1.0 Calculate Function Points Procedure

1. PURPOSE

The purpose of this procedure is to produce an estimate of software size from software requirements. Function Points are an indirect measure of software size based on external and internal application characteristics, as well as application performance. Function Points have a significant cost estimating relationship (CER) to software costs. Once determined, Function Points can be input into empirical statistical parametric software cost estimation equations and models in order to estimate software costs. Function Points are widely reported to be well suited for measuring the size of management information system (MIS), database intensive, and 4GL based application (e.g., software) system development.

2. REFERENCES

- Function Point Counting Practices Manual (Release 4.1), International Function Point User's Group (IFPUG), May 1999.
- Albrecht, Allan J., Measuring Application Development Productivity, Proceedings SHARE/GUIDE IBM Applications Development Symposium, October 1979.
- Lokan, C. J., An Empirical Analysis of Function Point Adjustment Factors, University of South Wales, December 1998.
- Longstreet, D., Function Points Step by Step, Longstreet Consulting, Inc., January 1999.

3. SCOPE

Function Points are indirect quantitative measures of application software functionality and size that are based on objective counts of external application interfaces factored together with subjective counts of internal application complexity and overall performance characteristics. This procedure is composed of three logical divisions, determining the unadjusted function point count, value adjustment factor, and Function Points.

Determining the unadjusted function point count consists of counting the number of external inputs, external outputs, external inquiries, internal logical files, and external interface files. Determining the value adjustment factor consists of rating system, input and output, and application complexity. Determining Function Points consists of factoring unadjusted function points and value adjustment factor together.

Function Points have two distinct purposes. The first purpose is to act as a basis for software measurement, comparison, and analysis (e.g., a software metrics approach). The second, and more important purpose, is to determine software size for input into software cost estimation models (e.g., equations) and tools that output effort (e.g., staff hours), which are based on empirical cost estimating relationships (CERs) between Function Points and effort.

4. SUMMARY

- Inputs and Outputs. The inputs listed are those which drive the processes and activities. The outputs are those produced by the processes and activities.
- Steps. Steps identify:
  - The high-level procedures that form the process
  - Skill requirements and roles required to perform the procedure
  - A brief description of the procedure purpose
- Workflow Diagrams. These provide a pictorial view of the sequence of the procedures, to include a second level of decomposition. The workflows should be reviewed along with the narrative to put the procedure into process context.
- Procedure Descriptions. Each of the high-level procedures identified under the "Steps" are decomposed into tasks. Procedure descriptions are attached and will be presented in the same format. The header information will link each procedure to its parent process.

5. ROLES AND RESPONSIBILITIES

Roles	Tasks
Task Manager	Task managers are responsible for estimating Function Points for software cost estimation purposes. Task managers must estimate unadjusted function points, which are based on external application interfaces, and value adjustment factors, which are based on application complexity and performance. Task managers must also determine the Function Points with assistance and input from technical staff as needed. Responsibilities occur in the following tasks: • 1.1 Determine Unadjusted Function Point Count • 1.2 Determine Value Adjustment Factor • 1.3 Determine Function Points

6. INPUTS AND OUTPUTS

Inputs	Description and Tasks
Transaction and File Inventory	Indirect quantitative measures of application software functionality and size that are based on objective counts of external application interfaces, such as external inputs, external outputs, external inquiries, internal logical files, and external interface files. ***External Inputs***: Elementary processes in which data crosses the boundary from outside to inside. This data may come from data input screens, electronically, or another application. The data can be control information or business information. If the data is business information it is used to maintain one or more internal logical files. If the data is control information it does not have to update an internal logical file.

Inputs	Description and Tasks
	External Outputs*:** Elementary processes in which derived data passes across the boundary from inside to outside. The data creates reports or output files sent to other applications. These reports and files are created from one or more internal logical files and external interface files. ***External Inquiries: Elementary processes with both input and output components that result in data retrieval from one or more internal logical files and external interface files. This information is sent outside the application boundary. The input process does not update any internal logical files and the output side does not contain derived data. ***Internal Logical Files***: User identifiable groups of logically related data that reside entirely within the application boundary and are maintained through external inputs. ***External Interface Files***: User identifiable groups of logically related data that are used for reference purposes only. The data reside entirely outside the application and are maintained by another application. The external interface file is an internal logical file for another application. This input criterion is required for the following tasks: • 1.1 Determine Unadjusted Function Point Count
General Systems Characteristics	Indirect and subjective measures of application software functionality and size that are based on individual rating of internal application complexity and overall performance characteristics, such as system, input and output, and application complexity. This input criterion is required for the following tasks: • 1.2 Determine Value Adjustment Factor

Outputs	Description and Tasks
Function Points	Function Points are measures of software size, functionality, and complexity used as a basis for software cost estimation. This output criterion is required for the following tasks: • 1.3 Determine Function Points

7. STEPS

Responsibility	Step	Activities
Task Manager	1.1	**Determine Unadjusted Function Point Count:** Determining the unadjusted function point count consists of counting the number of external inputs, external outputs, external inquiries, internal logical files, and external interface files, and determining the unadjusted function point count total.
Task Manager	1.1.1	**Determine External Inputs:** External inputs are elementary processes in which data crosses the boundary from outside to inside. This data may come from data input screens, electronically, or another application. The data can be control information or business information. If the data is business information it is used to maintain one or more internal logical files. If the data is control information it does not have to update an internal logical file. Multiply the count by 3, 4, or 6 for low, average, or high.
Task Manager	1.1.2	**Determine External Outputs:** External outputs are elementary processes in which derived data passes across the boundary from inside to outside. The data creates reports or output files sent to other applications. These reports and files are created from one or more internal logical files and external interface files. Multiply the count by 4, 5, or 7 for low, average, or high.
Task Manager	1.1.3	**Determine External Inquiries:** External inquiries are elementary processes with both input and output components that result in data retrieval from one or more internal logical files and external interface files. This information is sent outside the application boundary. The input process does not update any internal logical files and the output side does not contain derived data. Multiply the count by 3, 4, or 6 for low, average, or high.
Task Manager	1.1.4	**Determine Internal Logical Files:** Internal logical files are user identifiable groups of logically related data that reside entirely within the application boundary and are maintained through external inputs.

		Multiply the count by 7, 10, or 15 for low, average, or high.
Task Manager	1.1.5	**Determine External Interface Files:** External interface files are a user identifiable group of logically related data that are used for reference purposes only. The data reside entirely outside the application and are maintained by another application. The external interface file is an internal logical file for another application. Multiply the count by 5, 7, or 10 for low, average, or high.
Task Manager	1.1.6	**Determine Unadjusted Function Point Count Total:** Add the weighted number of external inputs, external outputs, external inquiries, internal logical files, and external interface files together. The result is the unadjusted function points.
Task Manager	1.2	**Determine Value Adjustment Factor:** Determining value adjustment factor consists of rating system, input and output, and application complexity, and determining value adjustment score.
Task Manager	1.2.1	**Rate System Complexity:** Rating system complexity consists of rating data communications, distributed data processing, performance, and heavily used configuration complexity, and determining system complexity. Rate each of the four system complexity factors on an ordinal scale from 0 to 5, representing no influence, incidental, moderate, average, significant, and essential.
Task Manager	1.2.1.1	**Rate Data Communication Complexity:** How many data communication facilities are there? Data communication describes the degree to which the application communicates directly with the processor. The data and control information used in the application are sent or received over communication facilities. Terminals connected locally to the control unit are considered to use communication facilities. A protocol is a set of conventions, which permit the transfer or exchange of information between two systems or devices. All data communication links require some type of protocol. Scoring guidelines are as follows:

		0 Application is pure batch processing or a standalone PC. 1 Application is batch but has remote data entry or remote printing. 2 Application is batch but has remote data entry and remote printing. 3 Application includes online data collection or TP (teleprocessing) front end to a batch process or query system. 4 Application is more than a front-end, but supports only one type of TP communications protocol. 5 Application is more than a front-end, and supports more than one type of TP communications protocol. Teleprocessing is now almost universal. Only 10% of projects report a "below average" score of 2 or less; 56% have an "above average" score of 4 or 5. The scores tend to be lower for banking projects, and for projects developed on personal computers. There has been a gradual decline, from above average towards average, from 1991 to 1996.
Task Manager	1.2.1.2	**Rate Distributed Data Processing Complexity:** How are distributed data and processing functions handled? Distributed data processing describes the degree to which the application transfers data among components of the application. Distributed data or processing functions are a characteristic of the application within the application boundary. Scoring guidelines are as follows: 0 Application does not aid the transfer of data or processing function between components of the system. 1 Application prepares data for end user processing on another component of the system such as PC spreadsheets and PC DBMS. 2 Data is prepared for transfer, then is transferred and processed on another component of the system (not for end-user processing). 3 Distributed processing and data transfer are online and in one direction only. 4 Distributed processing and data transfer are online and in both directions. 5 Processing functions are dynamically performed on the most appropriate component of the system.

		Of all the general systems characteristics, this has one of the greatest percentages of “below average” values. The distribution is bimodal: systems tend to be either monolithic, or to have distributed processing as a characteristic of above average importance. There tends to be more distributed processing in engineering systems. Distributed processing is more common on midrange platforms than on other platforms. It is more common in transaction/production systems and office information systems than in management information systems and decision support systems. It is more important in new developments than in enhancement projects.
Task Manager	1.2.1.3	**Rate Performance Complexity:** Was response time or throughput required by the user? Performance describes the degree to which response time and throughput performance considerations influenced the application development. Application performance objectives, stated or approved by the user, in either response or throughput, influence (or will influence) the design, development, installation, and support of the application. Scoring guidelines are as follows: 0 No special performance requirements were stated by the user. 1 Performance and design requirements were stated and reviewed but no special actions were required. 2 Response time or throughput is critical during peak hours. No special design for CPU utilization was required. Processing deadline is for the next business day. 3 Response time or throughput is critical during all business hours. No special design for CPU utilization was required. Processing deadline requirements with interfacing systems are constraining. 4 In addition, stated user performance requirements are stringent enough to require performance analysis tasks in the design phase. This characteristic has a broad spread: scores are below average for 32% of projects, average for 30%, and above average for 38%.

		Performance is more important for transaction/production systems than for management information systems. It is more important for new developments than for enhancement projects.
Task Manager	1.2.1.4	**Rate Heavily Used Configuration Complexity:** How heavily used is the current hardware platform? Heavily used configuration describes the degree to which computer resource restrictions influenced the development of the application. A heavily used operational configuration, requiring special design considerations, is a characteristic of the application. For example, the user wants to run the application on existing or committed equipment that will be heavily used. Scoring guidelines are as follows: 0 No explicit or implicit operational restrictions are included. 1 Operational restrictions do exist, but are less restrictive than a typical application. No special effort is needed to meet the restrictions. 2 Some security or timing considerations are included. 3 Specific processor requirements for a specific piece of the application are included. 4 Stated operation restrictions require special constraints on the application in the central processor or a dedicated processor. 5 In addition, there are special constraints on the application in the distributed components of the system. Scores for this characteristic are generally low: 66% below average, 20% average, and 14% above average. Scores are lower for transaction/production systems and office information systems than for management information systems and decision support systems. They are lower for new developments than for enhancement projects; higher for midrange projects than for other platforms; and higher for engineering systems. Scores increase from 3GL projects to 4GL projects to application generator projects.
Task Manager	1.2.1.5	**Determine System Complexity:** Add the four weighted scores together for system complexity.
Task Manager	1.2.2	**Rate Input and Output Complexity:** Rating input and output complexity consists of rating transaction

		rate, on-line data entry, end-user efficiency, and on-line update complexity, and determining input and output complexity. Rate each of the four input and output complexity factors on an ordinal scale from 0 to 5, representing no influence, incidental, moderate, average, significant, and essential.
Task Manager	1.2.2.1	**Rate Transaction Rate Complexity:** How frequently are transactions executed? Transaction rate describes the degree to which the rate of business transactions influenced the development of the application. If the transaction rate is high and it influences the design, development, installation, and support of the application. Scoring guidelines are as follows: 0 No peak transaction period is anticipated. 1 Peak transaction period (e.g., monthly, quarterly, seasonally, annually) is anticipated. 2 Weekly peak transaction period is anticipated. 3 Daily peak transaction period is anticipated. 4 High transaction rate(s) stated by the user in the application requirements or service level agreements are high enough to require performance analysis tasks in the design phase. 5 High transaction rate(s) stated by the user in the application requirements or service level agreements are high enough to require performance analysis tasks and, in addition, require the use of performance analysis tools in the design, development, and/or installation phases. Scores for transaction rate are shared across the range from 0 to 4; there are few scores of 5. Transaction rates are more important than usual for banking systems, and less for engineering systems. They are more important on mainframes than on other platforms. Although one might expect them to be more important for transaction/ production systems, there are no significant differences between application types. Scores for this characteristic rise steadily from 1991 through 1996.
Task Manager	1.2.2.2	**Rate On-Line Data Entry Complexity:** What percentage of the information is entered on-line?

		Online Data Entry describes the degree to which data is entered through interactive transactions. Online data entry and control functions are provided in the application. Scoring guidelines are as follows: 0 All transactions are processed in batch mode. 1 1% to 7% of transactions are interactive data entry. 2 8% to 15% of transactions are interactive data entry. 3 16% to 23% of transactions are interactive data entry. 4 24% to 30% of transactions are interactive data entry. 5 More than 30% of transactions are interactive data entry. This characteristic has by far the highest scores of all, and the least variation. Fully 60% of projects score this characteristic at 5, the maximum value possible. According to IFPUG's guidelines, a score of 5 means that more than 30% of transactions involve interactive data entry. Perhaps 30% is too low a threshold nowadays; a higher value might provide more useful discrimination. The scores are lower (generally 3's) for a block of COBOL/ mainframe/banking projects from one organization. The score is 5 for just about everything else.
Task Manager	1.2.2.3	**Rate End-User Efficiency Complexity:** Was the application designed for end-user efficiency? End-user efficiency describes the degree of consideration for human factors and ease of use for the user of the application measured. Online functions emphasize a design for end-user efficiency (e.g., navigational aids, menus, online help and documents, automated cursor movement, scrolling, remote printing via online transactions, and pre-assigned function keys). Scoring guidelines are as follows: 0 None of the above. 1 One to three of the above. 2 Four to five of the above. 3 Six or more of the above, but there are no specific user requirements related to efficiency. 4 Six or more of the above, and stated requirements for end-user efficiency are strong enough to require design tasks for

		human factors to be included (for example, minimize key strokes, maximize defaults, use of templates). 5 Six or more of the above, and stated requirements for end-user efficiency are strong enough to require use of special tools and processes to demonstrate that the objectives have been achieved. This characteristic has a broad spread, with a slight tendency towards higher scores: 34% are below average, and 43% are above average. End user efficiency is more important for management information systems than for transaction/production systems. The scores are lower, and have a flatter distribution, for new developments than for enhancement projects. Again, scores increase from 3GL projects to 4GL projects to application generator projects.
Task Manager	1.2.2.4	**Rate On-Line Update Complexity:** How many internal logical files are updated by on-line transaction? Online update describes the degree to which internal logical files are updated online. The application provides online update for the internal logical files. Scoring guidelines are as follows: 0 None. 1 Online update of one to three control files is included. Volume of updating is low and recovery is easy. 2 Online update of four or more control files is included. Volume of updating is low and recovery easy. 3 Online update of major internal logical files is included. 4 In addition, protection against data lost is essential and has been specially designed and programmed in the system. 5 In addition, high volumes bring cost considerations into the recovery process. Highly automated recovery procedures with minimum operator intervention are included. Scores for on-line update tend to be high (half are above average), but they are mostly 3's and 4's rather than 5's. Scores tend to be higher for transaction/production systems. They are lower on personal computers than on other platforms. Once

		again, scores increase from 3GL projects to 4GL projects to application generator projects.
Task Manager	1.2.2.5	**Determine Input and Output Complexity:** Add the four weighted scores together for input and output complexity.
Task Manager	1.2.3	**Rate Application Complexity:** Rating application complexity consists of rating complex processing, reusability, installation ease, operational ease, multiple sites, and facilitate change complexity, and determining application complexity. Rate each of the six application complexity factors on an ordinal scale from 0 to 5, representing no influence, incidental, moderate, average, significant, and essential.
Task Manager	1.2.3.1	**Rate Complex Processing Complexity:** Does the application have extensive logical or math processing? Complex processing describes the degree to which processing logic influenced the development of the application. The following components are present: sensitivity control, special audit processing, security processing, logical processing, mathematical processing, exception processing, complex processing, and device independence. Scoring guidelines are as follows: 0 No sensitive control, logical processing, mathematical processing, exception processing, or complex processing. 1 Any one of sensitive control, logical processing, mathematical processing, exception processing, or complex processing. 2 Any two of sensitive control, logical processing, mathematical processing, exception processing, or complex processing. 3 Any three of sensitive control, logical processing, mathematical processing, exception processing, or complex processing. 4 Any four of sensitive control, logical processing, mathematical processing, exception processing, or complex processing. 5 All five of sensitive control, logical processing, mathematical processing, exception processing, or complex

		processing. This characteristic has a normal (bell-shaped) distribution, centered about the average score of 3. There are few scores of 0 or 5. Scores for complex processing are highest on mainframes and lowest on microcomputers; highest in 3GL projects and lowest in 4GL projects. The scores are higher, and have a flatter distribution, for new developments than for enhancement projects. Processing complexity increases steadily from 1991 through to 1996.
Task Manager	1.2.3.2	**Rate Reusability Complexity:** Was the application developed to meet one or many user needs? Reusability describes the degree to which the application and the code in the application have been specifically designed, developed, and supported to be usable in other applications. Scoring guidelines are as follows: 0 No reusable code. 1 Reusable code is used within the application. 2 Less than 10% of the application considered more than one user's needs. 3 Ten percent (10%) or more of the application considered more than one user's needs. 4 The application was specifically packaged and/or documented to ease re-use, and the application is customized by the user at source code level. 5 The application was specifically packaged and/or documented to ease re-use, and the application is customized for use by means of user parameter maintenance. The importance of reusability is generally low, with 59% below average and only 14% above average, but it is a very mixed bag. Reusability is slightly more of a concern for decision support systems than for other types of software. It is less of a concern on personal computers than on mainframes.
Task Manager	1.2.3.3	**Rate Installation Ease Complexity:** How difficult is conversion and installation?

		Installation ease describes the degree to which conversion from previous environments influenced the development of the application. Conversion and installation ease are characteristics of the application. A conversion and installation plan and/or conversion tools were provided and tested during the system test phase. Scoring guidelines are as follows: 0 No special considerations were stated by the user, and no special setup is required for installation. 1 No special considerations were stated by the user but special setup is required for installation. 2 Conversion and installation requirements were stated by the user, and conversion and installation guides were provided and tested. The impact of conversion on the project is not considered to be important. 3 Conversion and installation requirements were stated by the user, and conversion and installation guides were provided and tested. The impact of conversion on the project is considered to be important. 4 In addition to 2 above, automated conversion and installation tools were provided and tested. 5 In addition to 3 above, automated conversion and installation tools were provided and tested. This characteristic has one of the broadest distributions. Scores are low overall (54% below average, 22% above average), but both extremes are well represented. Ease of installation is of no concern in 20% of projects, and extremely important in 15%. Scores are higher for enhancement projects than for new developments; higher for mainframe projects than for other platforms; and higher for engineering systems than for other business areas.
Task Manager	1.2.3.4	**Rate Operational Ease Complexity:** How effective/automated are startup, backup, and recovery? Operational ease describes the degree to which the application attends to operational aspects, such as start-up, back-up, and recovery processes. Operational ease is a characteristic of the application. The application minimizes the need for manual activities, such as tape mounts, paper handling, and direct on-location manual intervention.

		Scoring guidelines are as follows: 0 No special operational considerations other than the normal back-up procedures were stated by the user. 1 Any one of manual start-up, back-up, and recovery; automatic start-up, back-up, and recovery; tape mount minimization; or paper handling minimization. 2 Any two of manual start-up, back-up, and recovery; automatic start-up, back-up, and recovery; tape mount minimization; or paper handling minimization. 3 Any three of manual start-up, back-up, and recovery; automatic start-up, back-up, and recovery; tape mount minimization; or paper handling minimization. 4 Any four of manual start-up, back-up, and recovery; automatic start-up, back-up, and recovery; tape mount minimization; or paper handling minimization. 5 The application is designed for unattended operation. Unattended operation means no operator intervention is required to operate the system other than to start up or shut down the application. Automatic error recovery is a feature of the application. Ease of operation is rarely much of an issue. Scores for this characteristic are nearly the lowest overall: 62% are below average, and only 14% are above average. Scores are spread from 0-3, with 2 the most common score. The only significant difference emerges when projects are grouped by application type: scores are higher for management information systems and decision support systems than for transaction/ production systems and office information systems.
Task Manager	1.2.3.5	**Rate Multiple Sites Complexity:** Was the application designed for multiple sites/organizations? Multiple sites describes the degree to which the application has been developed for multiple locations and user organizations. The application has been specifically designed, developed, and supported to be installed at multiple sites for multiple organizations. Scoring guidelines are as follows: 0 User requirements do not require considering the needs of more than one user/installation site.

		1 Needs of multiple sites were considered in the design, and the application is designed to operate only under identical hardware and software environments. 2 Needs of multiple sites were considered in the design, and the application is designed to operate only under similar hardware and/or software environments. 3 Needs of multiple sites were considered in the design, and the application is designed to operate under different hardware and/or software environments. 4 Documentation and support plan are provided and tested to support the application at multiple sites and the application is as described by 1 or 2. 5 Documentation and support plan are provided and tested to support the application at multiple sites and the application is as described by 3. This characteristic has the lowest scores of all: 68% are below average, and 33% have the minimum possible score of 0. Scores are very low for legal systems, and high for engineering systems. They are higher for new developments than for enhancements or redevelopments; higher for 3GL projects than for others; and higher on midrange computers than on mainframes. Once again, scores are higher for management information systems and decision support systems than for transaction/production systems and office information systems.
Task Manager	1.2.3.6	**Rate Facilitate Change Complexity:** Was the application designed to facilitate change? Facilitate change describes the degree to which the application has been developed for easy modification of processing logic or data structure. The following characteristics can apply for the application: flexible query and report facility to handle requests (e.g., simple, average, and complex) and business control data is kept in tables that are updated the same or the next day. Scoring guidelines are as follows: 0 None of the above. 1 Any one of simple, average, or complex querying and reporting, or immediate or next-day business control data maintenance. 2 Any two of simple, average, or complex querying and

		reporting, or immediate or next-day business control data maintenance. 3 Any three of simple, average, or complex querying and reporting, or immediate or next-day business control data maintenance. 4 Any four of simple, average, or complex querying and reporting, or immediate or next-day business control data maintenance. 5 All five of simple, average, or complex querying and reporting, or immediate or next-day business control data maintenance. Each score for this characteristic is well represented, but generally scores are low: 53% are below average, and 20% above average. The distribution is bimodal, with the two common scores being 0 (of no concern) and 3 (average concern). Scores are low for 3GL projects, and high for 4GL projects. They are low for new developments; low for mainframe projects; and high for engineering projects. Not surprisingly, this characteristic is more important for management information systems and decision support systems, and less important for transaction/production systems.
Task Manager	1.2.3.7	**Determine Application Complexity:** Add the six weighted scores together for application complexity.
Task Manager	1.2.4	**Determine Value Adjustment Factor Score:** Add the system complexity, input and output complexity, and application complexity factors together for the value adjustment factor Score.
Task Manager	1.3	**Determine Function Points:** Determining function points consists of factoring unadjusted function points and value adjustment factor together.
Task Manager	1.3.1	**Determine Complexity Factor:** Multiply the value adjustment factor with 0.01 and add 0.65 for the complexity factor.
Task Manager	1.3.2	**Determine Function Points:** Add the unadjusted function points with the complexity factor for function points.

FUNCTION POINTS ESTIMATION FORM

1.1 Determine Unadjusted Function Point Count

	Measurement Parameter	Count		Weighting Factor Low	Average	High		Total
1.	External Inputs		x	3	4	6	=	
2.	External Outputs		x	4	5	7	=	
3.	External Inquiries		x	3	4	6	=	
4.	Internal Logical Files		x	7	10	15	=	
5.	External Interface Files		x	5	7	10	=	
	Unadjusted Function Point Total →							

1.2 Determine Value Adjustment Factor

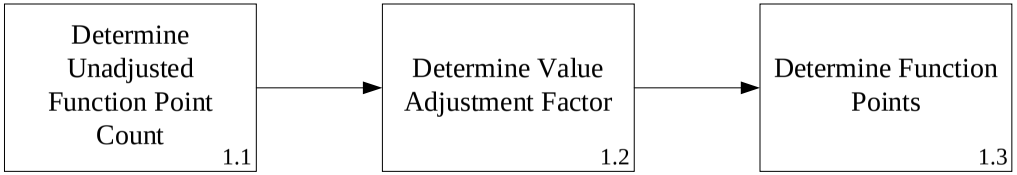
Rate Each Factor: (0 - No Influence, 1 - Incidental, 2 - Moderate, 3 - Average, 4 - Significant, 5 - Essential)

1.	How many data communication facilities are there?	
2.	How are distributed data and processing functions handled?	
3.	Was response time or throughput required by the user?	
4.	How heavily used is the current hardware platform?	
5.	How frequently are transactions executed?	
6.	What percentage of the information is entered online?	
7.	Was the application designed for end-user efficiency?	
8.	How many internal logical files are updated by on-line transaction?	
9.	Does the application have extensive logical or math processing?	
10.	Was the application developed to meet one or many user needs?	
11.	How difficult is conversion and installation?	
12.	How effective/automated are startup, backup, and recovery?	
13.	Was the application designed for multiple sites/organizations?	
14.	Was the application designed to facilitate change?	
	Value Adjustment Factor →	

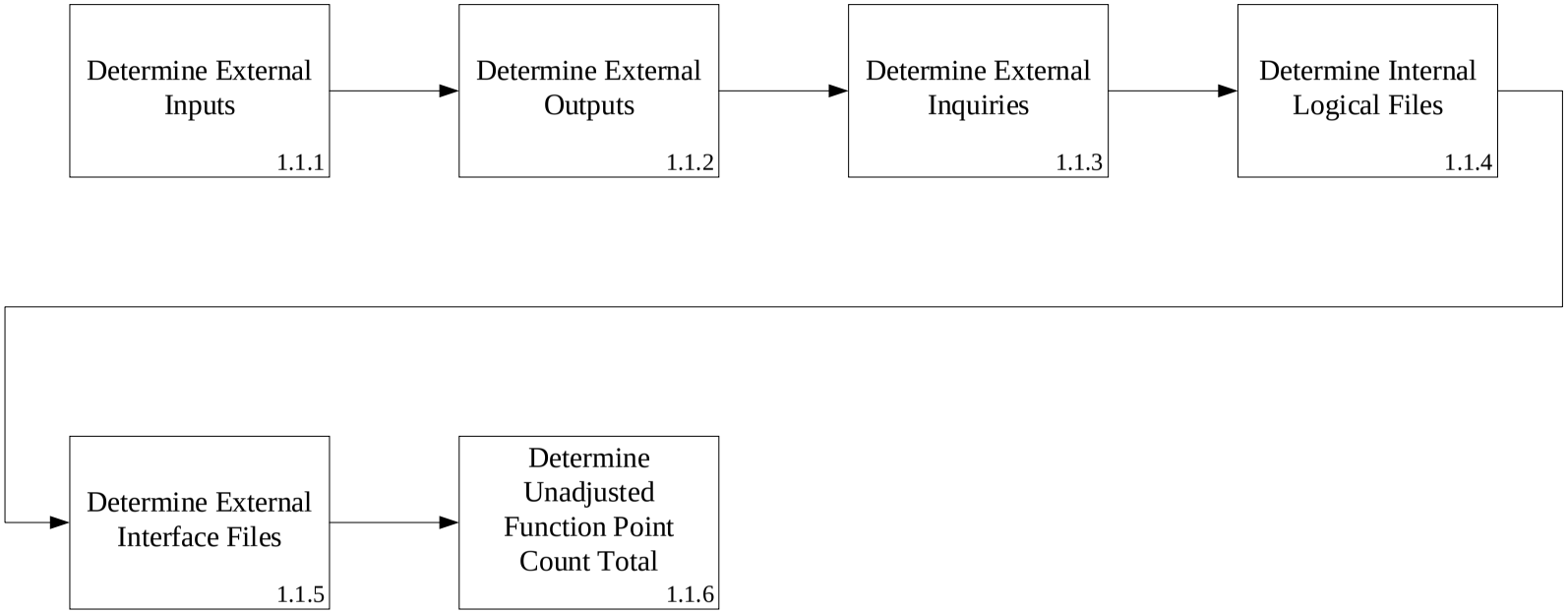
1.3 Determine Function Points

Unadjusted Function Points x (0.65 + 0.01 x Value Adjustment Factor) →

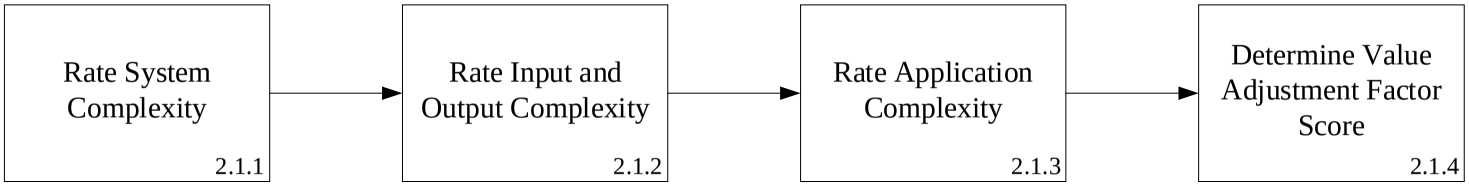
Calculate Function Points Workflow



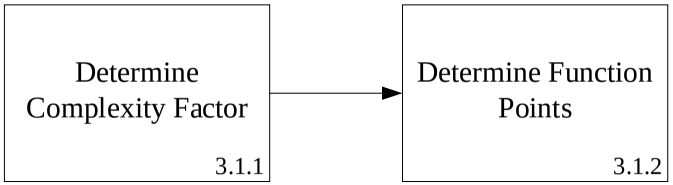
Determine Unadjusted Function Point Count Workflow



Determine Value Adjustment Factor Workflow



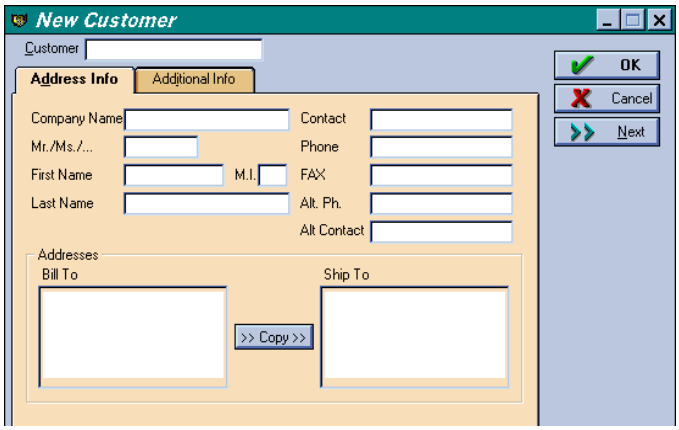
Determine Function Points Workflow



Example

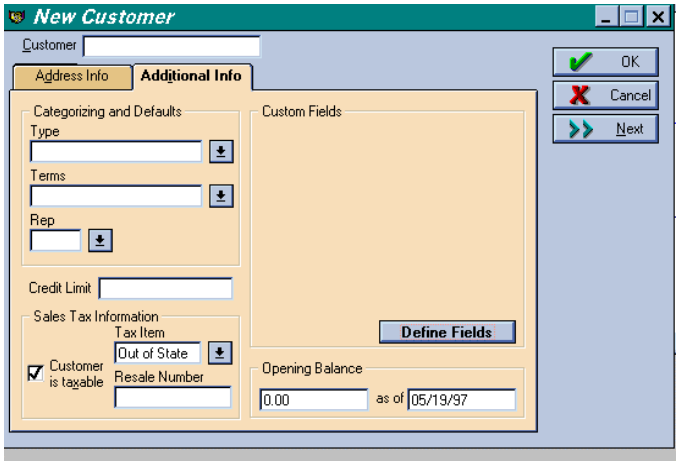
1. APPLICATION

This example is a simple desktop application for entering customer data. It is composed of a single dialogue box with two tabs and multiple input fields, buttons, check boxes, and pull down menu icons.



It is a standalone desktop application with no communication requirements. It should be reasonably interactive and promptly respond to user input and commands. It is an average desktop application on a moderately utilized PC that will be used several times a week. All information will be manually entered by the user and should be reasonably user friendly.

It will involve no complex processing or algorithms and will only be used by a single departmental administrator. It will install using a fast and easy to use wizard and the data should be backed up and recovered in a few seconds at the user's command. It will be located in one facility and rarely be enhanced.



2. DETERMINING UNADJUSTED FUNCTION POINT COUNT

Determining the unadjusted function point count consists of counting the number of external inputs, external outputs, external inquiries, internal logical files, and external interface files.

- ***External Inputs***: There is one Customer field and 3 macro-level buttons on the right for a total of 4 inputs. The Address Info tab consists of 12 input fields, a button, and the tab itself for a total of 14 inputs. The Additional Info tab consists of 8 input fields, 4 pull down menu icons, a checkbox, a button, and the tab itself for a total of 15 inputs. So, there are 4 macro-level inputs, 14 Address Info tab inputs, and 15 Additional Info tab inputs, for a total of 33 External Inputs. The External Inputs will be rated Low, therefore multiplied by 3, for a weighted External Input total of 99.

- ***External Outputs***: There are 3 External Outputs, the Address Info tab, the Additional Info tab, and the checkbox. The External Outputs will be rated Low, therefore multiplied by 4, for a weighted External Output total of 12.

- ***External Inquiries***: There are no External Inquiries for this application.

- ***Internal Logical Files***: There are 2 Internal Logical Files, the Address Info tab or record, and the Additional Info tab or record. The Internal Logical Files will be rated Low, therefore multiplied by 7, for a weighted Internal Logical Files of 14.

- ***External Interface Files***: There are no External Interface Files for this application.

The Unadjusted Function Point Total is therefore is 125, which is the sum of 99 External Inputs, 12 External Outputs, and 14 Internal Logical Files.

3. DETERMINING VALUE ADJUSTMENT FACTOR

Determining value adjustment factor consists of rating data communication, distributed data processing, performance, and heavily used configuration complexity.

- ***Data Communications***: There are no communication requirements. Therefore, Data Communications will be rated at 0.

- ***Distributed Data Processing***: This is a standalone desktop application. Therefore, Distributed Data Processing will be rated at 0.

- ***Performance***: It should be reasonably interactive and promptly respond to user input and commands. Therefore, Performance will be rated at 1.

- ***Heavily Used Configuration***: It is an average application on a moderately used desktop PC. Therefore, Heavily Used Configuration will be rated at 0.

- ***Transaction Rate***: The application will be used several times a week. Therefore Transaction Rate will be rated at 2.

- ***On-Line Data Entry***: All information will be manually entered. Therefore, On-Line Data Entry will be rated at 5.

- ***End-User Efficiency***: The application should be reasonably user friendly. Therefore, End-User Efficiency will be rated at 3.

- ***On-Line Update***: All information will be manually entered. Therefore, On-Line Update will be rated at 0.

- ***Complex Processing***: The application will involve no complex processing or algorithms. Therefore, Complex Processing will be rated at 0.

- ***Reusability***: The application will be used by a single department administrator and seldomly enhanced. Therefore, Reusability will be rated at 0.

- ***Installation Ease***: The application requires the development of a fast and easy to use wizard. Therefore, Installation Ease will be rated at 2.

- ***Operational Ease***: Data should be backed up and recovered at the user's command. Therefore, Operational Ease will be rated at 5.

- ***Multiple Sites***: The application will be located in one facility. Therefore, Multiple Sites will be rated at 0.

- ***Facilitate Change***: The application will rarely be enhanced. Therefore, Facilitate Change will be rated at 0.

The Value Adjustment Factor is therefore is 18, which is the sum of a Performance of 1, Transaction Rate of 2, On-Line Data Entry of 5, End-User Efficiency of 3, Installation Ease of 2, and Operational Ease of 5.

4. DETERMINING FUNCTION POINTS

- ***Function Points***: The number of Function Points are 104, which is the product of a Value Adjustment Factor of 18 multiplied by 0.01, added to 0.65, and multiplied by 125 Unadjusted Function Points (the total was rounded up from 103.75).