

Guideline for developing a software life cycle process in natural language generation projects

M^a del Socorro Bernardos

Departamento de Sistemas Informáticos y Programación, Facultad de Informática,
Universidad Complutense de Madrid. C/ Juan del Rosal, 8, 28040, Madrid, Spain
sbernardos@sip.ucm.es

Abstract. This paper is an attempt to provide a general framework that helps develop a natural language generation (NLG) project from the conception of the need to the retirement of the product. This framework consists of a set of systematic activities that can be carried out in order to build or modify an NLG system. Assuming that a NLG system is software, we take IEEE Std. 1074-1997 –a standard for developing a software life cycle process– as a general framework and adapt it for NLG projects. We focus especially on activities related to domain information, requirements, design, knowledge elicitation and reuse.

1 Introduction

Natural Language Generation (NLG) is a Natural Language Processing (NLP) area concerned with the production of human languages. It is a relatively new field that has only been treated as a matter of its own since 1980's [11]. One of the characteristics of young disciplines is their lack of standards and/or commonly accepted practices, and NLG is no exception. As a consequence, one of the first problems people involved in an NLG project have to face is that they do not have an NLG-specific framework that guides them along the project. This issue is especially important in real-world projects, where the solutions must not only perform as required, but also conform to externally-specified criteria, typically related to time, personnel and equipment. In this paper we suggest a way to solve this problem, by providing a set of systematic activities that involve all the work related to NLG projects.

IEEE [6] defines software as 'computer programs, procedures, and possibly associated data pertaining to the operation of a computer system'. According to this, the product of an NLG project can be considered as software. Thus, NLG systems can be built or modified using one of the existing standards for developing a software life cycle process (SLCP). The main problem to apply this approach derives from the fact that those standards have been mostly used within the software engineering (SE) field, and NLG is a knowledge-intensive process [5], therefore, requiring of knowledge engineering (KE) solutions too [10]. That is why those SLCP standards need some adaptation to be really useful for an NLG project.

One of the mostly accepted standards that have been applied to the development of an SLCP is IEEE 1074, whose last revision dates from 1997 [7]. This standard also takes into account the harmonization with other well-known standards, e.g. IEEE/EIA 12207-1996 [8]. That is why we have chosen IEEE1074-1997 as the basis of our proposal.