

FPA according to Nesma and IFPUG

Nesma FPA and IFPUG FPA are two leading Functional Size Measurement (FSM) methods, compliant with ISO/IEC 14143-1. This paper describes the differences. Sections 1 – 3 deal with the detailed FPA method, section 4 with the high level and indicative FPA methods.

1. History

Before 1990 there were major differences between the Nesma FPA and IFPUG FPA methods. From 1990 on Nesma and IFPUG have worked closely together to define a common standard. Between 1990 and 2004 the IFPUG FPA guidelines have gone through several revisions. Practically, this has resulted in a continually lower function point count for the same functional sample, using a newer version of the IFPUG CPM. In 2004 IFPUG CPM 4.2 was released. Since then the IFPUG FPA guidelines are stable. The Nesma FPA guidelines have been stable since version 1.0 of the manual was published in 1989.

As of IFPUG CPM 4.2 (2004) there are almost no differences between Nesma and IFPUG. The few that still exist are described below. The impact of these differences is very small.

Detailed background information about which guideline was changed when in the years before 2004 can be found in the document "[Historical Review Nesma and IFPUG](#)" (in Dutch only).

2. Actual versions of the Nesma and IFPUG CPM

Both Nesma and IFPUG maintain their own FPA Counting Practices Manual (CPM).

- Nesma CPM 2.1 (2003) Nesma Counting Practices Manual FPA, release.2.1 (English)
- Nesma CPM 2.2 (2004) Nesma Counting Practices Manual FPA, release.2.2 (Dutch)
- IFPUG CPM 4.3 (2010) IFPUG Counting Practices Manual FPA, release 4.3

Nesma CPM 2.1 is the version that was certified by ISO as ISO/IEC 24570.

Nesma CPM 2.2 is an exact Dutch translation of Nesma CPM 2.1. However, some cosmetic modernizations were made in the examples (for example, GUI based screens instead of text based). To avoid confusion, a new release number was assigned (2.2). However, the FPA rules and guidelines are identical. So Nesma CPM 2.1 and Nesma CPM 2.2 are interchangeable.

3. Comparison of the Nesma and IFPUG FPA guidelines

The Nesma and IFPUG FPA guidelines are almost the same. This section shows the differences.

Nesma and IFPUG use the same terminology, have the same criteria for identifying elementary processes and data functions, and differentiate the same five types of user functions: Internal Logical File (ILF), External Interface File (EIF), External Input (EI), External Output (EO), External Inquiry (EQ). In Dutch: ILGV, KGV, IF, UF, OF.

The rules for determining the type and complexity of a function are the same, with a few exceptions:

- a) External Inquiry vs External Output
- b) Complexity of an External Inquiry
- c) Implicit Inquiry
- d) Code data (Code tables)
- e) Physical media
- f) Queries with multiple selections ("and/or" situations)

In the following each topic is highlighted.

a) External Inquiry vs External Output

For IFPUG, an External Inquiry is defined as a function that presents data to a user from a logical file (ILF or EIF) without undergoing additional processing (such as calculations, updates to an ILF, etc.). In all other cases it is considered an External Output.

For Nesma, the same rules apply, but in addition, a unique selection key must have been entered and the output must be fixed in scope. In some cases, therefore, IFPUG will count an External Inquiry, while Nesma counts the same function as an External Output (e.g., "Show all customers").

The impact of this difference is *marginal* for the number of function points for a system or project because only the type of function (External Inquiry or External Output) is affected, not the number of counted functions.

b) Complexity of an External Inquiry

For Nesma the functional complexity of the input part of an External Inquiry is based on the complexity rules for an External Input function; the complexity of the output part is based on the rules for an External Output function. The more complex of the two will be used as the complexity of the External Inquiry.

For IFPUG the functional complexity is determined in the same way as for all other transactions, by counting the number of data-element-types crossing the application boundary and the number of involved data-functions.

In practice the impact of this difference is *marginal* for the number of function points for a system or project.

c) Implicit Inquiry

When modifying or deleting data, the data is often first presented to the user for viewing. This is known as an "implicit inquiry".

For Nesma, the underlying goal of a function is always the primary objective. Nesma therefore does not consider the implicit inquiry as a separate transactional function, but as an integral part of the modify- or delete function. The data element types presented to the user by the implicit inquiry are therefore added to those counted in the modify- or delete function. Nesma will only count the External Inquiry if it is specifically identified by the user for the purpose of querying data.

IFPUG does not have specific rules for this situation. Some IFPUG FPA-analysts will therefore count this as a separate External Inquiry function (if counted nowhere else), others not.

Usually the user will have defined this function as an (explicit) inquiry (and it will thus be counted). The implicit inquiry will then not be counted (again), because the same function cannot be counted twice. That's why the impact of this difference is *marginal* for the number of function points for a system or project.

d) Code tables

In general, entities can be seen as being composed of primary data ("business objects") or composed of secondary data (supportive data).

For the primary data, both Nesma and IFPUG follow the same guidelines.

Secondary data usually consist of code tables, also called “FPA-tables” by Nesma. As an example, consider the “translation table”: article code, article description.

Nesma will classify all code tables together as one ILF and/or one EIF. The number of record element types will be set equal to the number of identified code tables. Altogether, the FPA-tables-ILF will also count for one External Input, one External Output and one External Inquiry. For the FPA-tables-EIF no transactional functions are counted, even though External Outputs or External Inquiries may be present.

IFPUG considers code tables to be an implementation of technical or quality requirements for the user, and not part of the functional user requirements. That’s why IFPUG states, that in accordance with the ISO FSM standard, code tables and the transactional functions associated with them, are not to be counted using function points.

The impact of this difference is *marginal* for the number of function points for a system or project. The difference will be *at most* 25 function points for an FPA-tables-ILF, and 20 function points for an FPA-tables-EIF.

e) Physical media

Physical media is ignored in Nesma FPA. Nesma looks at the underlying functionality and explicitly states, that the same elementary process using different physical media, should be counted only once.

If the number of data element types and the logical processing are the same, input entered through different media will be counted as one External Input by Nesma. The same holds true for External Outputs and External Inquiries. Reports that can be presented on different media (print, screen, etc.) are counted as one External Output or External Inquiry (when the number of data element types and the logical processing remain the same).

IFPUG does not provide specific guidelines for this situation.

f) Inquiries containing multiple selection criteria (“and/or” situations)

Nesma states, that only mutually exclusive selections are to be counted.

IFPUG has no specific guidelines for this situation. Some IFPUG FPA analysts therefore, count every conceivable combination of selection criteria as separate functions, which may result in large differences in function points among IFPUG FPA analysts.

4. Early and Fast Function Point Analysis

To carry out a detailed FPA one needs detailed user requirements, and also detailed FPA guidelines on how to assess these detailed requirements.

Both Nesma and IFPUG recognize and promote the same two FPA methods that can be applied early in the software development life cycle or can be used to do a function point analysis fast:

- High Level FPA method (high accuracy, abstracting from complexity of functions)
- Indicative FPA method (rough order of magnitude of the functional size)

Nesma : [Early Function Point Analysis](#)

IFPUG : [Early Function Point Analysis and Consistent Cost Estimating](#) (uTip 3)