Compiler from JavaScript to Python

Author: Ing. Jozef Paľa **Tutor:** doc. Ing. Ján Janech, PhD.
University of Žilina, Faculty of Management Science and Informatics

The goal of this thesis

Major idea behind the whole concept of this thesis is to allow JavaScript developers to create their own script or application which would run on personal computers. JavaScript is one of the most common language amongst developers especially because its unique position in web environment. However, it has limitations which makes it nearly impossible to use it in environment outside of the web. Therefore, it is necessary to translate the application written in JavaScript to different language in order to make it possible to run the code on environment of personal computer. Design of this translation process as well as its proof of concept is main target of this thesis.

Current solutions

The idea to translate JavaScript into different languages is not new. During research of other existing solutions, applications with various purposes and approaches towards the problem were found and tested. This just supports the fact that JavaScipt is a very popular language and developers using it are interested in creating applications for different platforms with it and without further complications with learning new language and platform specific characteristics.

However, most of the solutions were tied to a specific platform or implementation was just using JavaScript engine on the background. A lot of them were abandoned without fully finished implementation.

One of the most successful solutions with active development and worldwide popularity was using JavaScript code extended with various libraries and APIs to develop smartphone applications. In this case, JavaScript source code is translated into native code for selected mobile platforms. This solution is very similar to target of this thesis. Unfortunately, it is limited to mobile platforms currently.

Activity diagram

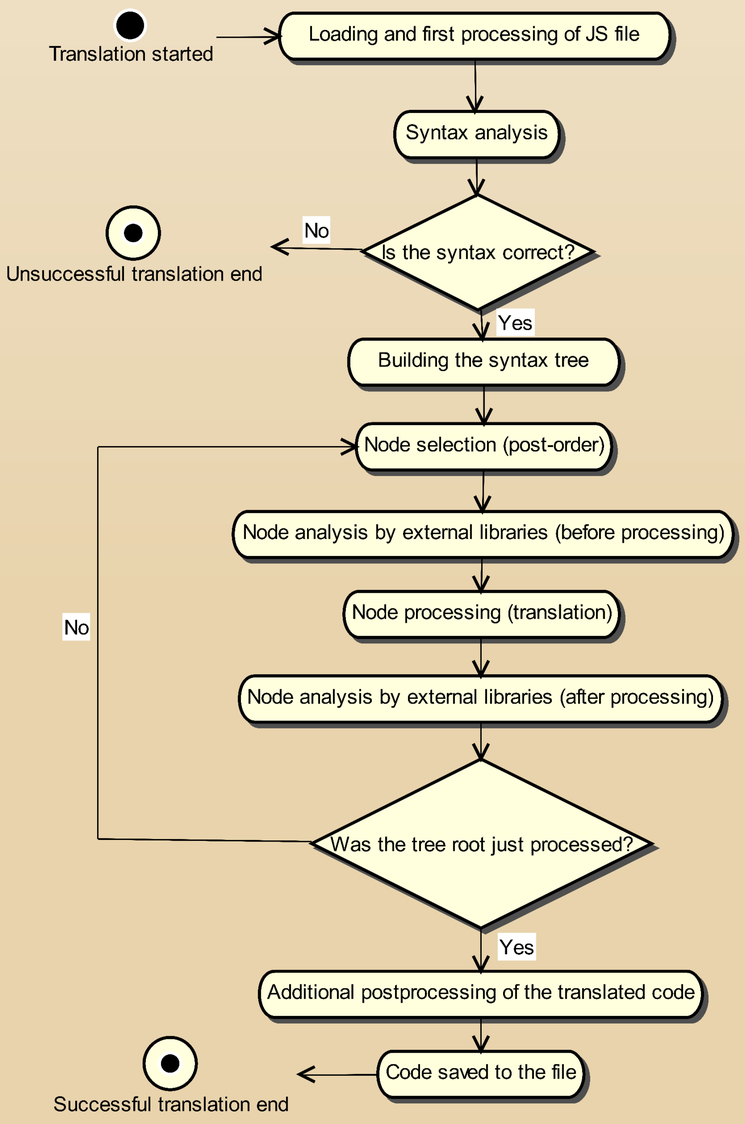

Diagram describing a process of translating the code.

Description

JavaScript belong among the most popular programming languages. This is caused by its simplicity which allows newcomers to grasp the language within few minutes or hours. Furthermore, huge influence of internet and demand for responsive and dynamic web applications helps bring more and more new developers to use this language. However, JavaScript is mostly limited just to its usage within a web browser.

On the other hand, Python is advanced general purpose programming language with huge family of various add-ons and libraries used mostly in desktop computers, but it is not limited just to that and due to its support for various operating systems it can be used in majority of existing environments.

However, what if JavaScript developer would like to create desktop script or application? Currently available options are limited and the developer ends up with learning a new language before they can create such script or application. This thesis aims to simplify that by translating code from a known language (JavaScript) directly to Python code. As an addition, this application offers options to extend functionality of JavaScript with custom libraries supporting advanced programing functions of Python language which would be otherwise unavailable.

Furthermore, the solution was designed with reusability in mind. This is important to not limit this thesis just for translating from JavaScript to Python. Even though the final solution translates from JavaScript to Python, it is possible with slight changes to translate code between any two languages.

Diagram of the solution's architecture

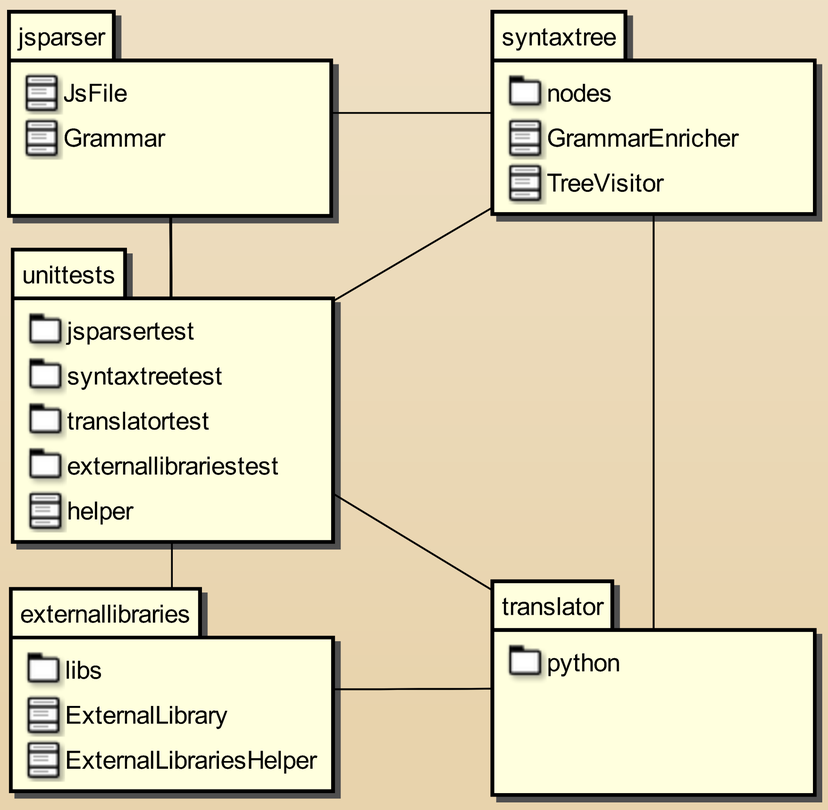

Diagram above visualize architecture of proposed solution. This architecture is kept simple and splits the application into four major packages. All of them are covered with unit tests represented by the fifth package. Brief description of those packages follows:

- **jsparser** – main entry point, JavaScript file is being loaded and processed here. JavaScript language grammar is defined here as well.
- **syntaxtree** – contains definition of how to convert components defined by grammar into the syntax tree, which is representation of code without the unnecessary information.
- **translator** – translation of code based on given syntax tree takes place here.
- **externallibraries** – mechanism to extend JavaScript functionality with more advanced tools and functions such as reading and writing to the file. External libraries scan nodes of syntax tree and they can apply custom modifications to it or they can provide different translation than one done by built-in translator.

To possibly support different languages, few modifications would have to be done. In case of changing the source language, new grammar for that language would have to be defined. Furthermore, description on how to convert this grammar into the syntax tree would have to be provided (classes Grammar and GrammarEnricher on the diagram above), When it comes to the output language, only translator would have to be defined. This translator has to implement TreeVisitor class from the above diagram.

Conclusion

Thesis successfully tackles the problem of translation from one source language to the another. Translation from JavaScript to python was created as a proof of concept of the proposed architecture and activity diagram. The application allows the web developers to create simple script or stand-alone application using just the JavaScript which is then translated to Python.