The paper presents an engineering model for developing socio-technical systems such as office applications. This model is composed of a methodology for analysis of the office work based in the structure of their components, the Tasks, Activities and Actions Methodology (T.A.& A.). A view mechanism on objects of the database is employed to represent user transactions. The relationship between the concepts of T.A.& A. methodology, objects and database views on objects are described.

1. INTRODUCTION

The modeling of the reality into an conceptual model for the development of conventional data processing systems starts when the designer knows what part of the system will be materialized as automatic procedures. This means that the decision on what segment of the office work will be left to the human activities and what will be automated is previous to the software engineering specification process.

In conventional systems, as an inventory system, the starting point for the development is a decision taken by the organization and methods group on what parcel of the system will be automated. E.g., the sales would correspond to a manual update of the database, or it would be associated with the invoice printout and the data on the product might be obtained via an optical code bar scanner. This kind of modeling is a partial, and often very partial, modeling of the reality.

Office systems are socio-technical systems [Voss 86] which integrate human actions and automated procedures. Continuing improvement in computers system allows us to build more complex and closely tied interactions between the automated and human...
parcels of office work as a consequence of new human interfaces and more complex processing. This kind of system needs a comprehensive design methodology. However, the traditional methodologies used in software engineering are dedicated to the description and the development of the automated procedures, not for the description of a complete information system including the automated and the human parts of work. If this traditional modeling is accepted a big amount of information will be lost during the process. Worse, some important decisions about the automation of some parcel of the activities are taken without a clear statement of why this has happened.

Why are the actual limits of information systems modeling is so restricted? One of the first answers is that this kind of system is so complex that we are not able to create appropriate models. Now arrives the next question: Can we improve the modeling methodologies? The response seems to be YES if we are able to use new requirements collection and analysis tools and additionally to apply formal verification methods to help to develop the correct representation of the reality.

The difficulties found in systems analysis methodologies in terms of passing from reality to a conceptual model are well-known. On one hand this model must be validated by the users and, on the other, the same model must be clear and non-ambiguous for the systems analysts. In this context, an environment for the development of Office Information Systems (OIS) must be able to support alternative and equivalent representations in view of the difficulty to support these two conflicting needs with one single model.

From the reality to a working and reasonably reliable system a collection of techniques, tools and formalisms is needed. In an engineering environment the use of formal specifications to validate the design is an essential characteristic. The real problem to get a good system is:

The reason why it is so hard to say what we want is that we do not really know what we want (at least not in a complex system, which is where the problems are). One way of discovering what we want is to draw up a formal specification, derive a program conforming to it, and run it. (…) it is necessary to run the programs, not to debug the programs, but to debug the specifications - M. H. van Emden, in [Denning 89].

The purpose of our research is to develop both a conceptual model and an environment for the analysis, design and implementation of systems including human tasks in offices as an integrated part of the computerized environment.

In section the relations between net modeling, action of the user and the automated system are described; in section a representation of office work derived from the concepts of hierarchical decomposition of Structured Analysis [DeMarco 78] is shown; in section the use of Pr-T nets as a tool to represent the dynamics of the systems is described; and finally in section the correspondence between the representation of office work and the database schema is derived. This integration generates a conceptual model able to represent a complete environment in which not only the computerized but also the human actions are described.

2. ENGINEERING OF AN OFFICE SYSTEM

The full engineering cycle of an office system consists of five phases:
- requirement analysis,