

MODEL-DRIVEN DEVELOPMENT OF A DOMAIN SPECIFIC LANGUAGE (DSL) FOR ELECTRONIC DICTIONARY CREATION

Gamzayev Rustam

V.N. Karazin Kharkiv National University, Kharkiv

The motivation for creating a DSL for electronic dictionaries is to streamline the process of creating and maintaining electronic dictionaries while providing a high level of abstraction, flexibility, and expressiveness. With the increasing demand for electronic dictionaries to support language learning, translation, and other linguistic tasks, there is a need for efficient and effective tools for creating and maintaining such dictionaries. DSLs offer a way to address this need by providing specialized languages tailored for specific problem domains.

Existing approaches for electronic dictionary development include using general-purpose programming languages, markup languages, and database management systems. However, these approaches can be cumbersome and may lack the expressiveness and abstraction needed for efficient dictionary development. DSLs have been successfully applied in various domains, including software engineering, data analysis, and natural language processing (NLP), to improve productivity and software quality by providing a higher level of abstraction and expressiveness tailored to the specific domain.

In designing the DSL for electronic dictionary development, we identified the following key requirements that we need to follow:

Abstraction: The DSL should provide high-level constructs that abstract away from low-level implementation details, allowing developers to focus on the domain-specific aspects of electronic dictionary development.

Expressiveness: The DSL should provide constructs that enable developers to concisely and precisely represent the structure and content of electronic dictionaries.

Flexibility: The DSL should support various types of dictionaries, such as monolingual, bilingual, and multilingual dictionaries, and accommodate different language-specific features and requirements.

Extensibility: The DSL should allow for the addition of new features and constructs as needed, without requiring significant changes to existing dictionary models.

Integration: The DSL should integrate with existing tools and technologies for dictionary development, such as NLP libraries, databases, and user interface frameworks.

We designed the DSL using a model-driven approach, which involves defining a metamodel that captures the essential concepts and relationships in the domain of electronic dictionaries. The DSL's syntax and semantics were developed based on this metamodel using xText tool. Then using a custom implementation that translates the DSL models into a target representation, it is possible to handle user input and perform appropriate logic.