ON COMPATIBILITY – A PROPOSED SOLUTION

GÖRAN FRIES

Abstract.

The paper gives a definition of compatibility, and tries to answer the question why compatibility is of great importance. The paper also suggests a solution by using multilingual techniques in the creation of the complete system.

1. Problem definition.

A definition of compatibility and portability will first be presented by means of the following three criteria.

1) Ability to move high level language programs from one computer to another, possibly by means of macro processors or preprocessors.
2) Ability to run the same card deck on different computers except accounting information.
3) In addition to the second criterion: The property that output depends only on input data and program, and is independent of computer.

If the first criterion is satisfied we have portability, while the other two criteria imply compatibility of different degrees, from weak (2) to full (3).

The need of some kind of compatibility depends on a desire to solve some of the following problems:

1) Avoiding expensive reprogramming, if a change of computer takes place.
2) Exchanging software between different users.
3) Using another computing center in case of overloading or breakdown.
4) Using a (big) computer for production runs and another for test runs. In this case only full compatibility will guarantee the same results, including errors, on the two computers.
5) Using different computers in a computer network.

To solve the first problem, at least portability is needed while the other problems demand increasing degrees of compatibility.

Received October 10, 1975.
Today there exist some portable systems, but hardly any really compatible system. The only attempt at solving the compatibility problem seems to take place in Moscow, where the ALMO language is being developed. The ALMO language satisfies the second criterion. Also UNCOL was intended to solve the compatibility problem, but was a complete failure.

UNCOL was a proposed solution to solve the problem of programming communication with changing machines, first published in CACM August 1958. OBM is a system where it is possible to write a program regardless of computer, and then run it on any computer. The OBM system which will be described here will have an outer look similar to UNCOL, but OBM is related not only to the language level but rather to the standardized machine level.

A compatible system is considered to be inefficient and expensive to use, but this is only partly correct. If it is possible to choose the degree of compatibility, from weak to full, we may control the cost vs. efficiency. The computers will also become faster and the loss in efficiency will be less expensive than the increase in manual work without access to compatibility.

2. General design.

It seems natural to design a compatible system with an outer look similar to the UNCOL model. A really useful system should contain both compilers and an operating system. Some features of the system are explained in Fig. 1.

![Diagram of OBM system]

The system may be divided into four parts.

1) Compilers from high level languages to OBM.