

# The ObjectTalks Discourse Representation System

## Table of contents

1	Some initial information about ObjectTalks.....	2
2	About the ObjectTalks Discourse Management System.....	2
3	Typical Applications.....	2
4	Unique features.....	3
5	The ObjectTalks document Generation Process.....	3
6	Efficient Discourse Management .....	4
7	ObjectTalks software components.....	4
8	What's in the name?.....	5

## **1. Some initial information about ObjectTalks**

This page provides an overview of the ObjectTalks Discourse Management System (DMS). The ObjectTalks is my pet project I have been working on since somewhere around 1988. Needless to say, I didn't have a clue about what it was I was working on at that time.

## **2. About the ObjectTalks Discourse Management System**

The ObjectTalks DMS allows discourse (document content if you prefer) to be generated in the ObjectTalks discourse representation language. The generated discourse can in turn be translated to documents in different natural languages, with a variety of presentation forms, on a different media.

An example customer of our company might be a manufacturer of household appliances needing to provide information about his products in different formats and languages depending on the country in which his product is to be sold.

The ObjectTalks DMS allows this manufacturer to represent information about his products in a product model. Based on this product model, different document types (brochure, product summaries, product reference manual, etc.) would be defined. Given the product model and document types, ObjectTalks documents would then be generated (based on factual data in the product model). These documents would finally be mechanically translated from ObjectTalks to natural languages like (Japanese, Dutch, German and English), while maintaining a high quality of customizable translation.

In addition the ObjectTalks system would allow the translated documents to be presented in a variety of structures and formats. This to ensure that the translated documents are suitable for different target audiences such as consumers, resellers, support personnel and different presentation forms e.g. Web, paper, e-books.

Finally it is noted that the ObjectTalks translation process is sensitive to translation styles (legal, informal, marketing, technical, slang) so as to ensure that the translated text is most suitable for the target audience of the translated text.

## **3. Typical Applications**

The ObjectTalks DMS allows systems similar to the following to be readily implemented

- (Multilingual) multimedia product and project documentation generation
- (Multilingual) multi-channel self-service customer support
- (Multilingual) Help Desk knowledge bases
- (Multilingual) FAQ management
- (Multilingual) customer portals

## 4. Unique features

The ObjectTalks DMS is based on three unique pillars, namely unprecedented use of unrestricted Chomskian grammars, a distinguished extension of *intensional type* logic and a novel approach for representing the domain of discourse. This formal foundation makes the functionality provided by the ObjectTalks DMS unique.

Distinguishing features of the ObjectTalks DMS are:

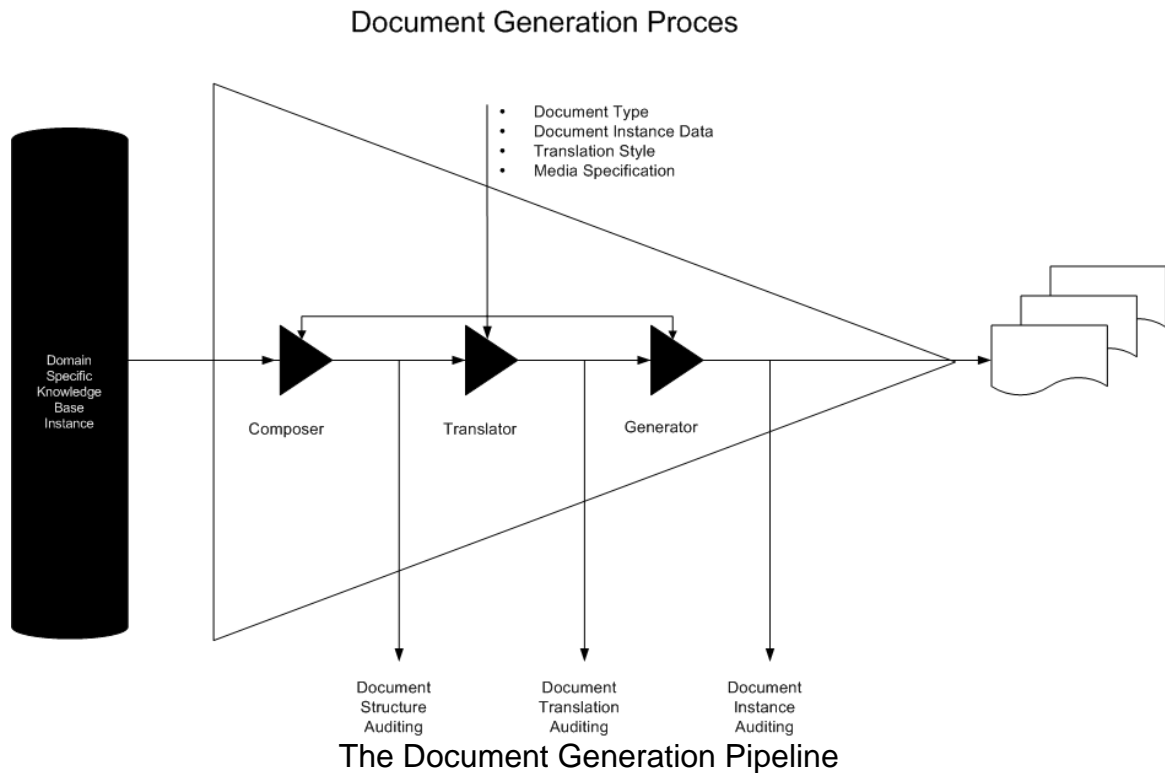
- Accurate and sophisticated knowledge representation techniques used to model the domain of discourse. This knowledge representation technique scales well, it can be implemented using readily available database technology.
- Higher order logic rules for establishing relationships between objects in the domain of discourse
- Sophisticated use of context sensitive Chomskian grammars used for document generation and parsing

## 5. The ObjectTalks document Generation Process

In ObjectTalks all documents are generated by *Document Generating Grammars*. Because the grammars actually generate discourse I also refer to them as *Discourse Generating Grammars*. These Discourse Generating Grammars generate the structure and the content of documents.

The document content (or discourse, if you prefer) generated are expressions in the ObjectTalks Discourse Representation Language. These expressions capture the logical significance (or the meaning) which is to be conveyed to the reader of the document. These expressions are then subjected to a translation process which maps the significance of the logical expressions to equivalent expressions in the target language, taking a translation style into account. The translated documents are finally to be presented in a target format such as HTML, PDF, RTF etc.

This process is shown in the following diagram.



## 6. Efficient Discourse Management

The ObjectTalks document generation is designed to reduce the costs of maintenance in the entire document production process.

Maintenance in the domain model occur at a single point (in the discourse model). Changes in document structure and content occurs at a single point (document type). Changes in the translation style occurs at a single point (in the translation module) and changes in the presentation occurs at a single point (presentation style sheet). A visualization of this is attempted in the following diagram.

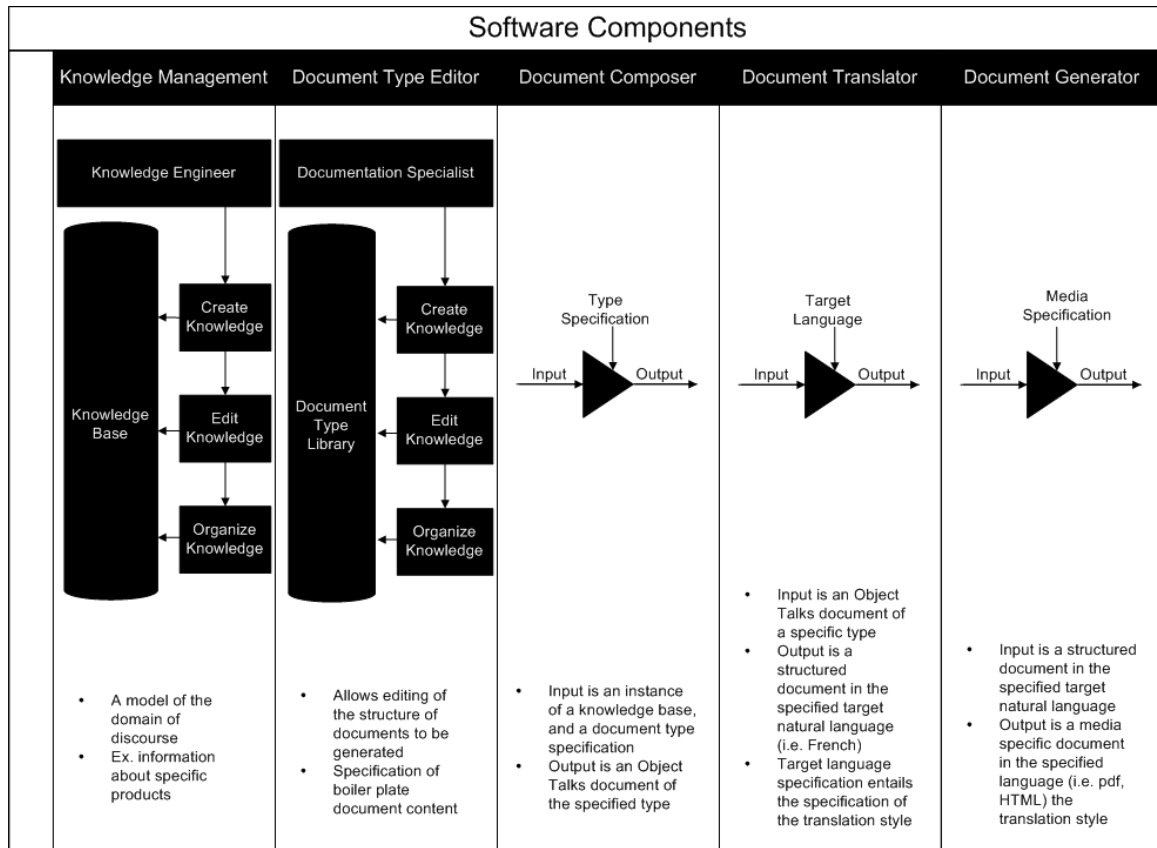
### Document Generation

## 7. ObjectTalks software components

The Software components are *The ObjectTalks application server* which manages an ObjectTalks universe of discourse and stores the raw data in conventional database management systems. The *Discourse model maintenance tool* will is a tool for maintainince discourse models and their associated applications. *Discourse Development tools*, used for the development of Documents Type, translation modules etc. etc.

The document composer, the document translator and the document generator provide the functionality suggested by their names. A selection of these components are presented in

the following diagram.



Software Components

## 8. What's in the name?

ObjectTalks is the name of everything I write. It started somewhere in the early nineties and everytime a wrote something new I would call it ObjectTalks. It seems only natural to call my Discourse Representation System ObjectTalks as well.