



# IFPUG功能点估算方法介绍及实例分析

作者：陈 云



# 主要内容

---

- 一、关于**IFPUG**组织
- 二、功能点分析方法概述
- 三、**IFPUG**方法的五大组件
- 四、**IFPUG**方法的估算流程
- 五、估算实例



# 一、关于IFPUG组织

- **IFPUG (International Function Point Users' Group)**
  - 国际功能点用户组织是一个致力于功能点分析研究的非营利性组织。
  - 1986年IFPUG组织成立，后续的功能点指南都是IFPUG组织发布的Function Point Counting Practices Manual系列版本。
  - 主页: <http://www.ifpug.org/>





# 主要内容

---

- 一、关于**IFPUG**组织
- 二、功能点分析方法概述
- 三、**IFPUG**方法的五大组件
- 四、**IFPUG**方法的估算流程
- 五、估算实例



## 二、功能点分析方法概述-1

- 功能点分析技术（**FPA, Function Point Analysis**）
  - 从用户的角度来估算软件开发的标准方法
  - 主要基于用户的逻辑功能需求来量化软件的功能点个数，而不拘泥于物理形式。
- 什么是功能点？
  - 功能点是度量软件规模的一个标准度量单位。
  - 一个软件的大小可以通过交付给用户的功能点数来度量。
- 与**LOC**代码行估算的区别
  - 功能点估算法常用于开发周期早期，而这个时候使用**LOC**代码行估算法，则误差会比较大。
  - 功能点估算法独立于具体的开发技术。**LOC**代码行估算法则与软件开发技术密切相关。
  - 功能点估算法是以用户为角度进行估算，**LOC**代码行估算法则是以技术为角度进行估算。



## 二、功能点分析方法概述-2

---

### ■ 适合范围:

- 数据操作密集系统的规模估算，如信息管理系统

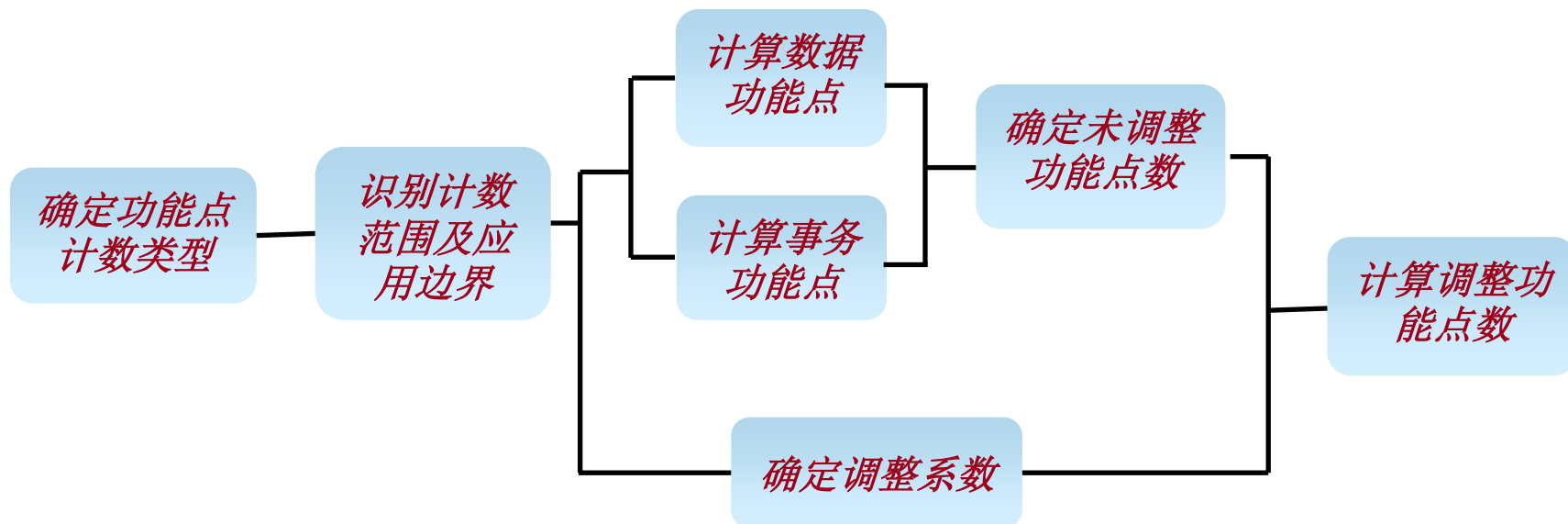
### ■ 不适合范围:

- 控制流程复杂、数据实时性高的系统，如实时系统、科学计算等



## 二、功能点分析方法概述-3

- 估算步骤：
  - 1、确定功能点计数类型
  - 2、识别计数范围及应用边界
  - 3、确定未调整功能点数
  - 4、确定调整系数值
  - 5、应用公式计算FP值





# 主要内容

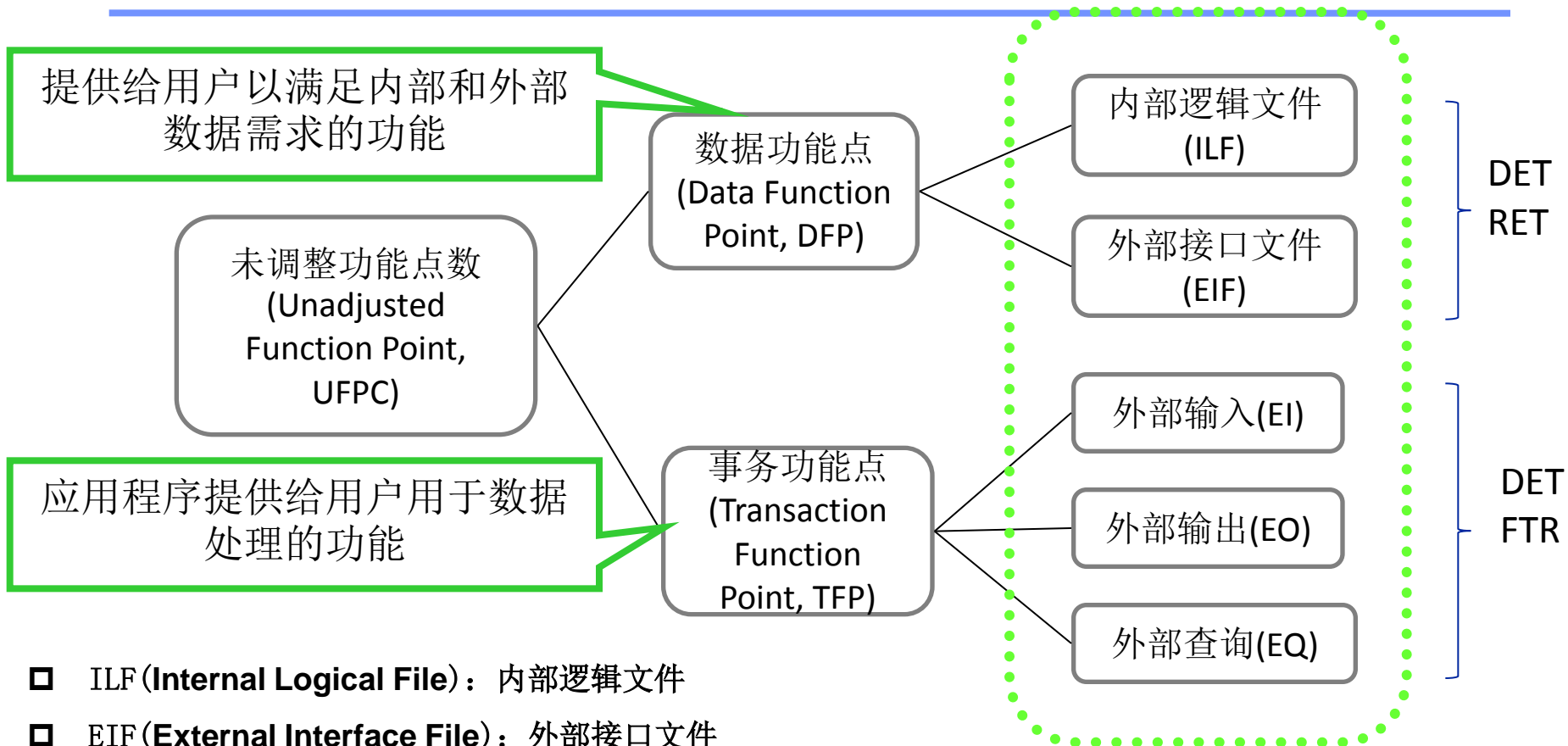
---

- 一、关于**IFPUG**组织
- 二、功能点分析方法概述
- 三、**IFPUG**方法的五大组件
- 四、**IFPUG**方法的估算流程
- 五、估算实例





### 三、IFPUG方法五大组件



- ❑ ILF (Internal Logical File): 内部逻辑文件
- ❑ EIF (External Interface File): 外部接口文件

- ❑ 数据元素类型 (Data Element Type, DET)
- ❑ 记录元素类型 (Record Element Type, RET)
- ❑ 引用文件类型 (File Referenced Type, FRT)

- ❑ EI (External Input): 外部输入
- ❑ EO (External Output): 外部输出
- ❑ EQ (External Inquiry): 外部查询

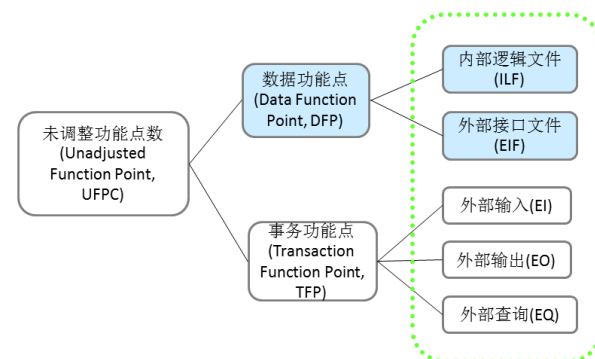


## 三、 IFPUG方法五大组件

---

- **3.1 数据功能点的概念**
- **3.2 DET/RET的识别规则**
- **3.3 数据功能点的计数实例**
- **3.4 事务功能点的概念**
- **3.5 事务功能点的识别实例**

## 3.1 数据功能点的概念

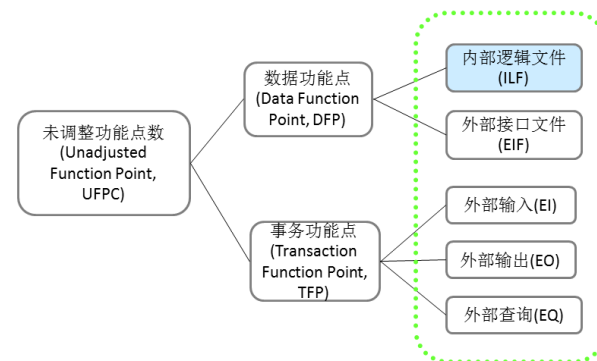




# 什么是ILF?

## ILF（内部逻辑文件）

- 定义：在系统边界内，用户可识别的一组逻辑上关联且被应用维护的数据或者控制信息。
  - 维护：通过基本流程修改数据的能力，包括增、删、改等。
  - 基本流程：对用户有意义的最小活动单元，并且它是自包含的，使系统处在一致的状态。
  - 控制信息：影响基本流程的数据，它说明处理什么数据、什么时候以及怎样处理数据。
- 主要目的：保存通过一个或多个基本流程处理的数据。
- 计算步骤：
  - ① 识别出ILF
  - ② 计算各个ILF的复杂度和功能点个数



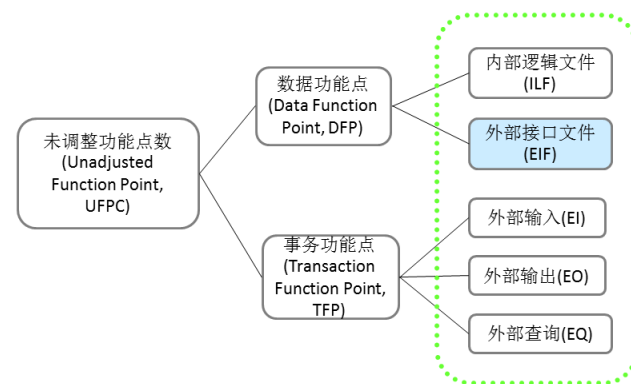


# 什么是EIF?

- **EIF（外部接口文件）**

- 定义：一组在应用边界内被查询，但在其他应用中被维护的、用户可识别的、逻辑相关数据或者控制信息。
- 主要目的：使数据在应用边界内通过一个或几个基本处理过程得以查询。

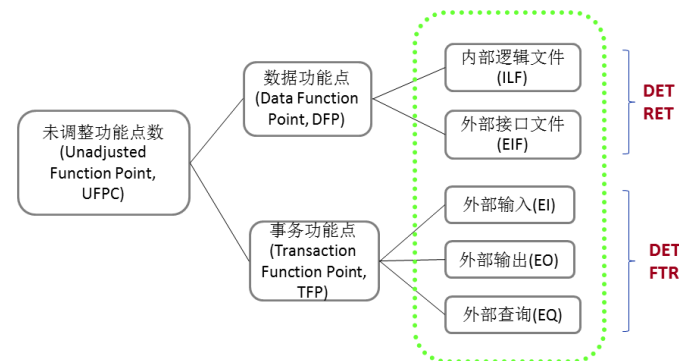
一个应用中的一个**EIF**必然是其他应用中的**ILF**。





# 什么是DET、RET、FTR?

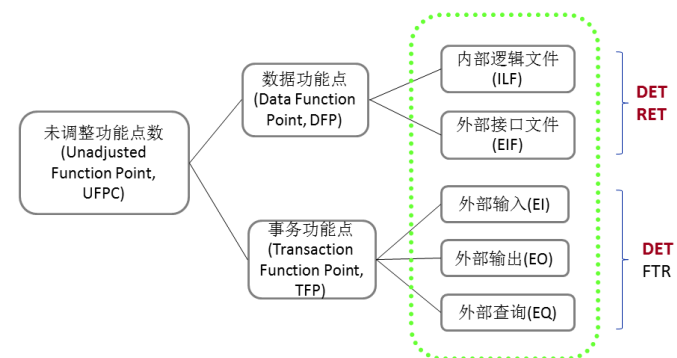
- **DET**（数据元素类型）
  - 用户可识别的无递归、不重复的信息单元。
- **RET**（记录元素类型）
  - 在ILF或EIF中，用户可识别的数据域的子集，可以通过检查数据中的各种逻辑分组来识别它们。
- **FTR**（引用文件类型）
  - 被事务功能读取或维护的内部逻辑文件，或者是被事务功能读取的外部接口文件。





# DET、RET的识别规则

## 3.2 DET、RET的识别规则





## 如何识别DET?

- **DET识别规则**

- **用户可识别，非重复的字段，通过基本流程处理ILF/EIF时获得。**

**eg:** 如果员工号在一个**ILF**或**EIF**中出现两次，一次是作为员工记录的主键，一个是作为家属信息的外键，员工号记为一个**DET**。

- **当两个应用程序维护或引用相同的ILF、EIF，但是各自使用不同的DET，那么仅计算它使用到的DET。**

**eg:** 应用程序**A**使用到的地址信息包括：**street address, city, state, zip code**。  
应用程序**B**可能把地址当做一个整体而未细化到个体，因此应用程序**A**计数有4个**DET**，而应用程序**B**计数有1个**DET**。

- **每个被用户用来和其他ILF或EIF建立关系的数据也是一个DET。**

**eg:** 在**HR**系统中，员工信息是一个**ILF**，员工职位名称也算作是员工信息的一部分，被算作一个**DET**，因为它可以把员工和职位联系起来，这类**DET**成为外键。





# 如何识别RET?

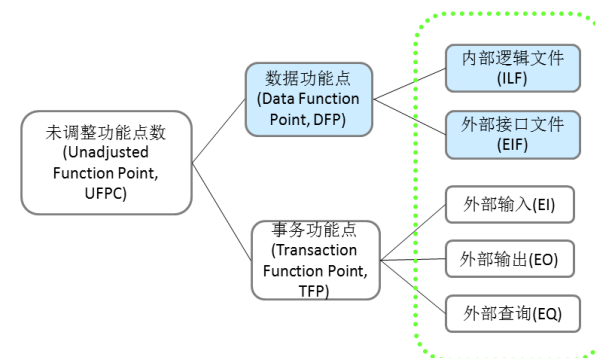
- **RET识别规则**

- RET是指一个EIF/ILF中用户可以识别的DET的集合。
- RET在ILF/EIF中分为两种类型
  - 可选的（**Optional**）和必选的（**Mandatory**）。
- 计算RET的规则为以下两点：
  - 在一个ILF/EIF中每一个可选或必选的集合都计为一个RET。
  - 如果一个ILF/EIF没有子集合，则ILF/EIF被计算为一个RET。
- RET识别例子：
  - 员工信息包括员工基本信息和家属信息。而员工又分为薪水工和小时工，不管是薪水工还是小时工都有家属信息，那么员工信息包含的RET有：薪水工基本信息，小时工基本信息（**Mandatory**）和家属信息（**optional**）3个。



# 数据功能点的计数实例

## 3.3 数据功能点的计数实例





## ILF计数例子—员工管理系统

- 用户需求
  - 能够输入、查询和导出职位信息。
  - 需要一起维护的信息包括：
    - 职位号 (**Job number**)
    - 职位名称(**Job name**)
    - 职位薪水等级 (**Job pay grade**)
    - 职位描述行数 (**Job description line number**)
    - 职位描述 (**Job description lines**)

Job:

职位号 (Job number)

职位名称(Job name)

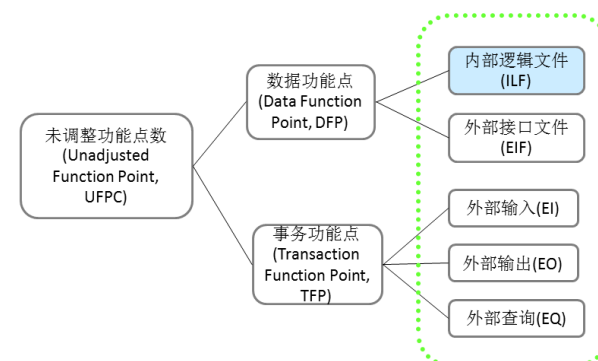
职位薪水等级 (Job pay grade)

Job description:

职位号 (Job number)

职位描述行数 (Job description line number)

职位描述 (Job description lines)



## ILF识别例子—员工管理系统

- 由此得到两组信息：**Job** , **Job description**

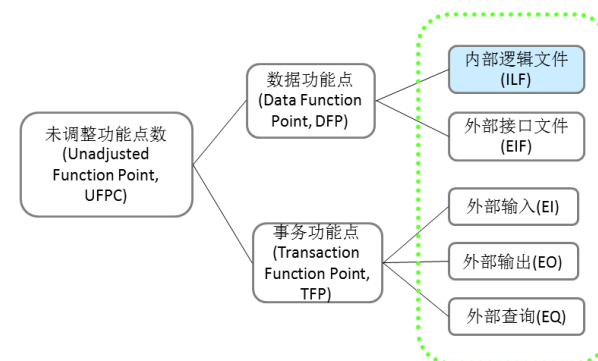
判断职位数据组是否为 ILF

ILF 识别规则	是否应用此规则?
这组数据或控制信息是逻辑上的并且用户可识别	没有, Job 应该包含 Job information, 否则不能满足用户添加一个职位信息的需求。
这组数据通过基本流程来处理	否

判断职位描述数据组是否为 ILF

ILF 识别规则	是否应用此规则?
这组数据或控制信息是逻辑上的并且用户可识别	没有, Job information 应该包含 Job, 否则不能满足用户添加一个职位信息的需求。
这组数据通过基本流程来处理	否

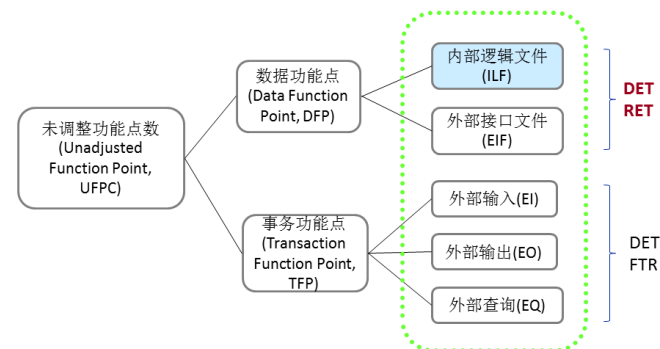
由此可见, 仅仅Job 或者Job description数据组都不是ILF, 这两者必须组合起来因为它们需要一起被维护。因此从用户角度来看, 只有一个ILF, 即职位信息。





# ILF计数例子—人力资源系统

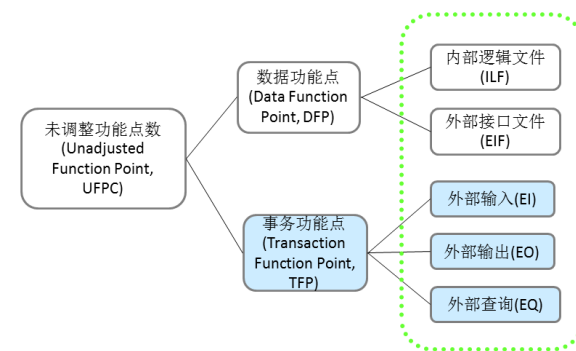
- 职位信息包含的**DET**:
  - 职位号 (Job number)
  - 职位名称(Job name)
  - 职位薪水等级 (Job pay grade)
  - 职位描述行数 (Job description line number)
  - 职位描述 (Job description lines)
- 职位信息包含的**RET**:
  - Job
  - Job description
- 职位信息总共包含: **2个RETs, 5个DETs**





# 事务功能点的概念

## 3.4 事务功能点的概念



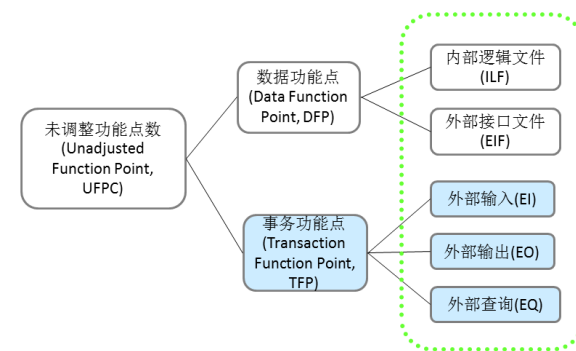
### 三、IFPUG方法五大组件—事务功能

- **EI（外部输入）**
    - 处理来自系统外部的数据或控制信息的基本流程。
    - 主要目的：维护一个或多个**ILF**、改变系统行为。
  - **EO（外部输出）**
    - 向应用边界之外发送数据或控制信息的基本流程。
    - 主要目的：通过处理逻辑把信息呈现给用户。处理逻辑至少包含一个数学公式或计算，或者创建派生数据。也能维护一个或多个**ILF**、改变系统行为。
  - **EQ（外部查询）**
    - 向应用边界之外发送数据或控制信息的基本流程。
    - 不维护**ILF**。
- 与EO的区别：不包括任何的数学公式或计算过程，不会生成任何的派生数据，仅向用户呈现信息。



# 事务功能点的识别实例

## 3.5 事务功能点的识别实例







# 事务功能点的识别实例

- 用户需求:
  - 实时添加职位信息
  - 能够给出错误提示
  - 保存添加的职位信息
  - 查看新添加的的职位信息
  - 计算已添加的职位总数

Job:

职位号 (Job number)

职位名称(Job name)

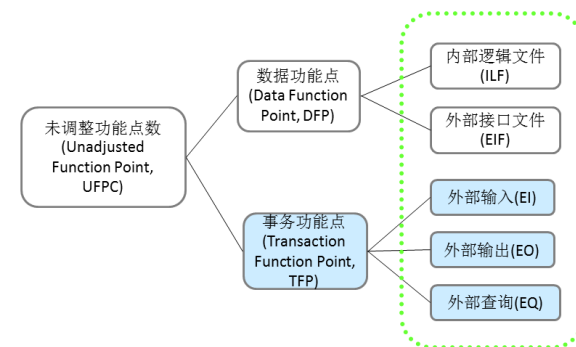
职位薪水等级 (Job pay grade)

Job description:

职位号 (Job number)

职位描述行数 (Job description line number)

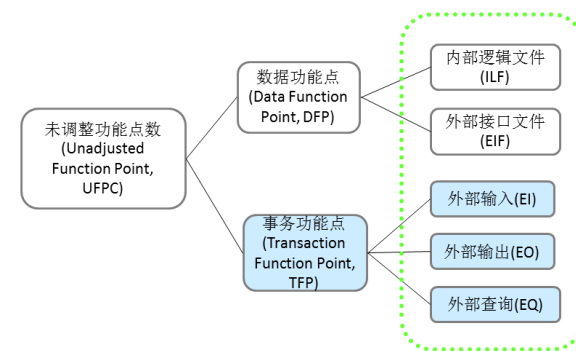
职位描述 (Job description lines)





## 识别EI/EO/EQ

- 分析用户需求可知:
- **EI**: 添加职位信息
- **EO**: 计算已添加职位总数
- **EQ**: 查看新添加的职位信息





# 识别EI

- EI（添加职位信息）包含的DET和FTR:

FTR 计数规则	是否应用此规则?
它所维护的每一个 ILF 算做一个 FTR。	维护了职位信息 ILF
通过 EI 读取的每个 ILF/EIF 计算为一个 FTR。	读取了职位信息 ILF
既被 EI 维护又被读取的 ILF 仅计算一次。	职位信息 ILF 既被维护又被读取，计算一次
FTR 个数总计：1 个	

+

DET 计数规则	是否应用此规则?
在 EI 的过程中，以用户角度识别的、通过应用系统边界输入系统内部的非重复字段，应算作一个 DET。	Job number, Job name Job pay grade, Job description line number(Repeated) Job description line(Repeated)
在 EI 的过程中，只要没有通过系统边界输入，即使它存在于系统内的一个 ILF 中，也不能算为一个 DET。	此处不使用
向系统边界外发送错误提示消息也算是一个 DET。	错误信息
虽然有多种方法调用相同的逻辑处理，但能够说明具体采用的动作。	新增操作
DET 个数总计：7 个	

调整功能点数  
Unadjusted  
Function Point,  
UFP

数据功能点  
(Data Function  
Point, DFP)

内部逻辑文件  
(ILF)

外部接口文件  
(EIF)

DET  
RET

事务功能点  
(Transaction  
Function Point,  
TFP)

外部输入(EI)

外部输出(EO)

外部查询(EQ)

DET  
FTR

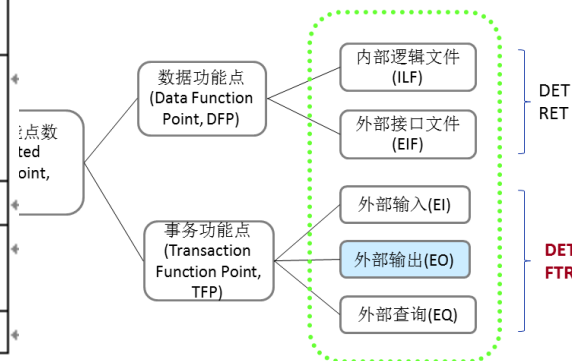


# 识别EO

- EO（计算已添加职位总数）包含的DET和FTR:

FTR 计数规则	是否应用此规则?
它所维护的每一个 ILF 算做一个 FTR。	维护了职位信息 ILF
通过 EI 读取的每个 ILF/EIF 计算为一个 FTR。	读取了职位信息 ILF
既被 EI 维护又被读取的 ILF 仅计算一次。	职位信息 ILF 既被维护又被读取，计算一次
FTR 个数总计：1 个	

DET 计数规则	是否应用此规则?
以用户角度识别的、通过应用系统边界输出到系统外部的非重复字段，应算作一个 DET。	Total Job number
对于用户可识别的进入系统边界的非重复字段，若是用来说明 when/why 基本处理会产生或得到这个数据。	此处不使用
如果一个 DET 既进入又离开边界，仅记作当前基本流程的一个 DET	此处不使用
虽然有多种方法调用相同的逻辑处理，但能够说明具体采用的动作也算作一个 DET。	统计职位总数操作
如果字段在基本流程处理时从 ILF 里中获得或者派生，但没有穿过系统边界，它也不能算作 DET。	此处不使用
不要把文字记作 DET	
不要把 paging variables 或者系统产生的标志记作 DET	
DET 个数总计：2 个	



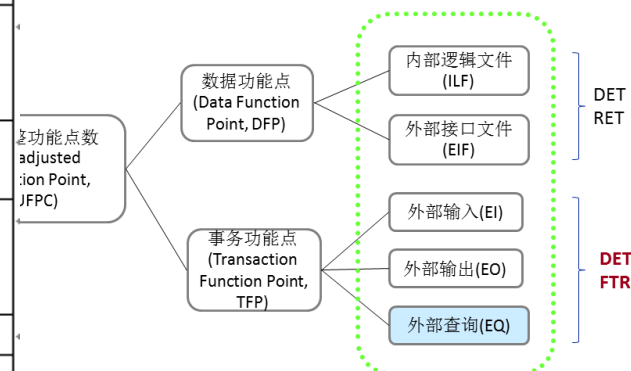


# 识别EQ

- EQ（查看新添加的职位信息）包含的DET和FTR:

FTR 计数规则	是否应用此规则?
它所维护的每一个 ILF 算做一个 FTR。	维护了职位信息 ILF
通过 EQ 读取的每个 ILF/EIF 计算为一个 FTR。	读取了职位信息 ILF
既被 EQ 维护又被读取的 ILF 仅计算一次。	职位信息 ILF 既被维护又被读取，计算一次
FTR 个数总计：1 个	

DET 计数规则	是否应用此规则?
以用户角度识别的、通过应用系统边界输出到系统外部的非重复字段，应算作一个 DET。	Job number, Job name Job pay grade Job description line number(Repeated) Job description line(Repeated)
对于用户可识别的进入系统边界的非重复字段，若是用来说明 when/why 基本处理会产生或得到这个数据。	此处不使用
如果一个 DET 既进入又离开边界，仅记作当前基本流程的一个 DET	此处不使用
如果能把说明产生错误、处理已经完成或者处理需要继续的系统响应消息传到系统边界外，这类数据也算作 DET。	此处不使用
虽然有多种方法调用相同的逻辑处理，但能够说明具体采用的动作也算作一个 DET。	查看操作
如果字段在基本流程处理时从 ILF 里中获得或者派生，但没有穿过系统边界，它也不能算作 DET。	此处不使用
不要把文字记作 DET	
不要把 paging variables 或者系统产生的标志记作 DET	
DET 个数总计：6 个	





# 主要内容

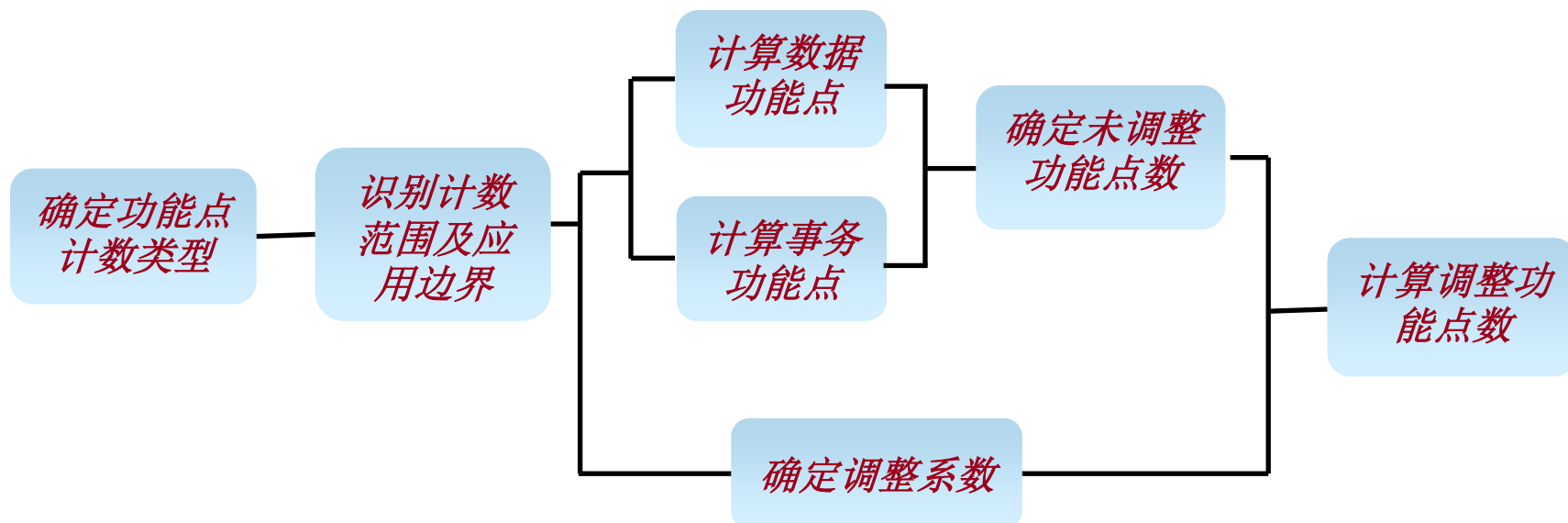
---

- 一、关于**IFPUG**组织
- 二、功能点分析方法概述
- 三、**IFPUG**方法的五大组件
- **四、IFPUG方法的估算流程**
- 五、估算实例



## 四、IFPUG方法估算流程（1）

- 估算步骤：
  - 1、确定功能点计数类型
  - 2、识别计数范围及应用边界
  - 3、确定未调整功能点数
  - 4、确定调整系数值
  - 5、应用公式计算FP值





## 四、IFPUG方法估算流程（2）

- **1、确定功能点计数类型**
  - 新开发项目类型：计算用户第一次安装该软件后，能使用的功能。
  - 增强型项目类型：仅计算对现有系统进行增加、删除或修改的功能。
  - 应用程序类型：计算当前系统的功能点个数。每当增强型项目更改系统的功能后，应用程序类型会更新当前的功能点个数。
- **2、识别计数范围及应用边界**
  - 确定系统包含哪些功能模块。





## 四、IFPUG方法估算流程（3）

### 3、确定未调整功能点数

表1 ILF 和 EIF 数据组件的复杂度级别

DET \ RET	1~19	20~50	≥51
1	低	低	平均
2~5	低	平均	高
≥6	平均	高	高

表2 EI 的复杂度级别

DET \ FTR	1-4	5~15	≥16
0~1	低	低	平均
2	低	平均	高
≥3	平均	高	高

表3 EO 和 EQ 的复杂度级别

DET \ FTR	1~5	6~19	≥20
0~1	低	低	平均
2	低	平均	高
≥3	平均	高	高

表4 功能要素复杂度加权因子表

功能要素	低	平均	高
外部输入数 EI	3	4	6
外部输出数 EO	4	5	7
外部查询表 EQ	3	4	6
内部逻辑文件数 ILF	7	10	15
外部接口文件数 EIF	5	7	10





## 四、IFPUG方法估算流程（4）

- 4、确定调整系数值(VAF, value adjustment factor)

GSC (general system characters)

调节参数	描述
E1	数据通讯
E2	软件性能
E3	可配置性
E4	事务效率
E5	实时数据输入
E6	用户界面复杂度
E7	在线升级
E8	复杂运算
E9	代码复用性
E10	安装简易性
E11	操作方便性
E12	跨平台要求
E13	可扩展性
E14	分布式数据处理

DI (degree of influence)

分数	系统影响程度
0	未显示出来或者无影响
1	偶然的影响
2	适度的影响
3	一般的影响
4	显著的影响
5	至始至终的强烈影响

$$VAF = TDI * 0.01 + 0.65 \quad (TDI = 14 \text{ 个系统调整因子 } DI \text{ 的总和})$$



## 四、IFPUG方法估算流程（5）

### • 5、应用公式计算FP值

■ 原始计算公式:  $FP\ Count = UFP * VAF$

■ 新开发类型:

–  $FP\ Count = (UFP + CFP) * VAF$

- CFP (conversion function point): 考虑系统整合或数据迁移部分的工作量。

■ 增强类型:

–  $FP\ Count = [(ADD + CHGA + CFP) * VAFA] + (DEL * VAFB)$

- ADD: 被添加的功能点个数
- CHGA: 所改动的功能点个数
- DEL: 被删除的功能点个数。
- VAFA: 功能增强后的功能点调整因子
- VAFB: 功能增强前的功能点调整因子



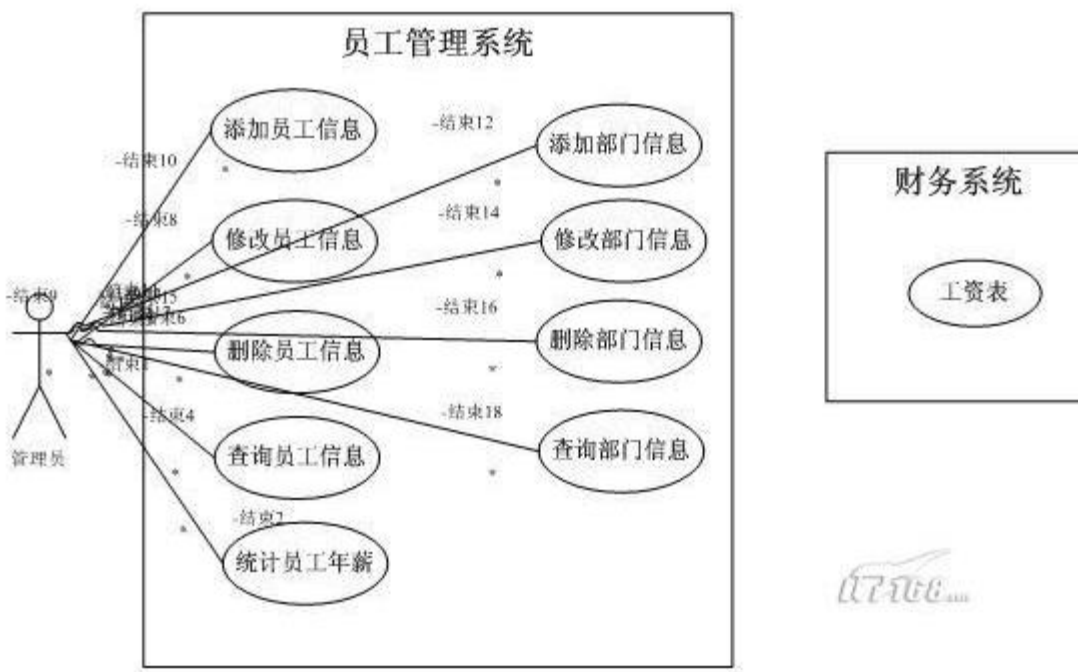
# 主要内容

---

- 一、关于**IFPUG**组织
- 二、功能点分析方法概述
- 三、**IFPUG**方法的五大组件
- 四、**IFPUG**方法的估算流程
- 五、估算实例

## 五、估算应用实例

- 例子：在员工管理系统中添加一个员工资料，会使用到员工的一般信息、教育情况、工作经历。员工隶属于某个部门，在本系统中会有一个对部门进行维护的功能。员工的工资则由另外一个财务系统提供。因此，其用例图如下所示：





## 五、估算应用实例

假设员工基本信息如下所示：

员工ID  
员工名称  
性别  
生日  
婚否  
所属部门名称  
工资等级  
——受教育的时间  
——学校名称  
——所学专业  
  
——工作时间  
——工作单位  
——工作部门

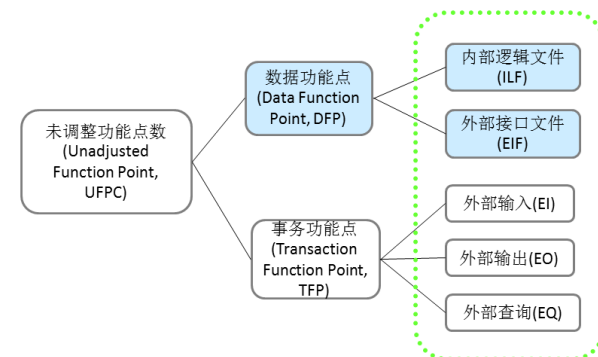
假设部门信息如下所示

:

部门编号  
部门名称  
部门描述信息

假设工资表信息如下所示：

工资等级  
金额  
单位





## 五、估算应用实例

- 由需求可知，员工管理系统主要有两大功能：

- 员工管理和部门管理

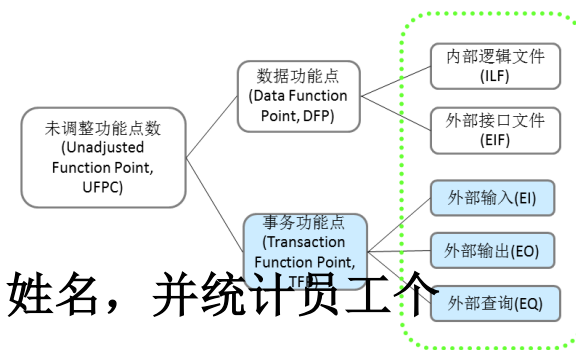
- 相关基本流程：

- 员工管理：

- 添加员工信息
- 修改员工信息
- 删除员工信息
- 查看员工列表（只显示员工ID，姓名和部门名称）
- 查询员工信息
- 统计员工年薪

- 部门管理

- 添加部门信息
- 修改部门信息
- 删除部门（下属员工也删除）
- 统计部门员工总数（显示下属员工信息ID、姓名，并统计员工个数）





## 五、估算应用实例

- 识别数据功能
  - **ILF**: 员工信息、部门信息
  - **EIF**: 工资信息

	DET	RET	复杂度	功能点数
ILF				
员工信息	ID /姓名/性别/生日/婚否/所属部门名称/工资等级/工作时间/工作单位/工作部门/受教育时间/学校/专业 合计: 13 DETs	基本个人信息 工作经历 受教育情况 合计: 3 RETs	低	7
部门信息	部门编号、部门名称、部门描述 合计: 3 DETs	部门基本信息 合计: 1 RET	低	7
EIF				
工资信息	工资等级、金额、单位 合计: 3 DETs	工资基本信息 合计: 1 RET	低	5
数据功能点数总计: 19FPs				







## 五、估算应用实例

- 识别事务功能——EI

EI	DET	FTR	复杂度	功能点数
添加员工信息	ID /姓名/性别/生日/婚否/所属部门名称/工资等级/工作时间/工作单位/工作部门/受教育时间/学校/专业 合计： 13 DETs	员工信息 合计： 1 FTRs	低	3
修改员工信息	同上	同上	低	3
删除员工	同上	同上	低	3
添加部门信息	部门编号、部门名称、部门描述 合计： 3 DETs	部门信息 合计： 1 FTR	低	3
修改部门信息	同上	同上	低	3
删除部门	员工信息全部的字段、部门信息的全部字段 合计： 15 DETs	员工信息、部门信息 合计： 2 FTRs	中	4
EI功能点数： 19FPs				



## 五、估算应用实例

- 识别事务功能——EO、

EO	DET	FTR	复杂度	功能点数
统计员工年薪	ID /姓名/工资等级/金额/单位 合计： 5 DETs	员工信息、 工 资信息 合计： 2 FTRs	低	4
统计部门员工 总数	ID /姓名/部门编号/部门名称/ 员工总数 合计： 5 DETs	员工信息、 部 门信息 合计： 2 FTRs	低	4
EQ				
查看员工列表	员工ID/员工姓名/所属部门 合计： 3 DETs	员工信息 合计： 1 FTRs	低	3
查询员工信息	员工信息的全部字段 合计： 13 DETs	员工信息、 部 门信息 合计： 2 FTRs	中	4
EO/EQ功能点数： 15FPs				



## 五、估算应用实例

功能点总数： 53FPs	
数据功能点数	ILF: 14FPs
	EIF: 5FPs
事务功能点数	EI: 19FPs
	EO: 8FPs
	EQ: 7FPs



---

• **Thanks!**