Abstract. Goal-oriented Requirements Engineering (GORE) has con-
formed in the recent years a sound research working domain, where the
*i* ([1]) approach is being playing a leading role. The idea of starting any
software process with a well-founded goal modeling process is widely ac-
cepted, as it is essential for the quality of a software product to know and
to trace effectively why software services exist, and what goal(s) justify
and explain their existence.

With all this theoretical and practical background, it is the time to fo-
cus on an efficient GORE in Practice perspective. This is the main topic
of this keynote. By GORE in Practice it is meant to provide a Model-
driven development (MDD) approach based on a Model-driven architec-
ture (MDA) and any other accepted Model-driven engineering proposal
for defining a software production process- where a strict goal model-
ing step has to be the very first step of the process. Having *i* as a
reliable theoretical reference model, with also successful practical expe-
riences reported ([2]), this first step that corresponds to a Computation
Independent Model (CIM) in MDA terms, should not more attention. It
is just a matter of using *i*, and extending is use in practice.

But this goal-oriented model needs to be extended to elaborate a require-
ments model where not only goals, but i) system services that materialize
them and ii) data that those services require to occur, are properly spec-
ified. This extension can be done using a set of diverse Requirements
Engineering (RE) techniques, and in this keynote we will analyze in par-
ticular how a BPMN-based notation under a Communication Analysis
approach can be used. At the end of this CIM-based step, a full require-
ments models is available, where user goals and their associated system
services are precisely identified.

The next problem to be solved is again in MDA terms- the CIM to PIM
(for Platform-Independent Model) transformation. Now, how to face a
sound model transformation that takes an extended *i* model as the
input model in order to generate its associated executable conceptual
schema, becomes the challenge. Under the umbrella of the Conceptual
Programming notion ([3], the Conceptual Schema-based Software Devel-
opment approach ([4]), the Extreme Non-Programming proposals ([5])
and all this type of proposals, we will analyze in the keynote how current
tools that provide such a conceptual model compiler (i.e, Integranova
([6]), WebRatio ([7]), . . . ) can be used to provide the GORE in Practice capabilities introduced before.
Characterizing how to convert an extended goal-model -where the services that materialize any user goal are clearly specified- into an executable conceptual schema makes possible to ensure that any piece of code of the final application is the representation of a corresponding goal. This allows to assess traceability from goals to code, making goals really become the key notion of the proposed Requirements Model-based Software Process.

**Keywords:** Goal-Oriented Requirements Engineering, istar, Requirements Model-driven based Development, Conceptual Programming, From Goals to Code

**References**

7. [https://www.webratio.com/site/content/en/home](https://www.webratio.com/site/content/en/home)