MK II和COSMIC-FFP方法的比较分析

黄祖旭 2022141461121

1. **方法概述与主要区别**

|  |  |  |
| --- | --- | --- |
| **方法** | **核心概念** | **适用场景** |
| **COSMIC-FFP** | 基于数据移动（输入、输出、读、写）进行功能点估算，关注用户与系统的交互。每个操作有固定的权重。 | 通过事务复杂度和数据群的数量、数据元素数量来评估功能点，考虑事务的复杂性和系统内外的数据存储。 |
| **MK II** | 通过事务复杂度和数据群的数量、数据元素数量来评估功能点，考虑事务的复杂性和系统内外的数据存储。 | 适用于复杂系统，特别是面向对象的系统，能够处理复杂的事务与数据操作。 |

1. **功能点估算维度**
2. 数据类型与评估方式

* COSMIC-FFP
  + 基于四种数据操作（输入、输出、读、写）的数量来评估功能点。
  + 通过计算每个模块的数据操作和它们的权重，估算出系统的总功能点。
  + 主要考虑系统与外部的交互，因此适用于对交互性要求较高的应用系统。
* MK II：
  + 基于功能类型和复杂度的评估，结合数据群（如数据库）和事务的复杂度来计算功能点。
  + 评估包括事务的复杂度、数据元素（DET）的数量和数据群（RET）的数量。
  + 通过这些因素的组合，更加精细地反映了系统的复杂性，尤其适合面向对象系统的开发。

1. 比较分析

* COSMIC-FFP关注的是数据操作的类型和数量，简单明了，易于实施。但它忽略了系统内部数据存储和事务的复杂性，可能在面对大型复杂系统时低估了工作量。
* MK II更注重事务复杂度和数据元素的数量，提供了更加详细的分析，并能够反映数据的存储和事务操作的复杂性。因此，对于涉及多层数据交互和复杂操作的系统，MK II能提供更精确的工作量估算。

1. **功能点权重与复杂度分类**

* COSMIC-FFP
  + 只考虑数据移动操作（E、X、R、W）来评估功能点，并且每个数据操作都有固定的权重。例如，每个输入（E）操作权重通常为1，每个输出（X）操作也为1。
  + 功能点的计算较为直接，适合快速估算。
* MK II：
  + 事务和数据群的复杂度根据其对系统操作的影响，分为简单、中等、复杂三个等级。
  + 事务复杂度影响功能点的分配，数据群（RET）和数据元素（DET）数量也直接影响功能点估算的结果。
  + MK II方法的权重分配更加灵活和精细，能够更好地反映不同模块和功能的工作量差异。
* 比较分析：
  + COSMIC-FFP：采用固定权重的简化方法，快速且易于理解，适用于项目需求较简单、开发复杂度较低的场景。
  + MK II：通过对事务复杂度和数据元素的细化分析，为每个模块赋予动态权重，提供更为准确的工作量估算，尤其适用于复杂系统、面向对象系统和数据库操作密集的应用。

1. **工作量估算差异**

* COSMIC-FFP
  + 计算简单，侧重于输入输出交互的功能点总和，不考虑功能复杂性，工作量估算通常较为保守，偏向较快的开发周期。
  + 由于缺乏复杂度考量，可能低估需要处理的大量数据或复杂事务。
* MK II：
  + 更精细化的复杂度评估（通过事务和数据元素），能够准确捕捉系统复杂度，通常给出较高的功能点估算，从而提供较长的开发周期。
  + 对复杂度较高的系统，估算出的工作量更为准确，反映了系统在数据管理、事务处理方面的真实复杂度。
* 比较分析：
  + COSMIC-FFP：给出的功能点较少，工作量估算较轻，适合于对功能需求简单的系统。
  + MK II：给出的功能点较多，反映了系统更高的复杂度和较长的开发周期，适合功能复杂、涉及大量数据操作的系统。

1. **总结**

* COSMIC-FFP
  + 优点：快速、简单、易于实施，适合于需求明确、功能单一的系统。
  + 缺点：对于复杂系统，可能低估工作量，特别是在涉及大量数据存储和复杂事务时。
* MK II：
  + 优点：细致、准确，能够应对复杂系统的工作量估算，适用于大规模、多层次的数据交互和事务系统。
  + 缺点：估算过程较复杂，实施时间较长，适合于开发团队经验丰富且项目复杂的场景。

1. **对比总结**

|  |  |  |
| --- | --- | --- |
| **特点** | **COSMIC-FFP** | **MK II** |
| **适用场景** | 适合功能较简单的业务应用或Web应用系统 | 适合复杂的面向对象系统、大型企业级应用、数据库密集型应用 |
| **功能点计算方式** | 计算简单、快速，主要依据数据操作（输入、输出、读、写） | 通过评估事务复杂度和数据元素，提供更细致的功能点估算 |
| **权重分配** | 固定权重 | 根据事务复杂度和数据群的数量动态分配权重 |
| **工作量估算** | 估算较为轻松，适用于简单项目 | 估算较为准确，适合复杂系统，通常需要更长的开发周期 |
| **精度** | 对简单系统较为准确，但对复杂系统可能存在低估 | 更适合复杂系统，能够准确估算数据处理和事务复杂度的工作量 |
| **实施复杂度** | 简单、快速，易于实现 | 实施较为复杂，需要较详细的数据和复杂度评估 |

1. **总结**

* COSMIC-FFP适合用于快速、简单的系统，尤其是那些功能需求明确且交互操作较少的项目。其简化的计算方式让它能够快速得到结果，但在面对复杂系统时可能会低估实际工作量。
* MK II则更为精确，能够细致分析系统复杂度，尤其适合那些功能复杂、涉及大量数据交互的系统。虽然其估算过程较为繁琐，但能提供更准确的工作量估算。

选择哪种方法取决于项目的复杂度、开发周期要求以及团队的经验。如果项目较为复杂，MK II能够提供更可靠的工作量估算；如果项目较为简单，且对估算时间要求较高，COSMIC-FFP是一个更高效的选择。