Methods and Software for Probabilistic Assessment Applications
Event Sequence Diagram Visualization

Mochuan Liu, Shandong University
Contact: mochuan.liu@mail.sdu.edu.cn
Program Advisor: Prof. Mihai A. Diaconeasa & Arjun Earthperson

Introduction

Probabilistic risk assessment (PRA) is a systematic and comprehensive approach that conducts a risk assessment of a variety of potential outcomes.
The **OpenPRA** program aims to provide a unique platform that integrates multiple PRA methods and tools into a holistic, easy-to-use, and highly customizable framework.
In this project, we are supposed to solve the issues in the development of the software OpenPRA **Web-Editor.** My job focus more on the frontend development, and center on the **event sequence diagram visualization**.

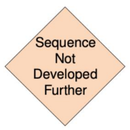
OpenPRA Web Editor
build v0.7.70

Methods

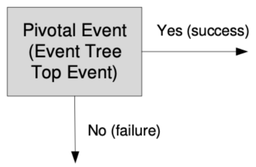
The **Event-sequences diagram(ESD)** is an analytical method intends to illustrate all possible success paths from a particular **accident-initiating event to a safe-shutdown condition.**

The start of ESD, typically something happens.

Turn to another page when ESD is to big too show on one page.



The end of ESD branch, all kinds of end conditions and expressions for not developed sequence.



Specific judgment conditions, used to form two divided branches of the ESD.

The comments in ESD, when it comes to nuclear engineering, talks about the plant state and condition.

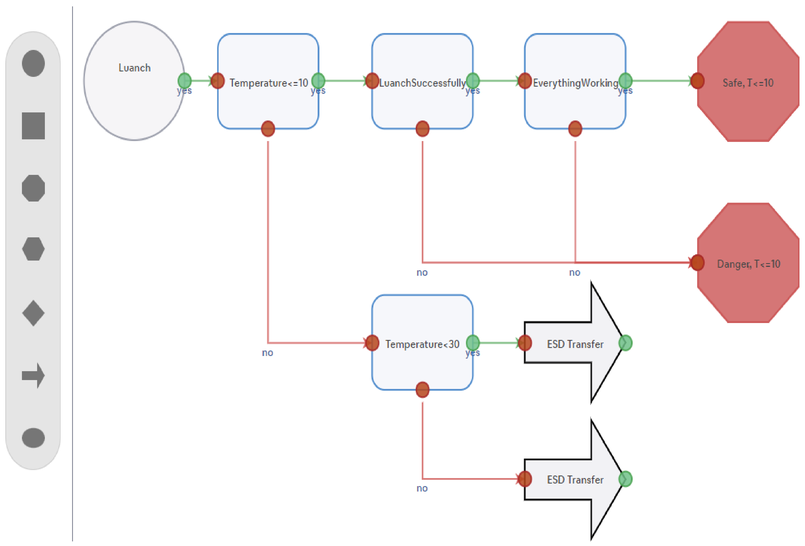
Tasks

Task 1: Event sequence diagram symbols
In the Event Sequences interface, only Initiating Event, Functional Event and End State were implemented more **specific symbols** need to be added in order to draw a more detailed event sequence diagram for PRA of nuclear plants.

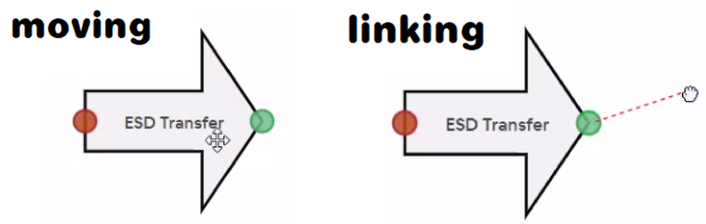
Task 2: Enable **arrow key** control and shortcuts
If we are allowed to use an arrow key to make control on the vertices, then the operation will be much more convenient. We will develop this in our project.

Results

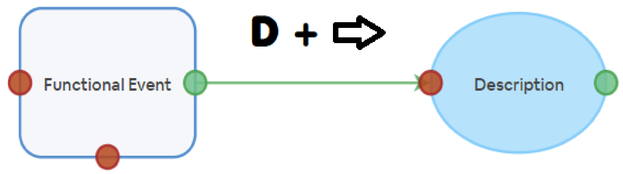
1. Many **Typescript classes** related to the symbol are modified in order to add new symbols, including the style, the shape, the vertex value and the geometry. Applying **vector math**, the generic shape of the symbols are redrew into more specific ones with significant colors.



2. **Ports** were added to each vertex so that it is easier to make links between vertex and separate the **linking** function and the **moving** function.



3. Two functions were implemented with arrow keys: **Navigation** and **Keyboard Shortcus.** We use “hotkey.js” library to bind keys with actions.



Conclusion

1. To implement a software, we need both frontend and backend support. Applying Typescript, React and MxGraph library, I have a deeper understanding in frontend develop and visualization.
2. ESD is a method of great significance in PRA. Using specific symbols, an ESD can show us the path of risk. Through the implementation, I learned much about ESD model.

References

1. PRA Procedures Guide
https://www.nrc.gov/docs/ML0635/ML063560439.pdf