INTERNATIONAL STANDARD

ISO/IEC 19761

Second edition
2011-03-15

Software engineering — COSMIC: a functional size measurement method

Ingénierie du logiciel — COSMIC: une méthode fonctionnelle de mesure de taille

Reference number
ISO/IEC 19761:2011(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 19761 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

This second edition cancels and replaces the first edition (ISO/IEC 19761:2003), which has been technically revised.

Introduction

Software is a major component of many corporate budgets. Organizations recognize the importance of controlling software expenses and analysing the performance of the budgets allocated to software development and maintenance in order to benchmark against the best in the field. To do so, measures and models using these measures are needed.

Measures are needed for analysing both the quality and the productivity associated with developing and maintaining software. On the one hand, technical measures are needed to quantify the technical performance of products or services from a developer's viewpoint. Technical measures can be used for efficiency analysis; to improve the performance of designs, for instance.

On the other hand, functional measures are needed to quantify the performance of products or services from a user's or owner's perspective; for productivity analysis, for instance. Functional measures must be independent of technical development and implementation decisions. They can then be used to compare the productivity of different techniques and technologies.

The Full Function Points (FFP) method was proposed in 1997 with the aim of offering a functional size measure specifically adapted to real-time software. Field test results, coupled with the feedback received from organizations which used it, motivated the authors to improve the method. Many improvements were also inspired by the work of the Common Software Measurement International Consortium (COSMIC). The results of these efforts were published in May 2001 as version 2.1 of the COSMIC-FFP Functional Size Measurement Method (as it was first named), aiming to be applicable to business application, real-time and system software [4].

ISO/IEC 19761:2003 was based on this version 2.1 of the COSMIC-FFP Functional Size Measurement method.

Extensive experience of using the method convinced the Common Software Measurement International Consortium of the need for various clarifications and improvements to the description of the method. Version 3.0 of the method was therefore published in December 2007 [5]. These various changes have not altered the underlying model for the measurement of software functional size since it was first published in 2001.

The International Standard aims to meet the needs of

a) software suppliers facing the task of translating customer requirements into the functional size of software to be produced as a key activity in their project cost estimating,

b) customers who want to know the functional size of delivered software as an important component of measuring supplier performance.

With version 3.0 of the method, the name of the method was simplified from “COSMIC-FFP” to “COSMIC”. The name of the unit of measure has also been simplified from “Cfsu” (COSMIC functional size unit) to “CFP” (COSMIC Function Point).

Software engineering — COSMIC: a functional size measurement method

1 Scope

This International Standard specifies the set of definitions, conventions and activities of the COSMIC Functional Size Measurement Method. It is applicable to software from the following functional domains:

a) application software;

EXAMPLE Banking, insurance, accounting, personnel, purchasing, distribution or manufacturing.

b) real-time software;

EXAMPLE Software for telephone exchanges and message switching, software embedded in devices to control machines such as domestic appliances, lifts and car engines, for process control and automatic data acquisition, and within the operating system of computers.

c) hybrids of the above.

EXAMPLE Real-time reservation systems for airlines or hotels.

This International Standard has not been designed for measuring the functional size of a piece of software, or its parts, which

— is characterized by complex mathematical algorithms or other specialized and complex rules, such as can be found in expert systems, simulation software, self-learning software and weather forecasting systems, or

— processes continuous variables such as audio sounds or video images, such as can be found in computer game software, musical instruments and the like.