

# IFPUG

INTERNATIONAL FUNCTION POINT USERS GROUP

191 Clarksville Road  
Princeton Junction, NJ 08550  
F 609.799.4900  
F 609.799.7032



A WORLD OF INFORMATION

# Introduction to Function Points

~~Scott Goldfarb~~

~~Mary S. Bradley~~

~~Mauricio Aguiar~~

~~Tom Cagley~~

International Function Point Users Group

191 Clarksville Rd.

Princeton Junction, NJ 08550

Tel: 609-799-4900

Email: [ifpug@ifpug.org](mailto:ifpug@ifpug.org)

Web: [www.ifpug.org](http://www.ifpug.org)

## Credits:

The International Function Point Users Group (IFPUG) would like to thank the following individuals and companies for their contributions to this presentation:

- Mary S. Bradley
- Mick Burn-Murdoch
- Carol Dekkers
- Sheila Dennis
- David Garmus
- Scott Goldfarb
- Cindy Woodrow
- Steven Woodward



# Agenda

- ▶ **Introduction**
- ▶ **Why use Function Points**
  - **Managing Your Software**
  - **Managing Your Organization**
  - **Function Points vs. Lines of Code**
- ▶ **How to Count Function Points**
- ▶ **Questions**

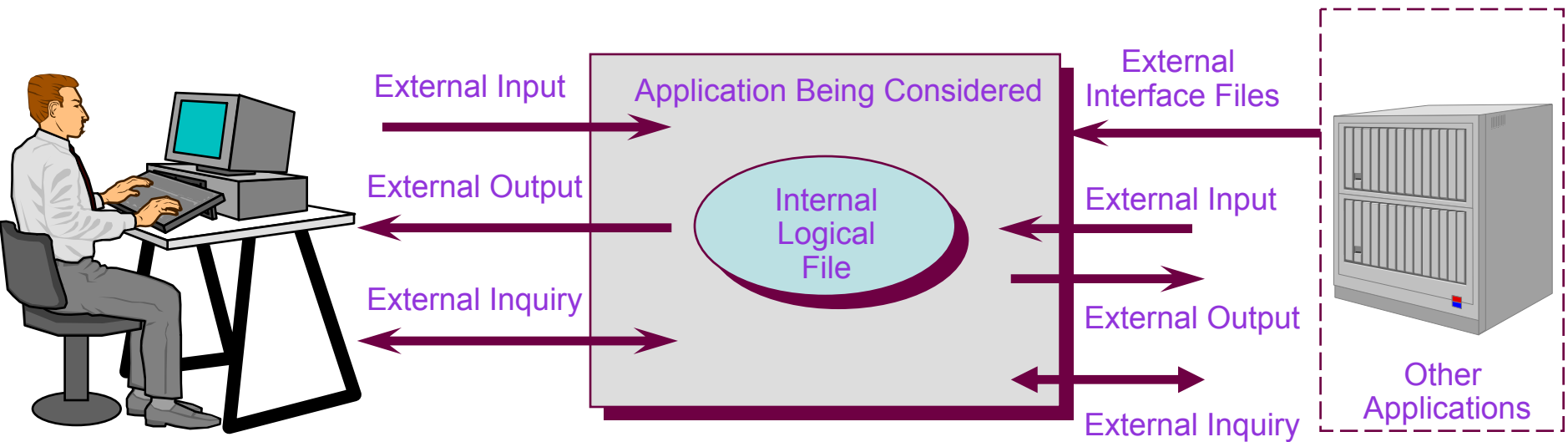


# Objectives of Function Point Analysis

- ▶ **Measures software by quantifying the functionality requested by and provided to the customer based primarily on logical design**
- ▶ **Measures software development and maintenance independently of technology used for implementation**
- ▶ **Measures software development and maintenance consistently across all projects and organizations**



# Function Points are a Unit of Measure



- Functionality as viewed from the user's perspective



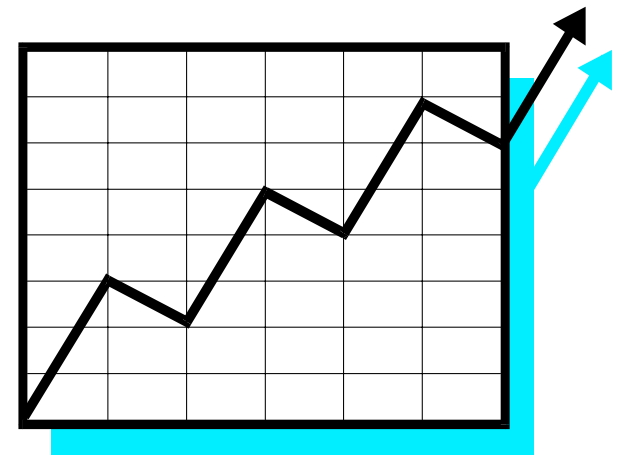
# Why Use Function Points

## ▶ **Managing Your Software**



# Software Development Challenges

- ▶ **Size of Requirements**
- ▶ **Changes to Requirements**
- ▶ **Estimation Based on Requirements**
- ▶ **Measuring and Improving Productivity and Quality**



# Size of Requirements

- ▶ **Requirements**
  - **Complete**
  - **Business Terms**
  - **Mutual Understanding**
  - **Document Assumptions**
  - **Size**

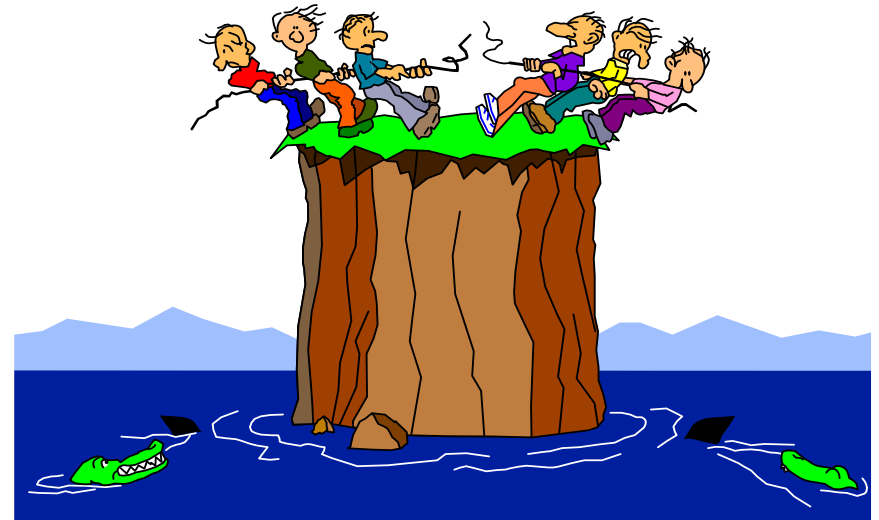




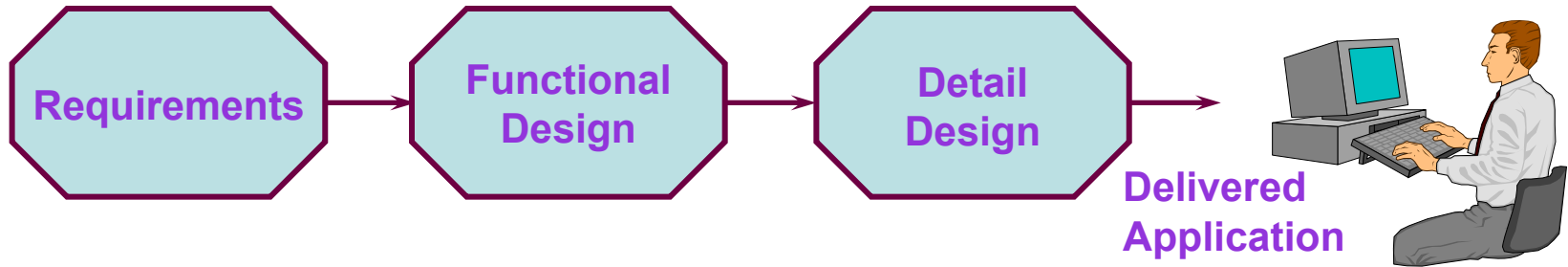
# Changes to Requirements

## ► Changes to Requirements

- Change Inevitable
- Trade-offs
- Customer Definition of Quality
- Size



# Changes to Requirements



**100 FPs**

**120 FPs**

**130 FPs**

**135 FPs**

- State code input screen changed (3 FPs)
- Interface to N&A file added (10 FPs)
- N&A inquiry and state code inquiry added (7 FPs)

- New regulatory table added (10 FPs)

- Summary report added (5 FPs)

**Impact**

**Effort**  
**Schedule**  
**Cost**

+ 1 month  
+ 2 weeks  
+ \$5 K

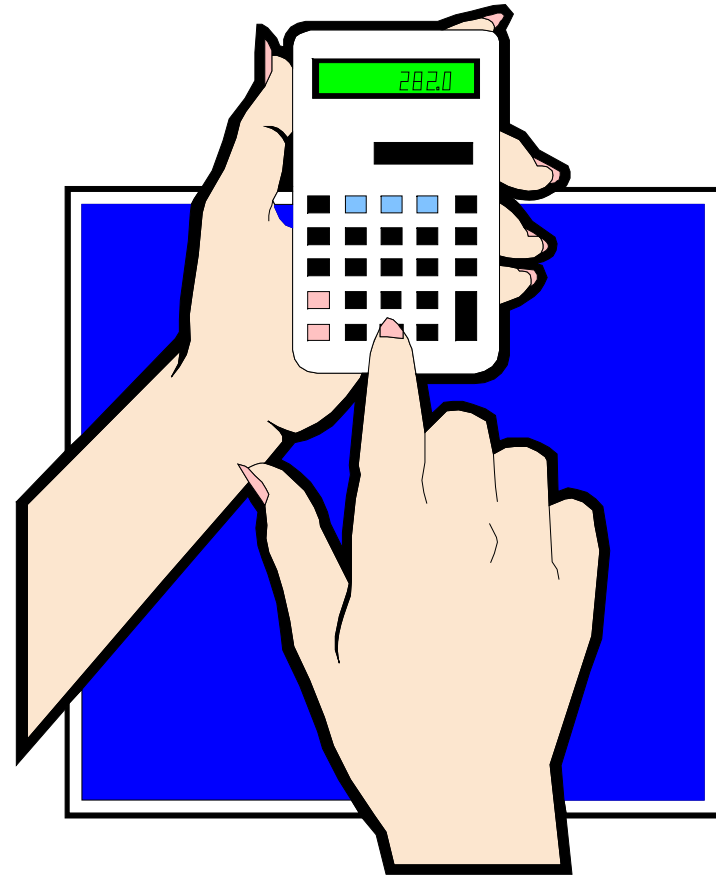
+ .5 month  
+ 1 week  
+ \$2.5 K

+ .25 month  
+ 2.5 days  
+ \$1.25 K



# Estimation Based on Requirements

- ▶ **Estimation Based on Requirements**
  - **Multiple Models**
  - **Weighted Inputs:**
    - **Language**
    - **Skills**
    - **Methodology**
    - **Risk Factors**
    - **Size**
  - **Historical Base**



# Estimating Examples

## Function Point Size

---

Project A – 100 FPs

Project B – 100 FPs

## Project Variables

---

- On-line/database
- New development
- C++
- Highly experienced development staff

- Batch
- Enhancement
- Cobol
- Average experienced development staff

## Project Estimate Based on Historical Data and/or Vendor Tool

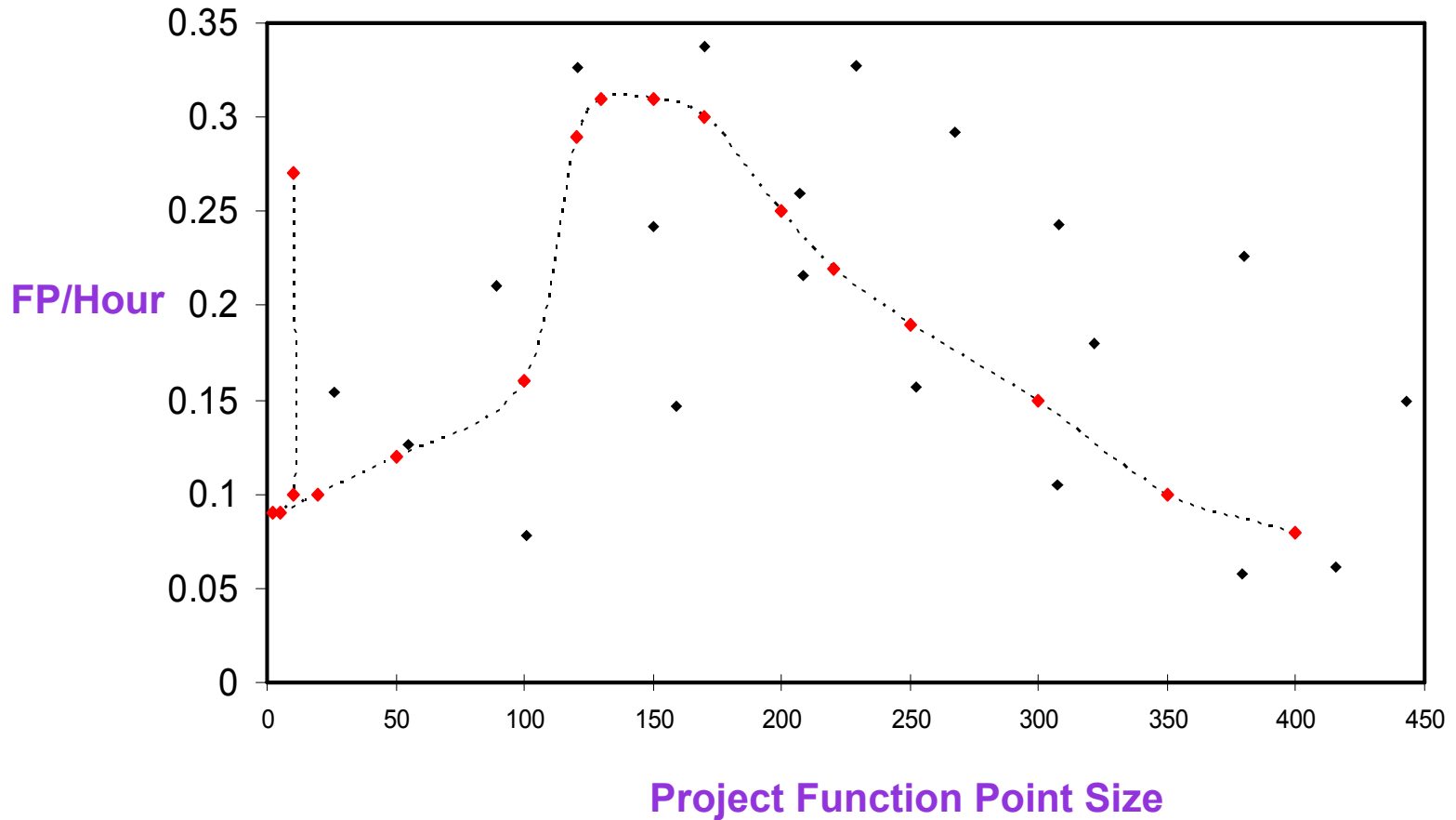
---

**Effort = 5 months**  
**Schedule = 3 months**  
**Cost (@ \$5K) = \$25,000**  
**KLOC = 6**  
**Delivered Defects = 25**  
**Productivity Rate = 20**  
**FP/Month.**

**Effort = 20 months**  
**Schedule = 6 months**  
**Cost (@ \$5K) = \$100,000**  
**KLOC = 10**  
**Delivered Defects = 100**  
**Productivity Rate = 5 FP/Month**



# Measuring and Improving Productivity



- ▶ Every organization has an optimum size/productivity range

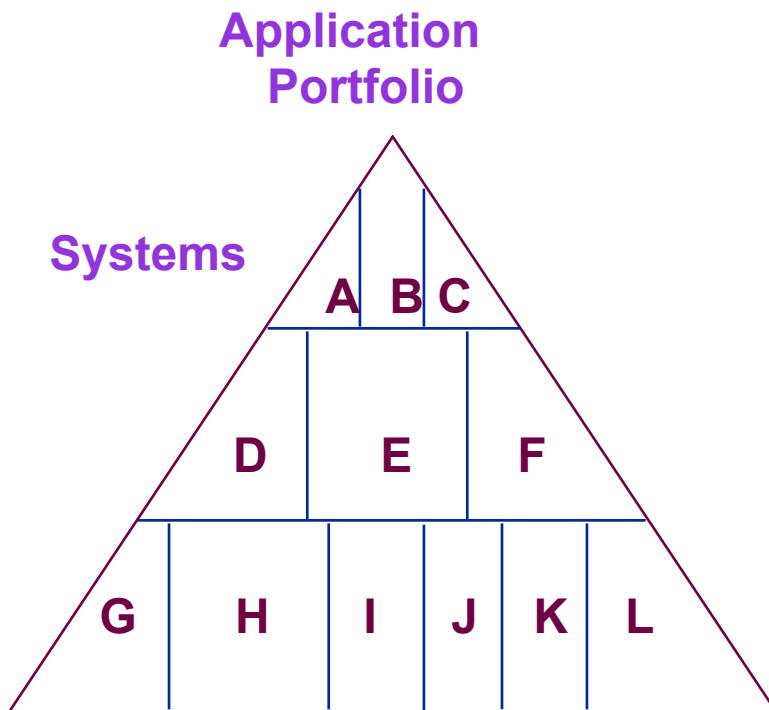


# Why Use Function Points

## ▶ Managing Your Organization



# Asset Management



**Size = 50,000 Function Points**

**Replacement Cost =  
\$300,000,000**

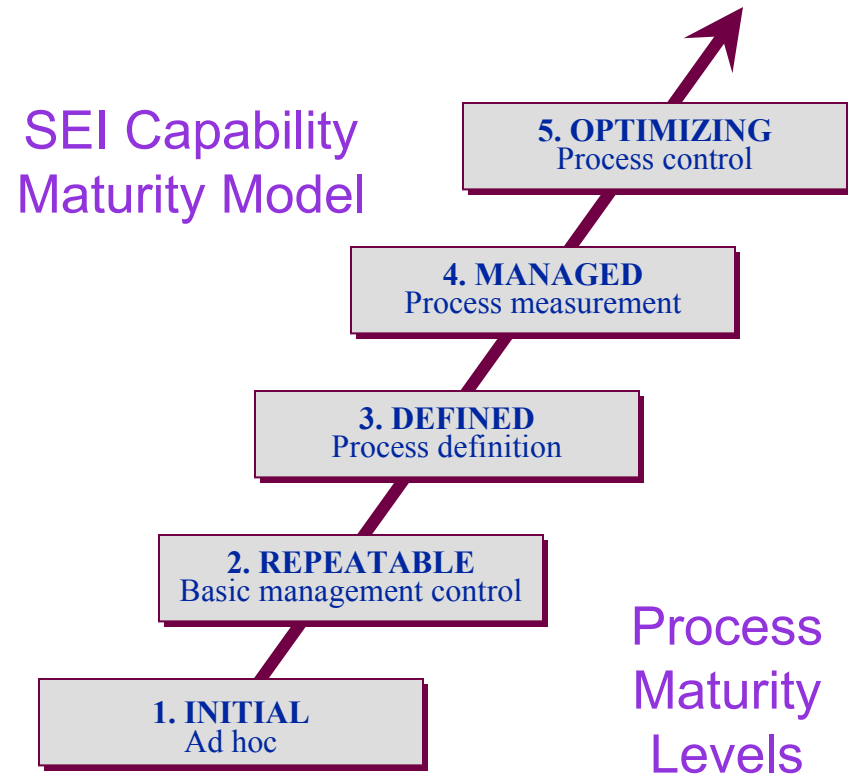
**Growth = 7% per year**

**Support Cost = \$20,000,000 per  
year**



# Function Points and the CMMI

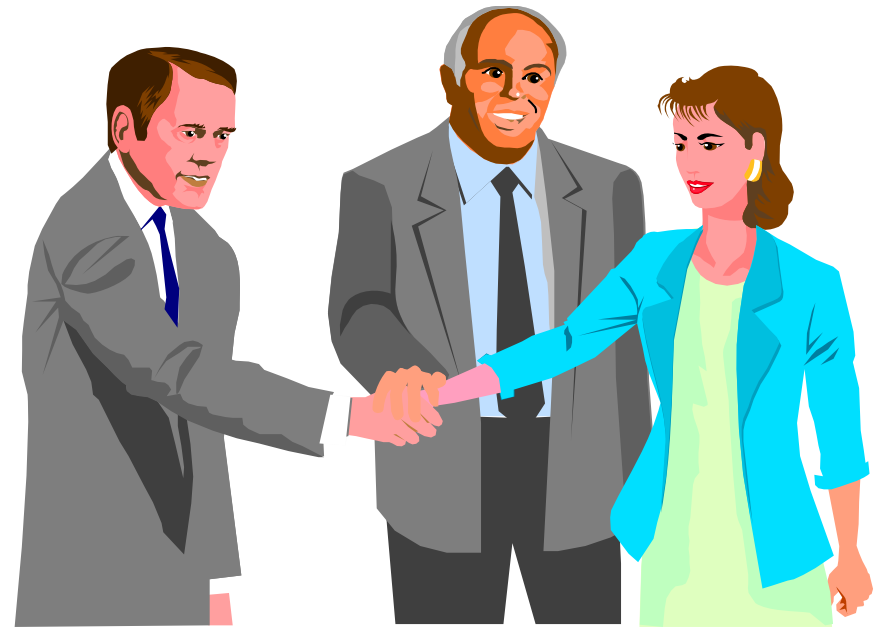
- ▶ **Function Points are the metric of choice for many of the activities required in the SEI CMMI Level 2**
- ▶ **With the next release of the CMMI, metrics becomes a Key Process Area in its own right**





# Improving Customer Relations

- ▶ **Predictable Time scales**
- ▶ **Predictable Costs**
- ▶ **Predictable Functionality**



# Organizational Improvement

- ▶ **Process Measurement**
- ▶ **Project Management Metrics**
  - Estimates
  - Productivity
  - Defect Densities
  - etc.
- ▶ **Benchmarking**



# Function Points & Metrics Help

- ▶ **Evaluate current in-house and contractor performance**
- ▶ **Establish quantifiable expectations**
- ▶ **Demonstrate objectives for contract/outsourcing are met**
- ▶ **Establish realistic commitments**
- ▶ **Determine fair compensation**
- ▶ **Establish “win win” relationships**



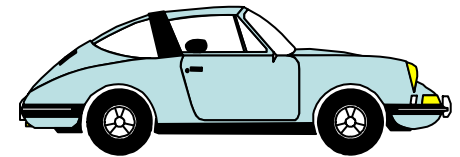
# Function Points vs. Lines of Code

- ▶ **Technology and platform independence**
- ▶ **Available from early requirements phase**
- ▶ **Consistent and objective unit of measure throughout the life cycle**
- ▶ **Objectively defines software application from the customer perspective**
- ▶ **Objectively defines a series of software applications from the customer's, not the technician's perspective**
- ▶ **Is expressed in terms that users can readily understand about their software**



# What is Wrong with LOC?

- ▶ **There is no standard for a line of code**
- ▶ **Lines of Code measure components, not completed products**
  - **Don't measure the panels produced; measure the number of cars assembled**
- ▶ **Measuring lines of code**
  - **Rewards profligate design**
  - **Penalizes tight design**
- ▶ **Positively misleading?**



# Classic Productivity Paradox

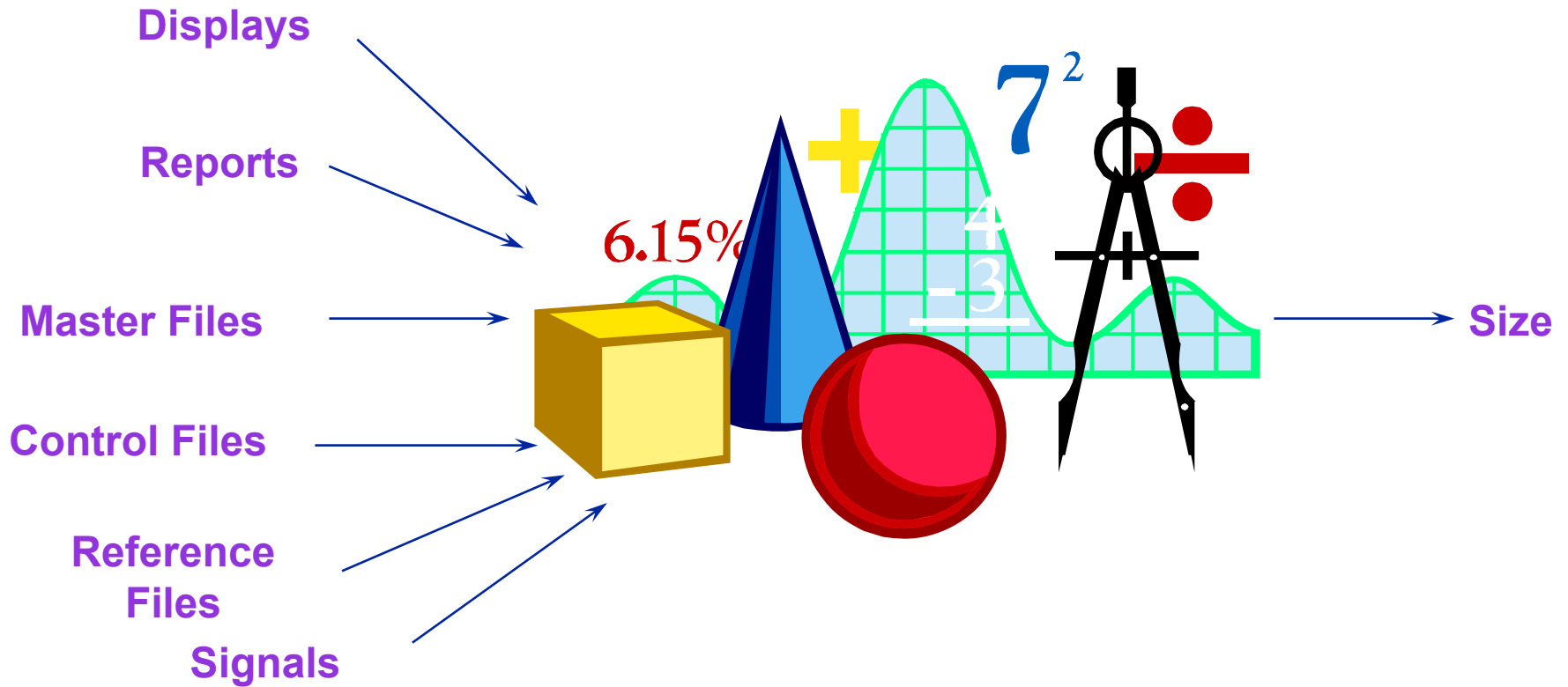
Lines of Code	10,000	3,000
Function Points	25	25
Total Months effort	25	15
Total Costs	\$125,000	\$75,000
Cost per Source Line	\$12.50	\$25.00
Lines per Person month	400	200
FPS per Person month	1.2	2
Cost per FP	\$5,000	\$3,000



# How to Count Function Points



# How to Count Function Points



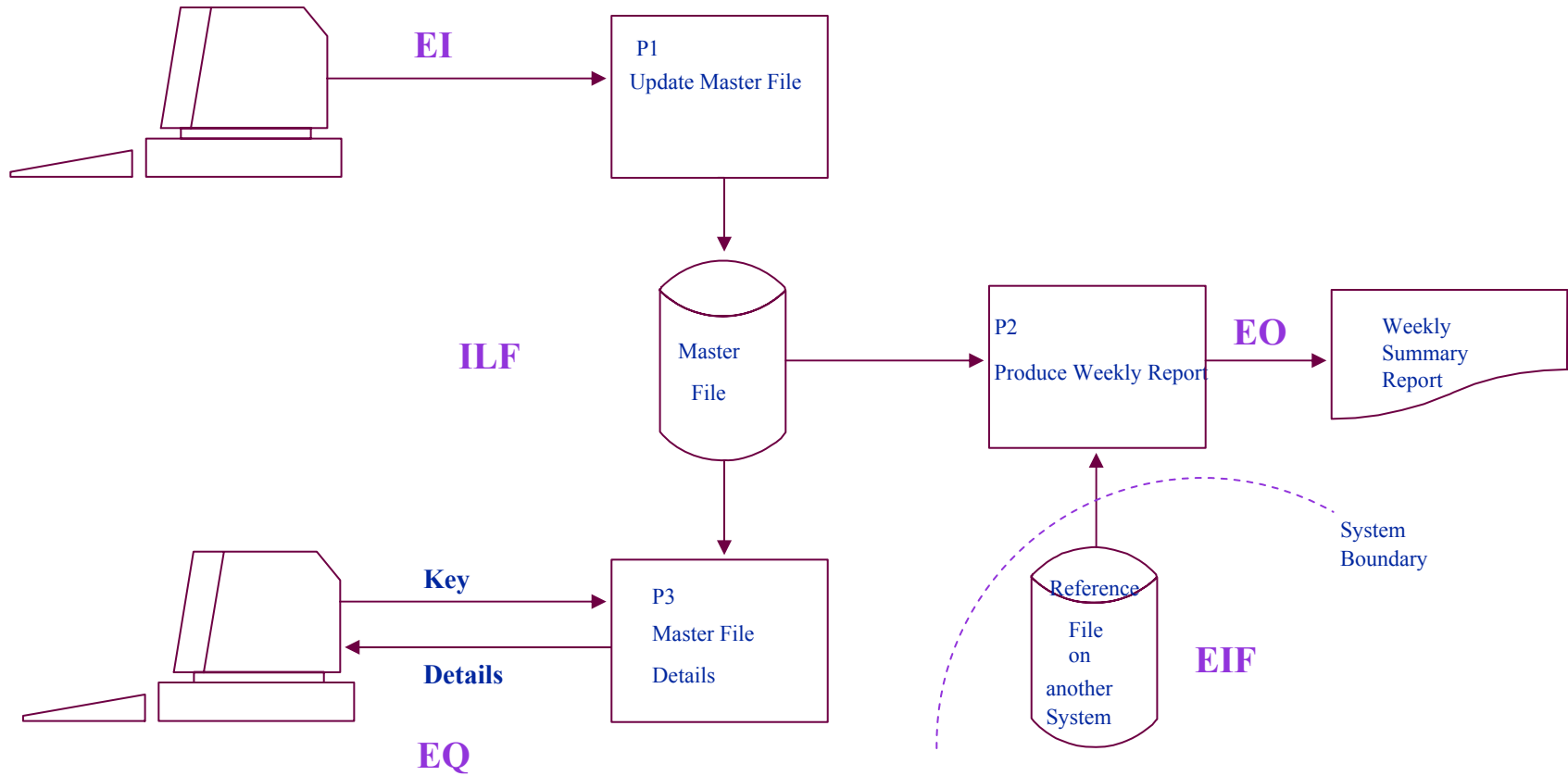


# Steps in FP Counting

- ▶ **Determine Type of Count (3 Types)**
  - Enhancement (Project) Function Point Count
  - Application Function Point Count
  - Development (Project) Function Point Count
- ▶ **Identify Counting Scope and Application Boundary**
- ▶ **Count Data Functions**
- ▶ **Count Transactional Functions**
- ▶ **Determine Unadjusted Function Point Count**
- ▶ **Determine Value Adjustment Factor**
- ▶ **Calculate Adjusted Function Point Count**

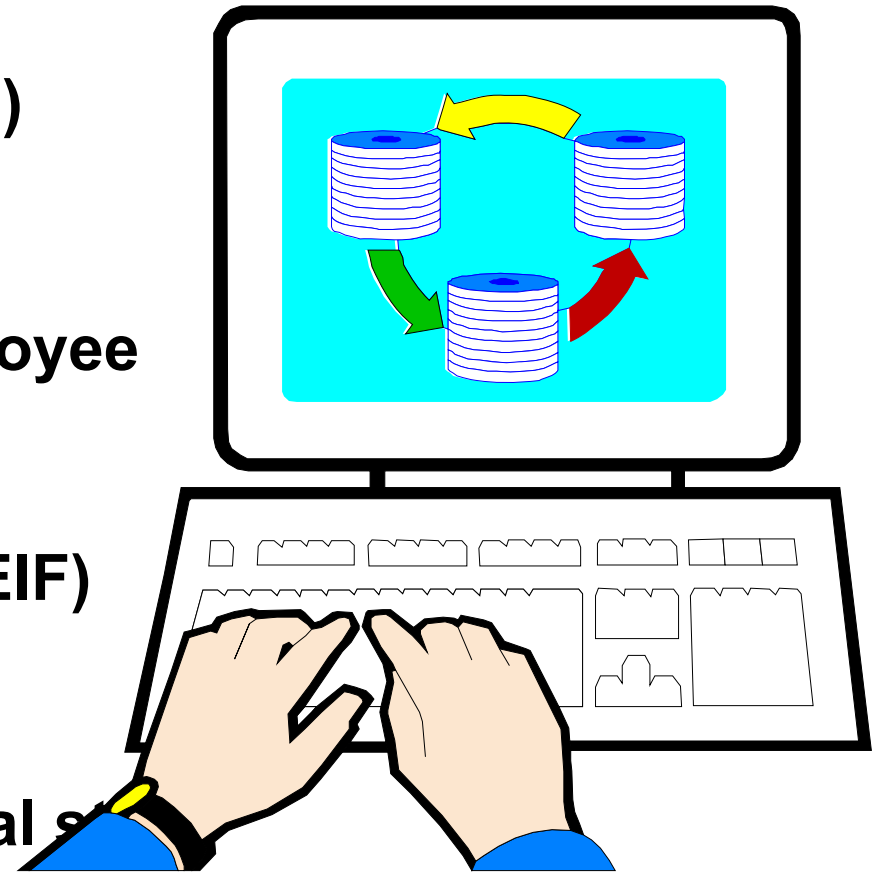


# FP Overview: What Is Counted



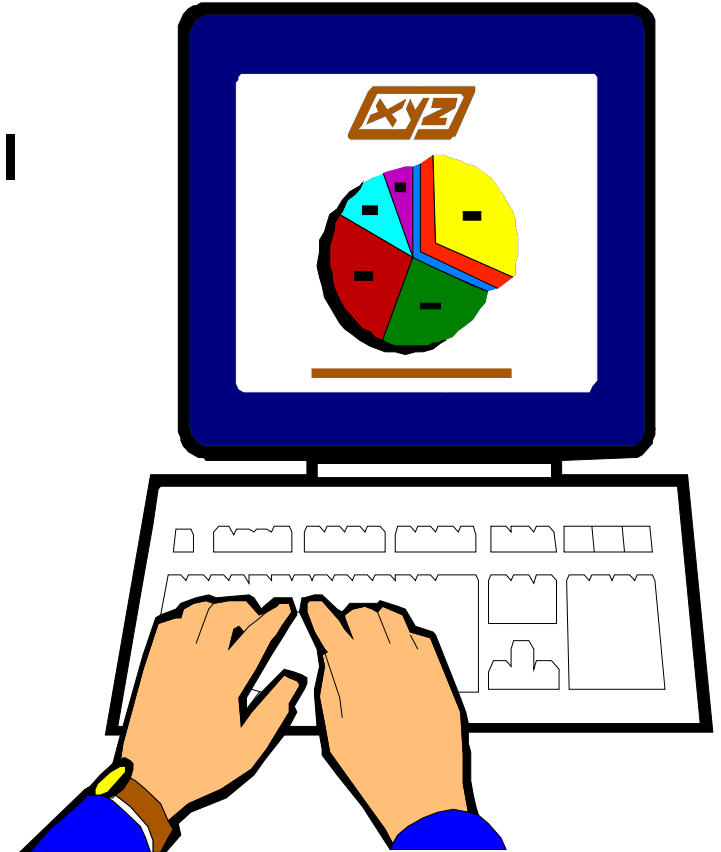
# Data Storage

- ▶ **Internal Logical File (ILF)**  
 Logical group of data maintained by the application (e.g., Employee file)
  
- ▶ **External Interface File (EIF)**  
 Logical group of data referenced but not maintained (e.g., Global table)



# Transactions

- ▶ **External Input (EI)**  
Maintains ILF or passes control data into the application
- ▶ **External Output (EO)**  
Formatted data sent out of application with added value (e.g., calculated totals)
- ▶ **External Query (EQ)**  
Formatted data sent out of application without added value



# Functions are Weighted Based on Complexity

## Data Element Types (DETs)

- **Number of user recognizable non-repeated fields**
- **Applies to data and transactional functions**

## File Types Referenced (FTRs)

- **Number of files referenced, read, created, or updated**
- **Applies to transactional functions**

## Record Element Types (RETs)

- **Number of data sub-groupings or unique record formats**
- **Applies to data functions**



# Functional Size (Unadjusted Function Size)

<b>Function Type</b>	<b>Low</b>	<b>Average</b>	<b>High</b>
<b>EI</b>	x 3	x 4	x 6
<b>EO</b>	x 4	x 5	x 7
<b>EQ</b>	x 3	x 4	x 6
<b>ILF</b>	x 7	x 10	x 15
<b>EIF</b>	x 5	x 7	x 10



# Value Adjustment Factor

- ▶ **Based on 14 General System Characteristics (User Business Constraints Independent of Technology)**
  - **Examples: data communications, response times, end user efficiency, multiple sites and flexibility**
- ▶ **Adjusts FP count by up to + / - 35%**

