ خط کد ۱۰ برابر بزرگتر از یک زمان بندی جاوا با ۱۰۰ خط کد است. اما شمارش خطوط کد به هیچ عنوان اسان نیست.

- **اولا چه چیزی خط کد محسوب می شود؟**
- **ایا توضیحات را در نظر می گیریم؟**

## □ نقاط تابعی

مشکلات ذاتی خطوط کد به عنوان یک متریک برای تخمین  
وبهره وری نیاز را برای یک متریک نرم افزاری بهتر فراهم کرد.

**الن البرخت** از شرکت ای.بی.ام **ایده نقاط تابعی** را پیشنهاد  
داد.

**نقاط تابعی** یک متریک مصنوعی شبیه به متریک هایی که  
هرروز استفاده می شوند مثل ساعت، کیلو، تن، مایل  
دریایی، درجه و سلیوس و... است. هرچند نقاط تابعی بر عملکرد و  
پیچیدگی یک سیستم کاربردی یا بخش خاصی از یک  
برنامه، تمرکز می کند.

**نکته:** نقاط تابعی مستقل از فناوری می باشند.

- امکان مقایسه برنامه های گوناگونی از زبان های برنامه نویسی یا استانداردهای سخت افزاری یکسان و یا متفاوت
- تحلیل نقطه تابعی قابل اعتماد است. شمارش نقاط تابعی نسبتا آسان است.

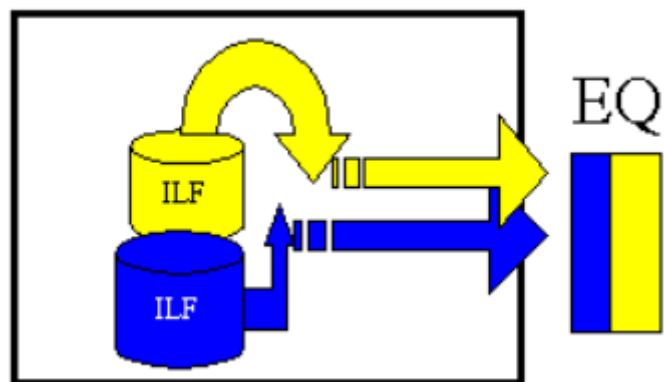
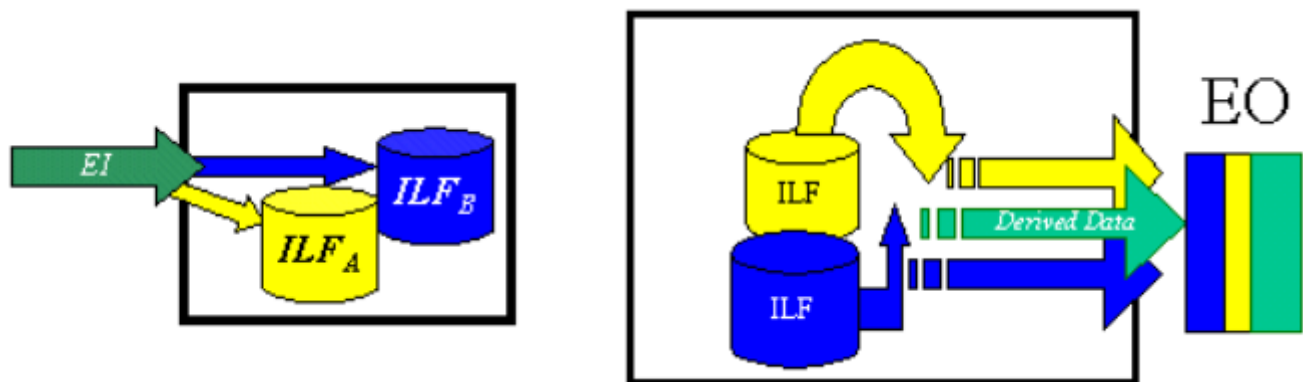
- سازمان اصلی نقاط تابعی:

- ۱- تیم بین المللی کاربران نقطه تابعی

- ۲- تیم کاربران نقطه تابعی انگلیس (UFUG)

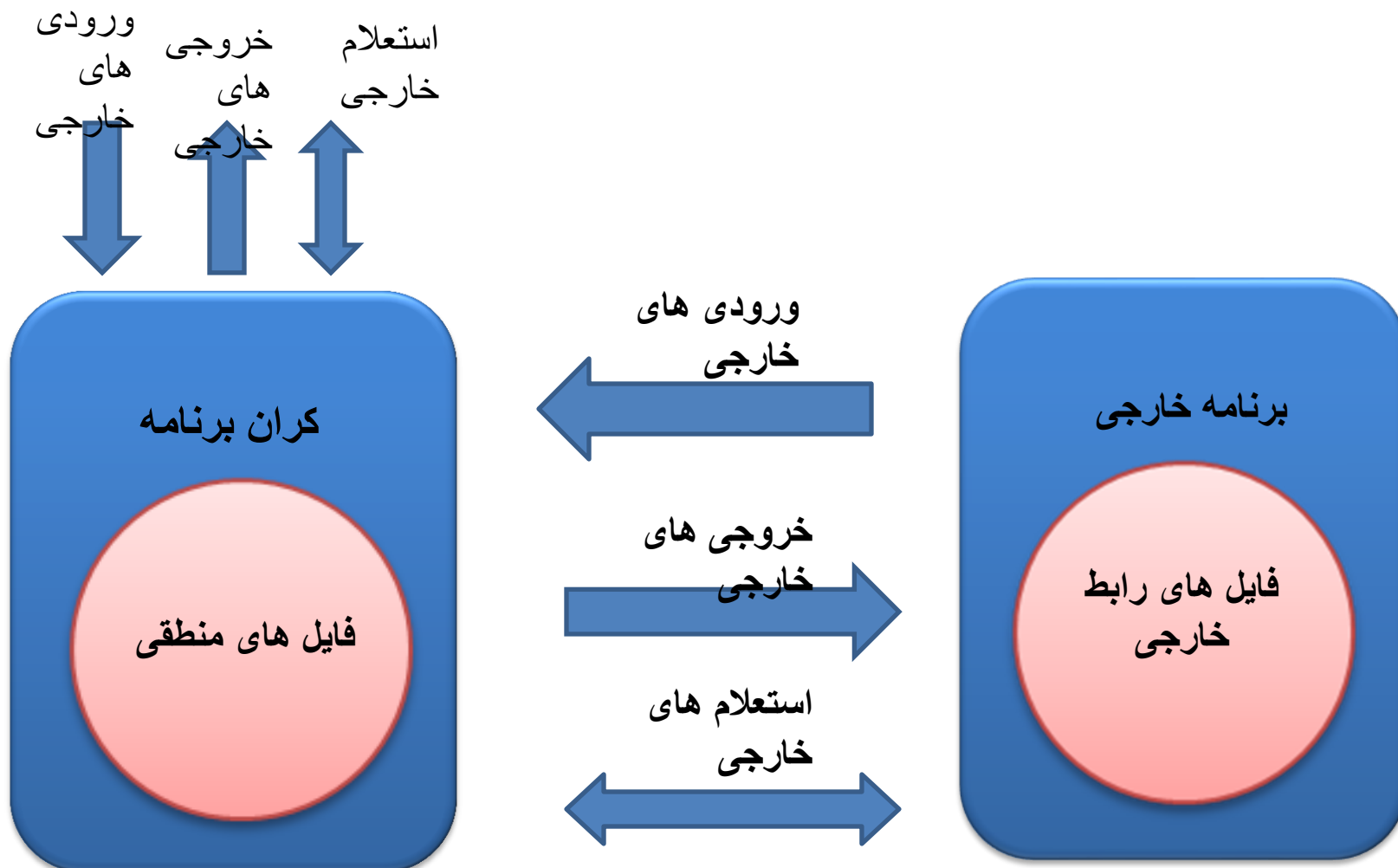
هردوی این سازمانهای غیرانتفاعی، برقوانین، راهنمایی ها، استانداردها و گواهی ها برای تحلیل نقطه تابعی، نظارت می کنند.

کلید اصلی برای شمارش نقاط تابعی، داشتن درک خوب  
از نیازمندیهای کاربر است.



- فایل منطقی داخلی (ILF)
- فایل رابط خارجی (EIF)
- ورودی خارجی (EI)
- خروجی خارجی (EO)
- پرس و جوی خارجی (EQ)





**فایل منطقی داخلی (ILF):** یک فایل منطقی است که داده ها را در محدوده برنامه ذخیره می کند. مثال هر موجودیت در یک دیاگرام موجودیت\_رابطه ای (ERD)، یک فایل منطقی داخلی محسوب می شود.

پیچیدگی این فایل می تواند به صورت کم، متوسط، زیاد براساس تعداد اجزای داده و زیر تیم های اجزای داده ای نگهداری شده به وسیله فایل دسته بندی شود.

یک زیر تیم ← مشتریان جدید برای موجودیتی به نام مشتری

اجزای داده ← شماره مشتری، نام، ادرس، تلفن و...

## ❑ فایل رابط خارجی (EIF)

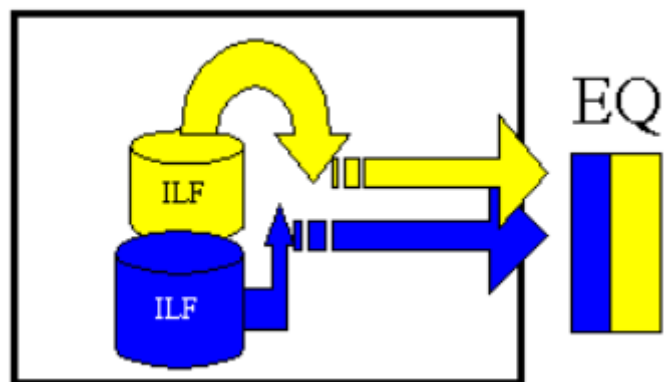
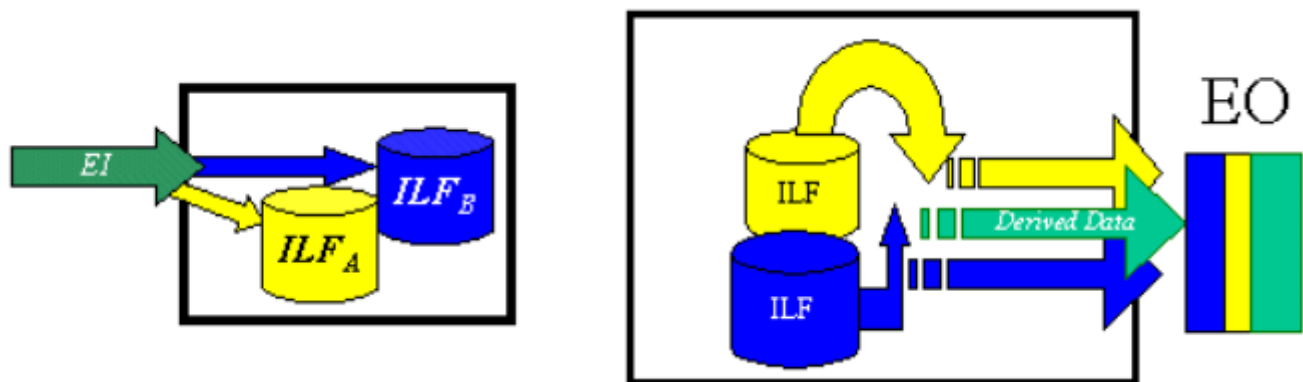
یک فایل رابط خارجی شبیه یک فایل منطقی داخلی می باشد، هرچند این فایلی است که به وسیله یک سیستم برنامه کاربردی دیگر، نگهداری می شود. پیچیدگی یک فایل رابط خارجی، به وسیله همان معیار استفاده شده برای فایل منطقی داخلی، تعیین می شود.

## ❑ ورودی خروجی (EI)

اشاره به فرایندها یا داده های تبدیلی دارد که خارج از برنامه بوجود می آیند و مرز برنامه را از بیرون به درون، طی می کنند. داده ها عموماً در یک یا چند فایل که برای برنامه، داخلی هستند (فایل های داخلی منطقی) اضافه حذف و یا بهنگام می شوند.

## ❑ خروجی خارجی (EO)

یک فرایند و یا مبادله می باشد که به داده اجازه می دهد از برنامه خارج شود. شامل گزارشات، پیغام های تایید، مجموع های مشتق شده و...

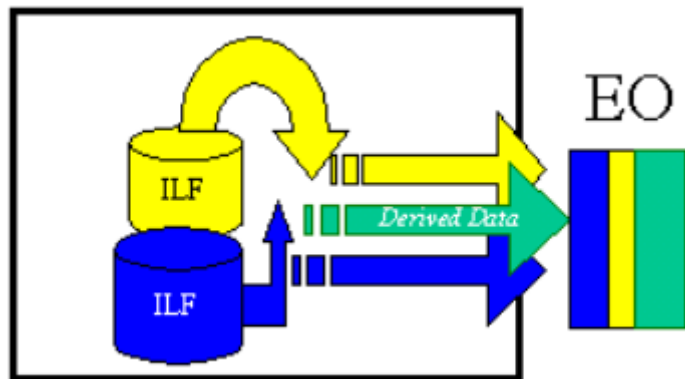


- فایل منطقی داخلی (ILF)
- فایل رابط خارجی (EIF)
- ورودی خارجی (EI)
- خروجی خارجی (EO)
- پرس و جوی خارجی (EQ)

مثال: ورودی خارجی EI

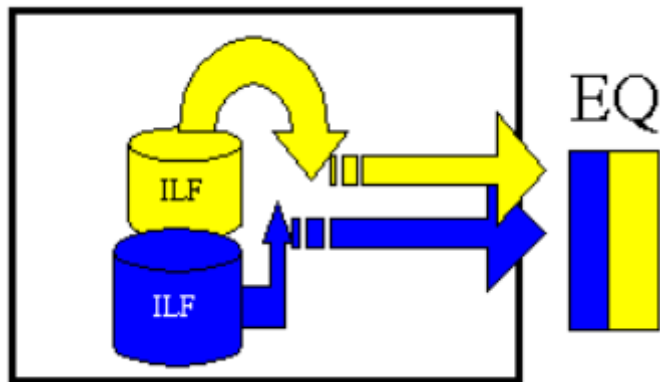
The image shows a 'New Vendor' dialog box with a teal title bar. The 'Vendor' field at the top is labeled with a blue '1'. Below it is the 'Address Info' tab. The form contains several input fields, each with a blue number indicating a data entry point: 'Company Name' (2), 'Mr./Ms./...' (3), 'First Name' (4), 'M.I.' (5), 'Last Name' (6), 'Address' (7), 'Contact' (8), 'Phone' (9), 'FAX' (10), 'Alt. Ph.' (11), 'Alt. Contact' (12), 'E-mail' (13), and 'Print on Check as' (14). On the right side of the dialog, there are two buttons: '15 OK' and 'Cancel'. The entire dialog is set against a yellow background.

Field Label	Field Number
Vendor	1
Company Name	2
Mr./Ms./...	3
First Name	4
M.I.	5
Last Name	6
Address	7
Contact	8
Phone	9
FAX	10
Alt. Ph.	11
Alt. Contact	12
E-mail	13
Print on Check as	14
OK	15



مثال: خروجی خارجی EO

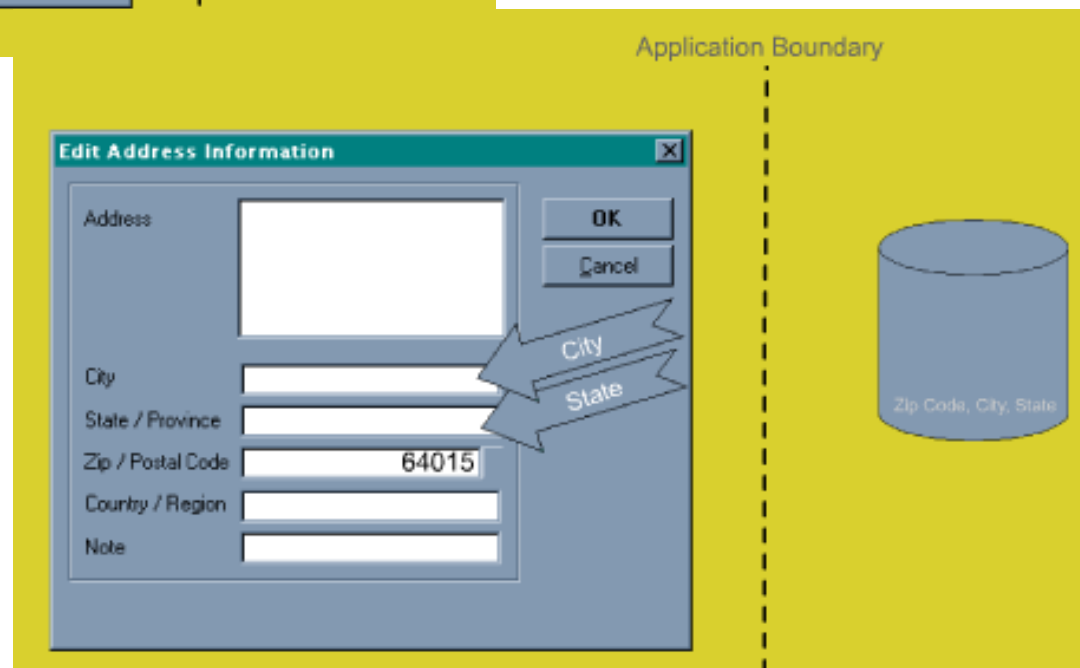
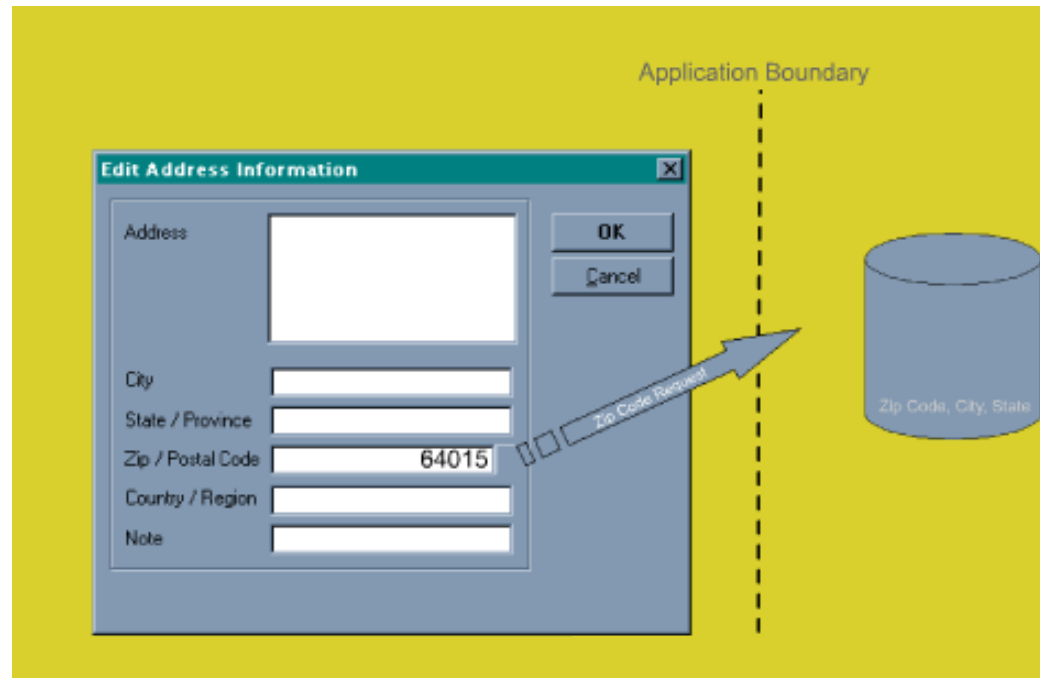
Reminders		
Due Date	Description	Amount
	<b>To Do Notes</b>	
	Money to Deposit	2,124.00
	Bills to Pay	-29,310.00
	Overdue Invoices	3,014.00
	Checks to Print	-9,675.00
	Paychecks to Print	-5,740.88
	Invoices/Credit Memos to Print	4,757.81
	Inventory to Reorder	



مثال: خروجی خارجی EQ

Employee List		
Soc. Sec. #		Name
333-44-5555		Dan T. Miller
444-55-6666		Gregg O. Schneider
569-87-1234		Elizabeth N. Mason
Employee Activities Reports Show All		

# مثال: خروجی خارجی EIF





## □ پرس وجوی خارجی (EQ)

یک فرایند یا مبادله که شامل ترکیبی از خروجی ها و ورودی ها برای بازیابی داده از فایل های داخلی یا فایل ها خارج برنامه می باشد.

○ هیچ داده ذخیره شده ای در فایل را تغییر نمی دهد یا بهنگام نمی کنند، بلکه این اطلاعات را فقط می خوانند.

○ پرس وجوهایی با منطق فرایندی یا قالب ورودی یا خروجی متفاوت، یک پرس وجوی خارجی واحد محسوب می شوند.

هنگامی که تمامی فایل های منطقی داخلی، فایل های رابط خارجی، ورودی های خارجی، خروجی های خارجی و پرس وجوهای خارجی شمرده شدند و پیچیدگی های مربوط به آن ها درجه بندی شد، یک نقطه تابع **ناسازگار (UAF)** تعیین می شود.

**ELF: ۲ متوسط**

**EO: ۴ کم، ۲ متوسط، ۱ پیچیده**

**ILF: ۳ کم، ۲ متوسط، ۱ پیچیده**

**EI: ۳ کم، ۵ متوسط، ۴ پیچیده**

**EQ: ۲ کم، ۵ متوسط، ۳ پیچیده**

جدول محاسبه UAF

پیچیدگی

جمع کل	بالا	متوسط	پایین	
56	$1 \times 15 = 15$	$2 \times 10 = 20$	$3 \times 7 = 21$	فایل های منطقی داخلی (ILF)
14	$\_ \times 10 = \_$	$2 \times 7 = 14$	$\_ \times 5 = \_$	رابط خارجی (EIF)
53	$4 \times 6 = 24$	$5 \times 4 = 20$	$3 \times 3 = 9$	ورودی خارجی (EI)
33	$1 \times 7 = 7$	$2 \times 5 = 10$	$4 \times 4 = 16$	خروجی خارجی (EO)
44	$3 \times 6 = 18$	$4 \times 5 = 20$	$2 \times 3 = 6$	استعلام خارجی (EQ)
200				مجموع نقاط تابعی تنظیم نشده (UAF)

## User requirements:

- user wants to add, change, delete Customer data, wants to inquire on Customer, and also requires four different reports on Customer with calculated data
- user wants to add, change, delete Product data, wants to inquire on Product, and also requires a report on Product with calculated data
- user wants to inquire on Supplier using supplier number, and also requires a report on Supplier with totaling results

مثال:

Data or transactional function	Function type	Complexity (by default)	Function points (unadjusted)
Customer	ILF	Low	7
Product	ILF	Low	7
Supplier	EIF	Low	5
Add Customer	EI	Average	4
Change Customer	EI	Average	4
Delete Customer	EI	Average	4
Inquire on Customer	EQ	Average	4
Report 1 on Customer	EO	Average	5
Report 2 on Customer	EO	Average	5
Report 3 on Customer	EO	Average	5
Report 4 on Customer	EO	Average	5
Add Product	EI	Average	4
Change Product	EI	Average	4
Delete Product	EI	Average	4
Inquire on Product	EQ	Average	4
Report on Product	EO	Average	5
Inquire on Supplier	EQ	Average	4
Report on Supplier	EO	Average	5
<b>Estimated functional size</b>			<b>85 fp</b>

## قدم بعدی در تحلیل نقطه تابعی

محاسبه یک فاکتور تطبیق مقدار (VAF) می باشد.  
فاکتور تطبیق مقدار، مبتنی بر درجه نفوذ (DI) است که معمولاً به آن  
تطبیق پیچیدگی فرایندی (PCA) گفته می شود و از ۱۴ مشخصه  
عمومی سیستم (GSC) مشتق می شود.

**تعیین درجه نفوذ کل** بر اساس مقیاس زیر از ۰ تا ۵ درجه بندی  
می شوند:

۰ = وجود ندارد یا نفوذ ندارد.

۱ = نفوذ خیلی کم

۲ = نفوذ کم

۳ = نفوذ متوسط

۴ = نفوذ زیاد

۵ = نفوذ خیلی زیاد

جدول Total Adjusted Function Point و GSC

مشخصه‌های عمومی سیستم

میزان تأثیر

3

ارتباطات داده

2

پردازش داده توزیع شده

4

کارایی

3

پیکربندی پر استفاده

3

نرخ تراکنش

4

ورودی داده Online

4

کارایی کاربر نهایی

3

بهنگام‌رسانی Online

3

پردازش پیچیده

2

قابلیت استفاده مجدد

3

سهولت نصب

3

سهولت عملکرد

1

چندین سایت

2

تسهیل تغییر

40

مجموع درجات تأثیر (TDI)

$$VAF = (40 * .01) + .65 = 1.05$$

$$FP = 200 * 1.05 = 210$$

$$VAF = (TDI * 0.01) + .65$$

عامل تنظیم ارزش

$$FP = UAF * VAF$$

مجموع نقاط تابعی تنظیم شده

- باتوجه به جدول بالا هنگامی که یک مجموع نقاط تابعی تطبیقی، محاسبه می شود، تعدادنقاط تابعی می تواند تبدیل به تخمین های توسعه شود.
- رویکرد دوم برتبدیل تعدادنقاط تابعی به تعدادمعدل خطوط کد، تمرکز می کند.
- تعدادخطوط کد بستگی به زبان برنامه نویسی دارد.
- یک برنامه یابخشی از برنامه که مجموع ۲۱۰نقطه تطبیقی ناسازگار دارد، اگر در زبان ماشین، نوشته شودمثلا ۱۳۴۴۴۰خط کد دارد، اما اگر با **visual basic5** شودنیاز به ۶۰۹۰ خط کد دارد



## کوکومو

- مخفف مدل هزینه ای سازنده می باشد که برای اولین بار در سال ۱۹۸۱ به وسیله بری بوئم در کتابش به نام مهندسی نرم افزار معرفی شد.
- براساس تخمین های خطوط کد از آن برای تخمین هزینه ها تلاش وزمان بندی استفاده می شود.
- کوکومو مثالی از یک مدل پارامتریک است زیرا ازمتغیرهای وابسته مانندهزینه یا مدت زمان براساس یک یا چند متغیرمستقل که اندیس های کمی از عملکردیا مشخصه های فیزیکی سیستم می باشند،استفاده می کنند.

**تخمین زدن با کوکومو، با تعیین نوع پروژه ای است که قرار است تخمین زده شود آغاز می شود. انواع پروژه ها عبارتند از:**

**سازمانی:** معمولی هستند انتظار می رود فناوری، فرایندها و افراد همگی با آرامی کار کنند. پروژه آسان محسوب می شود.

**تعبيه شده:** پروژه چالش برانگیزی است مثال سیستمی برای پشتیبانی از یک فرایند کسب و کار. افراد کم تجربه تر و فناوری تکامل نیافته تر

**نیمه جدا شده:** ممکن است آسان و سر راست نباشد اما سازمان این اطمینان را دارد که فرایندها، افراد و فناوری درون آن برای رویارویی با چالش این پروژه ها، کفایت می کند.

در کوکومو یک فرد - ماه ۱۵۲ ساعت تعریف می شود

میزان تلاش بر حسب فرد\_ماه با استفاده از معادله مناسب به دست می آید:

$$✓ \text{سازمانی: فرد_ماه} = 2.4 \times KDSI^{1.05}$$

$$✓ \text{نیمه جدا شده: فرد_ماه} = 3.0 \times KDSI^{1.12}$$

$$✓ \text{تعبیه شده: فرد_ماه} = 3.6 \times KDSI^{1.20}$$

✓ KDSI: هزاران دستورالعمل مبدا تحویل داده شده یعنی خطوط کد

همان‌طور که فردریک بروکس (۱۹۹۵)، اشاره می‌کند، افراد و ماه‌ها همواره قابل تعویض نیستند. تعداد افراد بیشتر، ممکن است ارتباطات را پیچیده‌تر کند و کارها را کند نماید. بنابراین، مدت زمان به طول انجامیدن، با یکی از فرمول‌های زیر تعیین می‌شود:

$$\checkmark \text{ سازمانی: مدت زمان} = 2.5 \times \text{Effort}^{0.38}$$

$$\checkmark \text{ نیمه جدا شده: مدت زمان} = 2.5 \times \text{Effort}^{0.35}$$

$$\checkmark \text{ تعبیه شده: مدت زمان} = 2.5 \times \text{Effort}^{0.32}$$

از آنجایی که پروژه نیمه جدا شده ما به 42.21 نفر-ماه نیاز دارد، مدت زمان ایجاد آن عبارت است از:

$$\begin{aligned} \text{مدت زمان} &= 2.5 \times \text{Effort}^{0.35} \\ &= 2.5 \times (42.21)^{0.35} \\ &= 9.26 \text{ ماه} \end{aligned}$$

سپس، می‌توانیم تعیین کنیم که چند نفر باید به تلاش انجام کار، نسبت داده شوند:

$$\begin{aligned} \text{افراد مورد نیاز} &= \text{Effort} + \text{Duration} \\ &= 42.21 + 9.26 \\ &= 4.55 \end{aligned}$$

بنابراین به 4.55 نفر برای کار روی پروژه نیاز خواهیم داشت. بسیار خوب، پس تا حدی مشکل است که 0.55 از یک نفر داشته باشیم، بنابراین به ۴ یا ۵ نفر نیاز خواهیم داشت. حتی می‌توان استدلال کرد که ۴ فرد تمام وقت و ۱ فرد پاره وقت، برای این پروژه لازم است.

علاوه، کوکوموی پیشرفته، شامل کلیه خصوصیات کوکوموی متوسط می باشد اما با یک ارزیابی از تأثیر محرک هزینه‌ای در چهار فاز توسعه: طراحی محصول، طراحی تفصیلی، کدنویسی / آزمودن و یکپارچه‌سازی / آزمودن.

امروزه کوکوموی ۲ در دسترس است و برای انواع پروژه‌هایی که با استفاده از زبان‌های برنامه‌نویسی نسل چهارم و یا ابزارهای دیگری مانند Delphi، Visual Basic، یا Power Builder انجام می‌شوند، مناسب‌تر است. هرچند، برای پروژه‌های سنتی‌تر که از زبان‌های برنامه‌نویسی نسل سوم استفاده می‌کنند، مدل کوکوموی اولیه، هنوز می‌تواند تخمین‌های خوبی فراهم کند که معمولاً به آن کوکوموی ۸۱ گفته می‌شود.

یک مدل تخمین زدن دیگر که باید با آن آشنا شوید، SLIM، می‌باشد که در اواخر دهه ۱۹۷۰ به وسیله لری پوتنام<sup>۱</sup> از مدیریت نرم‌افزاری کمیتی<sup>۲</sup> ایجاد شد (پوتنام ۱۹۷۸؛ پوتنام و فیتزیمونز<sup>۳</sup> ۱۹۷۹). مانند کوکومو، SLIM نیز از خطوط کد برای تخمین اندازه پروژه و یک سری ۲۲ تایی از سؤالات برای اصولی کردن مدل استفاده می‌کند.

## روش‌های مکاشفه‌ای<sup>۴</sup>

روش‌های مکاشفه‌ای، قواعد کلی هستند. رویکردهای مکاشفه‌ای، بر این حقیقت استوارند که همان فعالیت‌های پایه برای یک پروژه معمولی تولید نرم‌افزار لازم است و این فعالیت‌ها به یک درصد قابل پیش‌بینی از تلاش کلی نیاز خواهند داشت (روتزیم و بیسلی ۱۹۹۸). به عنوان مثال، هنگام تخمین زمان‌بندی برای یک فعالیت تولید نرم‌افزار، می‌توان بر اساس پروژه‌های قبلی، درصدی از تلاش کلی را به صورت زیر نسبت داد:

✓ ۳۰ درصد برنامه‌ریزی

✓ ۲۰ درصد کدنویسی

✓ ۲۵ درصد آزمودن اجزای سیستم

✓ ۲۵ درصد آزمودن سیستم



تی.کیپرز.جونز. در کتاب خود به نام تخمین هزینه‌های نرم‌افزار، تعدادی از روش‌های مکاشفه‌ای یا قوانین سرانگشتی برای تخمین پروژه‌های نرم‌افزاری بر اساس نقاط تابعی را فراهم کرده است. برخی از این قوانین عبارتند از:

✓ نقاط تابعی به توان 1.15، تعداد تقریبی صفحات برای اسناد کاغذی مرتبط با پروژه‌های نرم‌افزاری را پیش‌بینی می‌کنند.

✓ نیازهای تغییر یافته کاربران، با یک نرخ متوسط ۲ درصد در ماه از فاز طراحی تا کدنویسی، افزایش می‌یابد.

✓ نقاط تابعی به توان 1.2، تعداد تقریبی موارد آزمون ایجاد شده را پیش‌بینی می‌کنند.

✓ نقاط تابعی به توان 1.25، ظرفیت تقریبی خطا را برای پروژه‌های نرم‌افزاری جدید، پیش‌بینی می‌کنند.

✓ هر قدم آزمودن نرم‌افزار، ۳۰٪ خطاهای موجود را پیدا و برطرف می‌کند.

✓ هر بازرسی رسمی طراحی، ۶۵٪ خطاهای موجود را پیدا و برطرف می‌کند.

✓ هر بازرسی رسمی کد، ۶۰٪ خطاهای موجود را پیدا و برطرف می‌کند.

✓ برنامه‌نویسان پشتیبان، ۸ خطای کارکنان در ماه را می‌توانند برطرف کنند.

✓ نقاط تابعی به توان 0.4، زمان‌بندی توسعه تقریبی در ماه‌های تقویم را پیش‌بینی می‌کنند.

✓ نقاط تابعی تقسیم بر 150، تعداد تقریبی کارکنان لازم برای زمان‌بندی را پیش‌بینی می‌کنند.

✓ نقاط تابعی تقسیم بر 750، تعداد تقریبی کارکنان پشتیبانی مورد نیاز برای بهنگام نگه داشتن زمان‌بندی را پیش‌بینی می‌کنند.

✓ زمان‌بندی‌های توسعه نرم‌افزار را در تعداد کارکنان ضرب کنید تا تعداد تقریبی ماه‌های کاری تلاش را پیش‌بینی کنید.



چونز یک اظهار نظر مهم می‌نماید: قوانین سرانگشتی، آسان هستند، اما دقیق نیستند. همان‌طور که گارموس<sup>۱</sup> و هرون<sup>۲</sup> اشاره می‌کنند (گارموس و هرون ۱۹۹۶):

تفمین دقیق، تابعی است از به کارگیری یک فرآیند و تشفیص اینکه در ایبار فط مبنای تجربه که امکان دقت بالاتر برای آن فرآیند را فراهم می‌کند، باید تلاش شود. تفمین نیاز به گوی بلورین ندارد؛ تنها نیازمند تعهد است.

## ابزارهای تخمین زنی خودکار

تعدادی ابزار خودکار می تواند برای تخمین هزینه، زمان بندی و منابع مورد استفاده قرار گیرد. این ابزارها شامل صفحات گسترده، ابزارهای مدیریت پروژه، سیستم های مدیریت پایگاه داده، تخمین هزینه نرم افزاری و ابزارهای متدلوژی و یا فرآیندی می باشند. بسیاری از این ابزارها، نه تنها به تخمین زدن کمک می کنند، بلکه همچنین، به سازمان اجازه می دهند که یک پایگاه داده و یا یک انبار از پروژه های قبلی ایجاد کند. در واقع، نشان داده شده است که هنگامی که داده های گذشته دقیق باشند، تخمین ها معمولاً دقتی بین ۵ تا ۱۰ درصد دارند. به علاوه، ابزارهای تخمین زنی خودکار وقتی دقیق نیستند، عموماً محافظه کارانه تر هستند، در مقابل روش های دستی که عموماً خوش بینانه می باشند (جونز ۱۹۹۸).

با افزایش پیچیدگی پروژه‌های تولید نرم‌افزار، به همان اندازه، بازار ابزارهای تخمین نرم‌افزار نیز افزایش می‌یابد. برخی از ابزارهای خودکار در دسترس عبارتند از: کوکومو ۲، SLIM، CHECKPOINT، Knowledg Plan و Cost\*Xpert. تحقیقات نشان می‌دهد که پروژه‌هایی که از یک ابزار تخمین رسمی استفاده می‌کنند، شانس بهتری برای تحویل یک سیستم به موقع و در محدوده بودجه دارند.

## بهترین راه برای تخمین پروژه‌های فناوری اطلاعات چیست؟

متأسفانه، هیچ روش و یا ابزار واحدی برای تخمین دقیق پروژه‌های فناوری اطلاعات، وجود ندارد. شاید ایده خوبی باشد که از بیش از یک روش برای تخمین زدن استفاده کنیم. هرچند، به احتمال خیلی زیاد، دو تخمین متفاوت خواهید داشت.

اگر تخمین‌ها از روش‌های تخمین زدن متفاوت، نسبتاً نزدیک به هم باشند، آنگاه می‌توان آن‌ها را با درجه نسبتاً بالایی از اطمینان، میانگین گرفت. اگر تخمین‌ها خیلی متفاوت باشند، آنگاه باید نسبت به یکی یا هر دوی تخمین‌ها شک کرده و داده‌های جمع‌آوری شده را بازبینی کنید (روتزیم و بیسلی ۱۹۹۸).



تخمین‌های اولیه شما، احتمالاً باید بر اساس تجربیات گذشته یا داده‌ها از پروژه‌های قبلی، به بالا یا پایین تطبیق داده شوند. هرچند که در خیلی از مواقع، تخمین‌های اولیه به وسیله مدیریت بالا و یا کارآور به بحث گذاشته می‌شوند. به عنوان مثال، ممکن است به تخمینی برسید که پروژه ۱۲ ماه طول می‌کشد و ۱.۲ میلیون دلار هزینه بر می‌دارد. مدیریت بالا ممکن است مقابله کند و دستور دهد که پروژه در ۸ ماه انجام شود و بیش از ۷۵۰,۰۰۰ دلار هزینه بر ندارد، مگر اینکه بتوانید تخمین‌های خود را اثبات کنید. این مقابله ممکن است به دلیل یک نیاز واقعی برای کسب و کار باشد (یعنی واقعاً بعد از ۸ ماه به آن نیاز خواهند داشت و نمی‌توانند بیش از ۷۵۰,۰۰۰ دلار نیز هزینه کنند) یا اعتقاد آن‌ها به این است که شما زمان‌بندی و بودجه را دچار تورم کرده‌اید و تخمین‌های شما را می‌توان کمتر کرد. در نتیجه، ممکن است در نهایت روی یک پروژه راه‌پیمایی مرگ کار کنید.

اساساً به اینجا می‌رسد که آیا پروژه می‌تواند زودتر تحویل داده شود یا خیر. وظیفه یک مدیر پروژه است که نه تنها به یک تخمین برسد، بلکه بتواند همچنین از تخمین‌ها پشتیبانی کند. در غیر این صورت اجرای زمان‌بندی و بودجه پروژه می‌تواند بسیار مشکل باشد. کار کردن در ساعات طولانی و تحت فشار شدید، مطمئناً تأثیری منفی بر تیم پروژه خواهد داشت. تیم مدیر پروژه همیشه باید در درجه اول اهمیت باشد و محافظت آن‌ها با داشتن یک مهلت واقع‌گرایانه و منابع کافی بر حسب زمان‌بندی و بودجه پروژه، قدم اول است.