WER'04 Panel:

Requirements and Analysis where is the boundary if any?

Oscar Pastor
Universidad Politécnica de Valencia
DSIC; Camino de Vera s/n; 46022 Valencia, Spain
opastor@dsic.upv.es

Even if it seems to be at a first glance a simple question, this is really a very interesting one. For many years, Analysis has been the generic term to refer to those initial Software Production Process steps where design is not yet taken into account. In some way, the analysis phase has been considered as the natural step where Conceptual Schemas of an Information System are conceived and represented.

In this context, a lot of software production processes are structured around two main questions, representing each of them one independent layer:

- "What" the system is? This is where analysis is conventionally located,. The main deliverable of this phase is the Conceptual Schema.
- "How" is it going to be represented in the selected software technology. In this case, we are just transforming the Conceptual Schema into its corresponding final software product.

But an important question is missing. How to assure that the representation done in the Conceptual Schema is the right one, the one that properly represent the User Requirements? Requirements Engineering is seen as a still earlier step, where even the software system can still not being considered as a system agent. As the Conceptual Schema itself must be the correct representation of the User Requirements, a new transformation process going from Requirements to Conceptual Schema is required. This could be expressed through a third question, to be added to the two previous ones:

- "Why" the system is this one? Why will it be represented by a given Conceptual Schema and not by another one?

This three-level structure is giving a precise answer to the initial question. There is a clear boundary between Requirements and Analysis, and this boundary is allowing us moving from the "why" to the "what". Precise modelling techniques to face this Requirements Modelling phase are required, as well as precise transformation techniques to go from the resulting requirements model, to the Conceptual Schema that will represent it properly.

Considering the emerging fashion of Model Transformation techniques, especially within the context of the MDA proposal, this view provides a concise strategy to be followed to elaborate sound and efficient Software Production Processes. Their aim will be to provide a clear path –as automated as possible- from requirements to the software product, through the intermediate role played by the Conceptual Schema. In this advanced context, Analysis is not anymore related only to an independent Conceptual Modeling step. It must be also linked to the source Requirements Modeling activities and their proper alignment with all the other relevant software production process artefacts.