Association for Information Systems
AIS Electronic Library (AISeL)

ICIS 1996 Proceedings

International Conference on Information Systems (ICIS)

December 1996

# Estimating Size for Object-Oriented Software

Arlene Minkiewicz
*Lockheed Martin PRICE Systems*

Bruce Fad
*Lockheed Martin PRICE Systems*

# ESTIMATING SIZE FOR OBJECT-ORIENTED SOFTWARE

**Arlene F. Minkiewicz**
**Bruce Fad**
Lockheed Martin PRICE Systems

In recent years, Object Oriented (OO) technologies have emerged as a dominant software engineering practice. As happens with many new technologies, the growth of OO practices has required software developers and managers to rethink the way they have been estimating the size of their development projects. Traditional software measurement techniques have proven unsatisfactory. The Source Lines of Code (SLOC) metric and the Function Point metric were both conceived in a era when programming required dividing the solution space into data and procedures. This notion conflicts with the object-oriented paradigm. Traditional design techniques separate data and procedures while object-oriented designs combine them. There is no accounting in a SLOC or Function Point count for the fact that some of the behaviors of an object are inherited by objects derived from that object. What is needed is a size metric that incorporates an understanding of the key OO concepts, is easy to estimate in the early phases of the problem analysis, is countable in the finished product, and can be counted by any knowledgeable person — whether they be software analyst or end-user of the software being measured. The metric also needs to show a strong correlation to development effort in order to be useful as an effort predictor or productivity measure.

In this paper, a metric is proposed, Predictive Object Points (POPs), that has the potential to do all of the above. Unlike traditional measures, POPs are based on an object oriented paradigm, encapsulating object behavior and the interaction between objects. The POPs measure combines several contemporary metrics to establish an overall measure suitable for predicting effort and/or tracking productivity. The metric at the heart of the POP calculation is Weighted Methods per Class (WMC). WMC looks at each top level class (or each distinct object from the user's perspective) and assigns a weight to the behaviors of that class that are seen by the world. The "weight" of an object's behavior is determined by evaluating the effects that the behavior has on the objects in the system (by counting the properties that this behavior impacts) and the amount of control the objects in the system have over this behavior (by counting the parameters of the method or the pieces of information that get passed to it). The calculated WMC metric is combined with information about the groupings of objects into classes and the relationships between these classes of objects to arrive at a value which appears to correlate to the effort associated with implementing a solution. Because the metric is based on behavior, it mimics much of what is useful about a Function Point measure. Since behavior is something widely understood, a POP value can be determined by someone with only a little knowledge of implementation details. And, since the things that impact behavior "weight" are well understood, detailed counting rules can be established for the POP metric.