

jTLEX - A Java Library for Timeline Extraction

Student: Christopher Eberhard

Mentor: Mustafa Ocal

Instructor: Dr. Masoud Sadjadi, Florida International University

Problem

Current TimeML implementations provide an output of annotated text files which result in generated temporal graphs only revealing the ordering of events and times leading to inadequate information for many programs due to the current systems lacking full ordering and providing data abstraction.

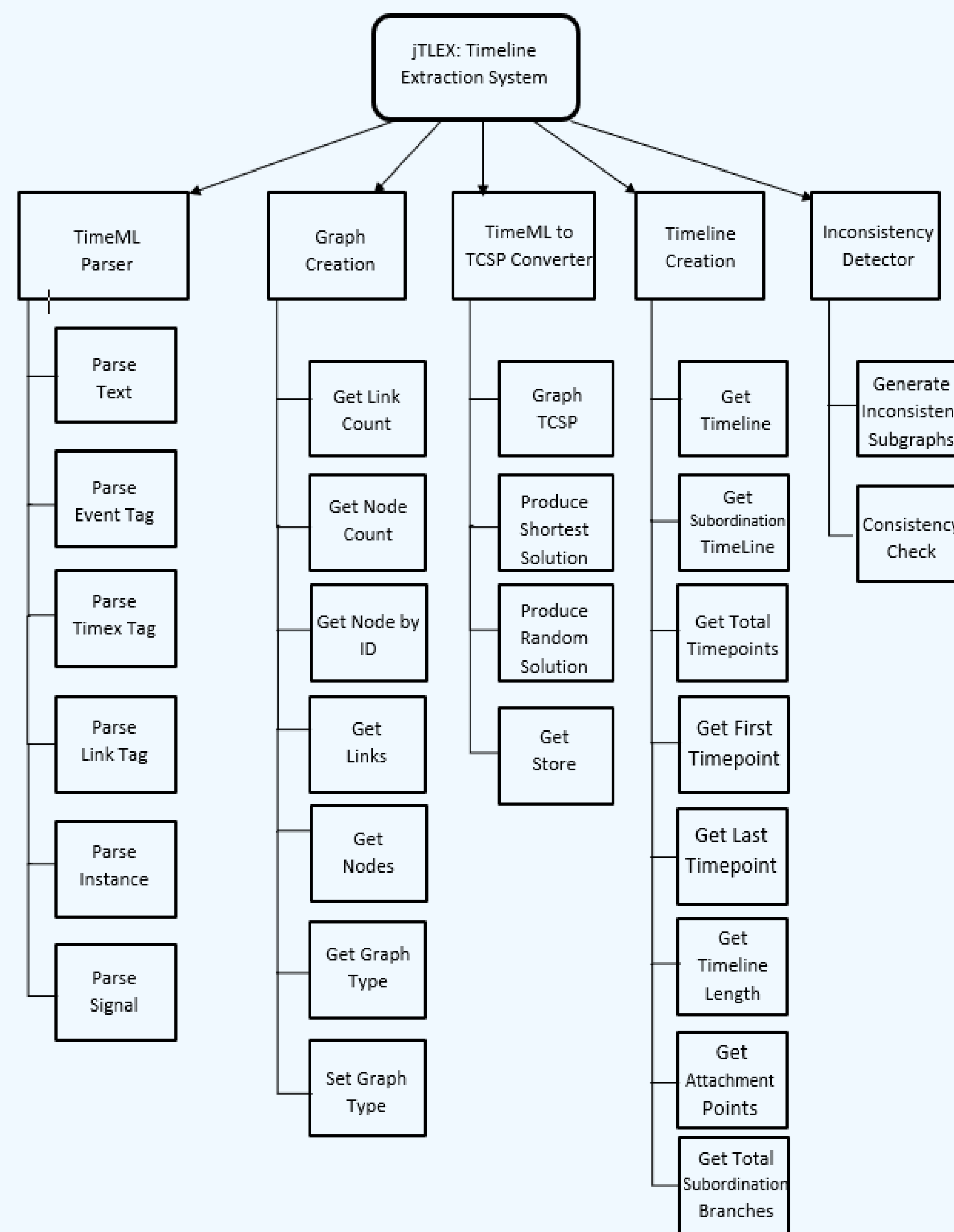
Current System

Currently the means of TimeML produces an output of its contents as annotated text files, the graphs generated from the files only revealing part of the information in the form of the order of the events and times. As a result of a lack of full ordering and data abstraction the process proves to be inadequate.

Requirements

The jTLEX java library allows for users to parse TimeML files to get the times, events, links, signals, and instances from a TimeML file. Users can then jTLEX to create a TimeML graph from the file which was parsed, after which said graph can be converted to a TCSP, checked for inconsistencies, or locate any indeterminant time points. Lastly jTLEX can provide an accurate timeline of the events from the TimeML file.

System Design



Object Design



Implementation



For the Implementation of jTLEX we used Eclipse as our java IDE, we also made use of subversion when creating the jTLEX java library to allow for the capability to . Lastly, we made use of Junit 5 in our method of creating tests for the classes we created for the jTLEX library to ensure that all the methods were operating correctly.

Summary

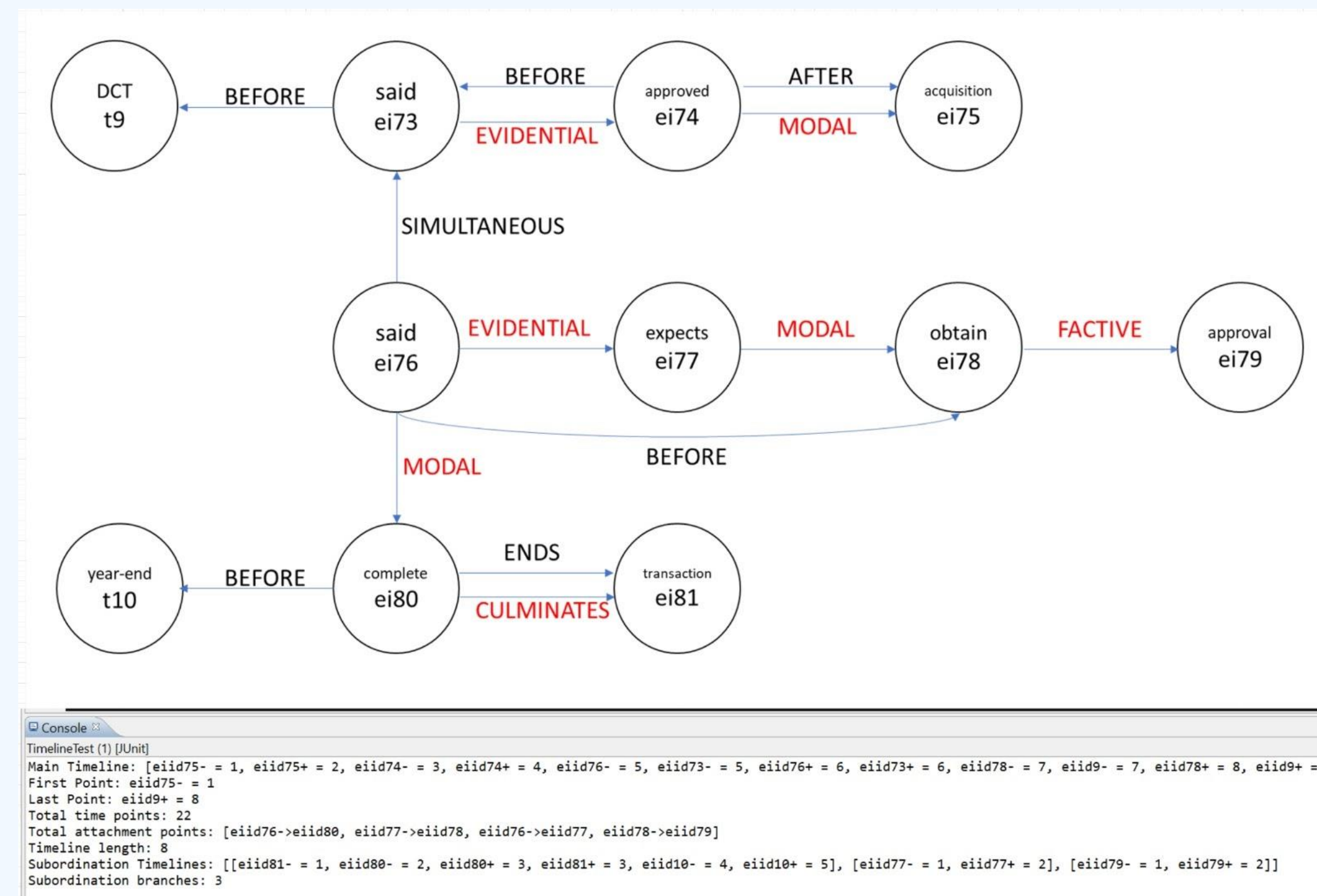
jTLEX seeks to provide a more sufficient means of extracting information from TimeML files to provide users to gain a proper timeline of the events that occurred with in the TimeML files and allow for accurate TimeML graphs to be generated presenting the information in a clear and accurate manner.

Verification

Test File	testUtils.java
Purpose	To ensure to proper operation of the Graph TCSP constraints generation method
Preconditions	The Graph TCSP method has an existing TimeML graph as input for conversion.
Input	A TimeML Graph.
Expectation	The Graph TCSP method generates a correct list of constraints based on the predetermined list of link type conversion algorithms.
Result	Pass

Test Files	Timebank Corpus
Purpose	To ensure that the TimeML parser method is capable of locating a specific instance in files and retrieve the id of the instance in the file where it was located.
Preconditions	There is a proper TimeML file as input for the method to read.
Input	A list of TimeML files passed to the TimeML parser one at a time.
Expectation	The TimeML Parser will locate the specified instance five times and provide the id of each instance in the file located.
Result	Pass

Screenshots



Acknowledgement

I thank Mustafa Ocal for assistance, cooperation and mentorship that I received throughout this process.