COS214 Project Report

Optimal Circuitry

Matthew Penning	u04948433
Nathan Opperman	u21553832
Danie Schoeman	u15036058
Matthew Verster	u16016239
Israel Bango	u04865503
Bandisa Masilela	u19028182

Contents

Introduction

The aim of this project was to develop a system which could simulate a war. The simulation developed makes extensive use of the principles laid out by R.J. Rummel in his notes Understanding Conflict and War (Rummel, 1975). Furthermore, the concept of war theatres, proposed by Carl von Clausewitz (Clausewitz & Graham, 2017), is used in the design of the simulation software.

The following chapters further describe the workings of the simulation software – how it was developed, reasons behind various design decisions, and the design patterns used to make the software understandable and extendable.

Research

To understand war and the simulation of war, one must have a grasp of the various components which make up a war. These components will be discussed here, before the content of this report, so that the reader may gain an insight as to why various components of the war simulation have been included in this project.

War theatres

A war theatre is essentially a closed system in which a series of battles between some parties take place (Clausewitz & Graham, 2017), it is important to note that a war theatre is separated from its surroundings by defining important events which take place within its boundaries. Since its boundaries are protected it is simpler to conceptualise in a simulation than an open system, and will require minimal outside dependencies.

Transport

Transportation plays a vital role in any war. The ability of a country to get consumables such as food rations, ammunition, and medical supplies to its front lines directly influences its ability to fight well (Brown, 2018). The risk of having a transport corridor destroyed by an enemy can be devastating to an army and its progress.

Entities

Entities tend to refer to the human element in a war, whether it be civilians, troops, medics, or even countries which neighbour the war theatre. These entities collectively establish the aims of the war which they represent.

Phases of War

A common way to break up a war into its constituent phases is as follows. There are six phases, they are:

1. The war takes shape
2. Deter the opponent
3. Seize the initiative and begin the war
4. Dominate the opponent
5. Stabilise the theatre
6. Enable civil authority

These phases fully describe a major conflict and allow the entities involved to plan around the eventualities of war (Scharre, 2016).

Functional Requirements

The war simulator will need to accommodate battles between states and incrementally update the data each participant holds, such as their total troop count. Finally, the software will have to output a result of the war, declaring a winner.

The software will allow multiple countries to form alliances with one another which will make it possible for them to share their troops' power with one another. The troops available to a country will contribute towards their strength rating, and this will be decreased accordingly when a battle commences between two countries.

The various types of troops will each carry their own strength ratings, for instance, an infantryman will have a strength rating of one. This directly translates to the amount of damage they are able to inflict on their opponent during battle.

Attacking rounds will take place in a turn-based fashion until a winner has been declared, at this point the simulation will cease and the result of the war will be given to the user.

The simulation software will make use of eleven design patterns in its implementation. The patterns in question are:

- Memento
- State
- Strategy
- Command
- Abstract Factory
- Observer
- Composite
- Factory Method
- Mediator
- Facade
- Chain of Command

Discussion of Design Patterns

This section serves to briefly describe the reasoning behind each design pattern's inclusion. More detailed information on each pattern can be found further on in this report.

Abstract Factory

This pattern, along with Factory Method, will be used to create the participants involved in the war. This involves creating the objects necessary to contain all the information relevant to each country, such as their name, troop count, and alliances.

Abstract Factory, specifically, will be used to create the militaries used by each of the countries. It is particularly well suited for this purpose as it allows the creation of sets of dependent objects without the need to specify their concrete classes. This allows the various groups in each military to be used in the same ways.

Factory Method

The Factory method is used to create the countries taking part in the war. This particular pattern has been chosen for this purpose as it allows the countries to all operate on the same interface, which allows them to operate easily with one another.

Memento & Command

The Memento pattern will be used in this system to save the current state of the war during the running of the simulation. The Command pattern will be used in conjunction with the Memento pattern and will act as the caretaker in this system and will facilitate the user controls affecting the simulation progress.

State & Strategy

Two more patterns which will operate closely are State and Strategy.

The State pattern will allow the country objects to alter their behaviour when their warring state changes between two different options: Attack and Defend. The first state, Attack, provides a country with the ability to launch an attack on another country and enter into a battle with it. The country being attacked is then assigned the Defend state which will allow it to react to the attack with certain behaviours only available to a defending country.

The Strategy pattern will offer three possible options to their state, these indicate whether the country wishes to attack or defend using either a Strong, Medium, or Weak attack/defence strategy. The strategy chosen will influence the military power of the related country during a battle.

Observer

The purpose of the Observer pattern is to watch the countries as they battle, and to inform all interested parties of the outcomes and how their data fields should be modified post-battle. Namely, the observer will update the troop counts for each of the countries involved in a battle.

Mediator

The Mediator works in conjunction with the Observer pattern. It will be used to mediate the phases of the war by using the concrete observers as colleagues, which will update the mediator of any any changes after an action such as attack/defend/move, the remaining colleagues will not be notified immediately, instead, this information is stored and validated until it signifies a phase change in the war, this will then be notified down to the other colleagues. Thus essentially functioning as a simulation master, or main communication channel for relaying the phase of the war to the other participants. The phases of the war will be determined by the extent of damage and reaction that each side causes or initialises in reaction to each other.

Composite

The Composite pattern maintains a list of countries that are members of a given alliance.

Each country member is represented as a leaf in the composite pattern. Countries can be added or removed and information about them can be printed to the simulation interface. Each country created by the Factory pattern mentioned above will have an AllianceCountry which is the composite class that instantiates the alliance.

Facade

The purpose of this design pattern is to provide the user with an interface which hides the complex underlying system from them and instead provides the user with a clean and understandable interface with which to use the simulation software.

Chain of Command

The theatre and the modes of transport rely on the Chain of Command. This pattern allows the simulation to pass the war theatre from one phase to another as it progresses through the different stages of the war. Each handler in the chain of command represents a new phase in the war.

Application of Design Patterns

Factory & Abstract Factory

To begin the simulation, the countries involved are created using both the Factory Method and the Abstract Factory pattern. These two patterns are represented by the following class diagrams:

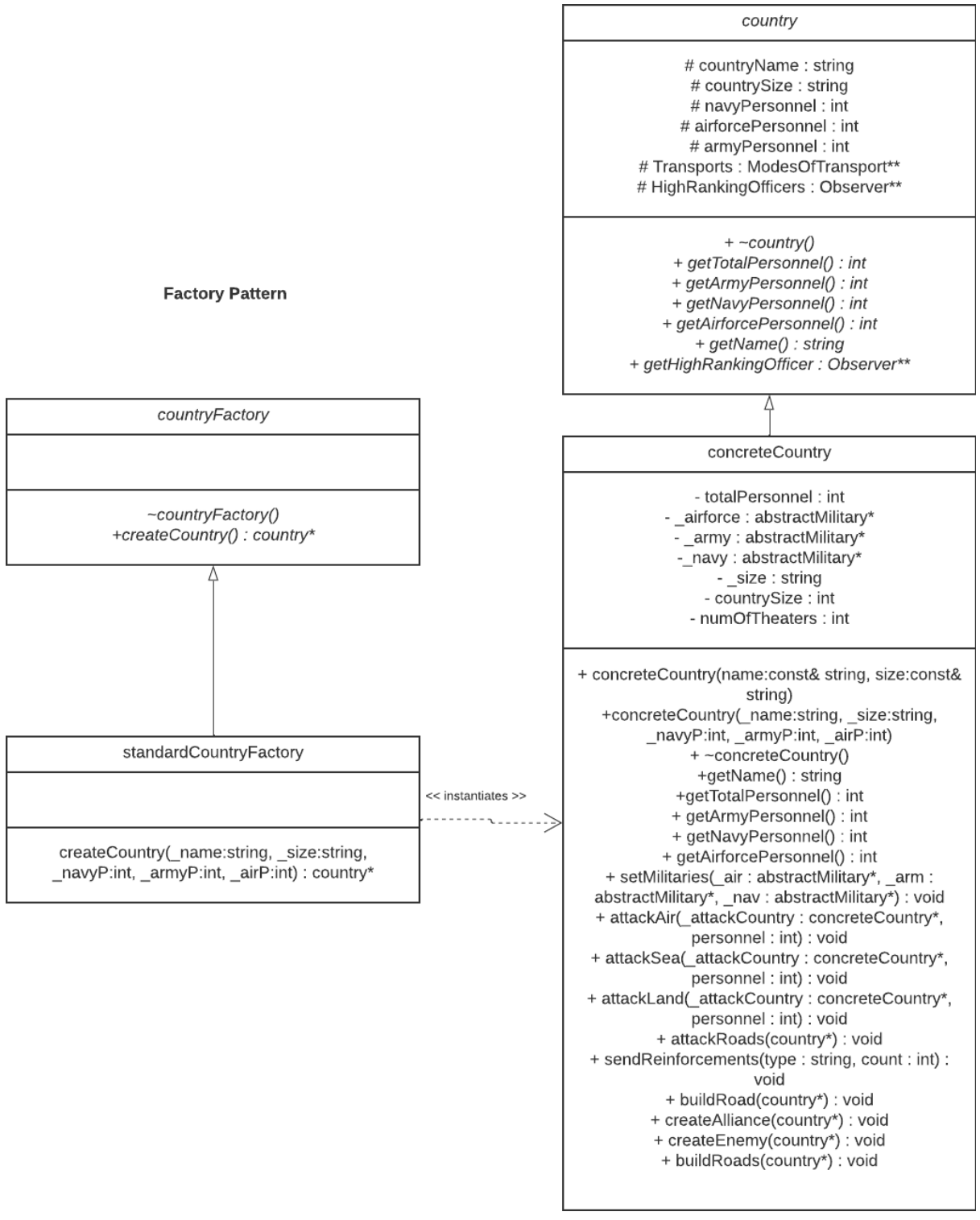

Figure: Class Diagram of Factory Method Design Pattern

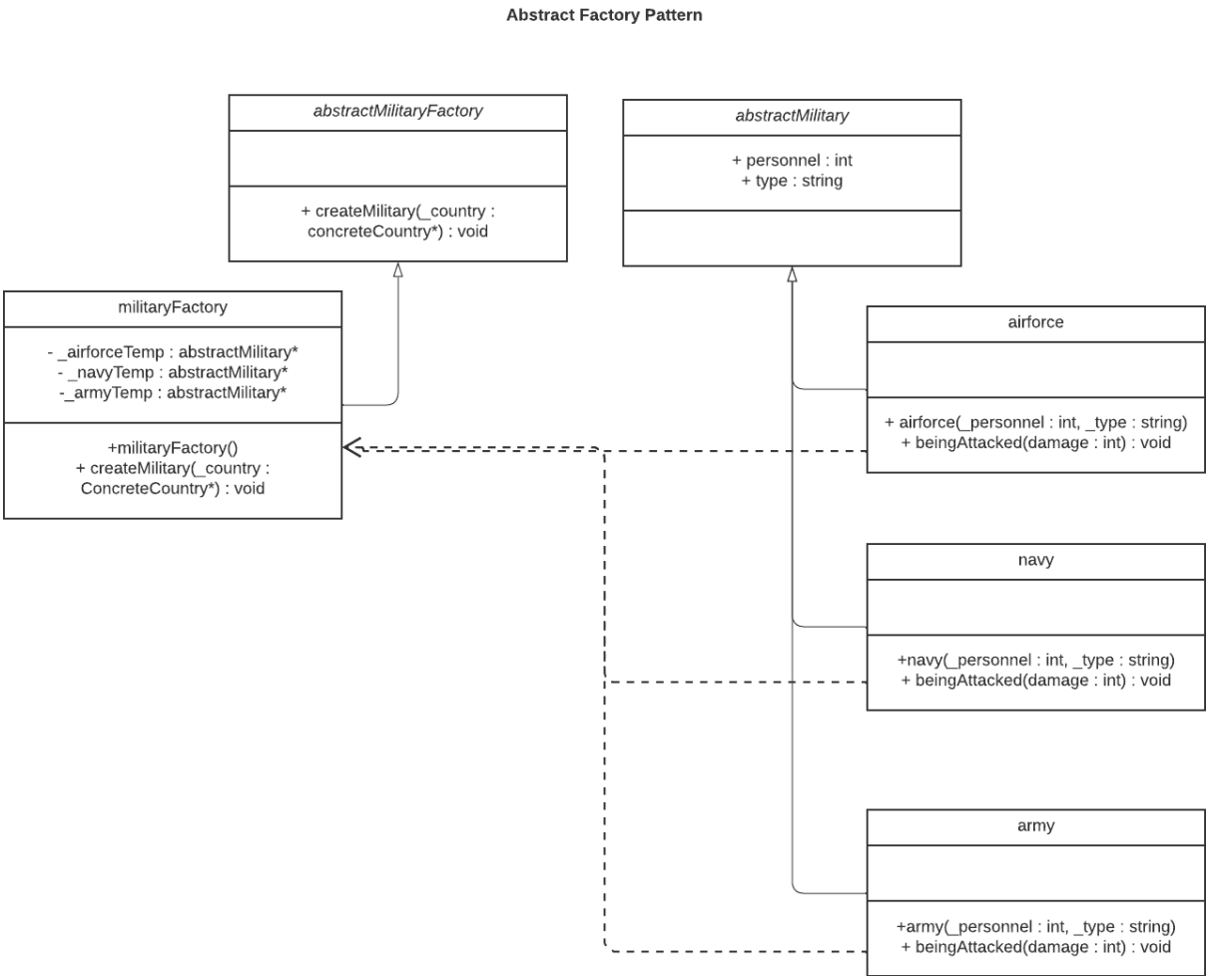

Figure: Class diagram of Abstract Factory Design Pattern

The Factory Method pattern is responsible for creating the country objects which are each provided with a military by the Abstract Factory pattern.

The objects created by these patterns are as follows:

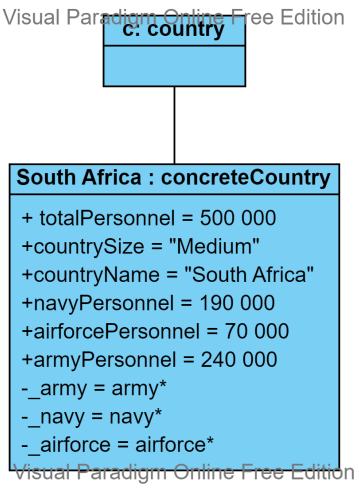

Figure: Object diagram showing a country after creation.

These objects will be modified during the course of the war and some of the possible changes can be seen in the following diagrams.

After the second phase of the war, a country object may appear as follows:

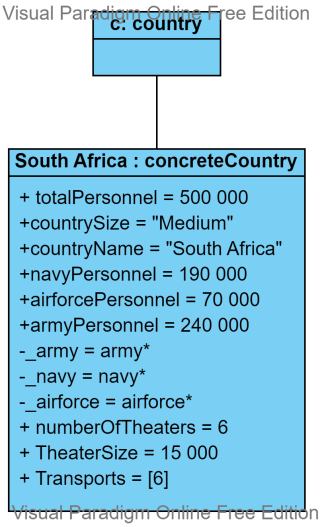

Figure: Country object after second phase of war.

The next two figures show a military object being attacked and how the data fields inside are updated to reflect the damage it sustained.

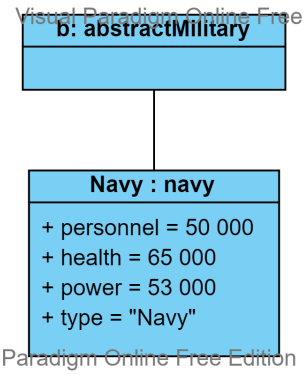

Figure: Naval object after being created and assigned to a country.

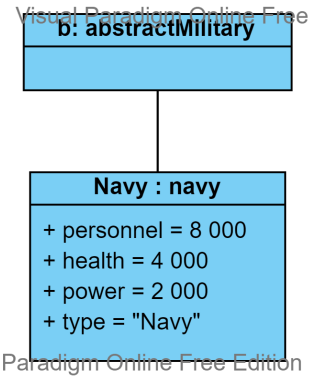

Figure: Naval object after a fierce fight with another naval military object.

Command & Memento

Reacting to user input as well as saving the simulation state, which will allow the user to undo actions, are made possible by the Command and the Memento patterns.

They are implemented as shown in the following figure.

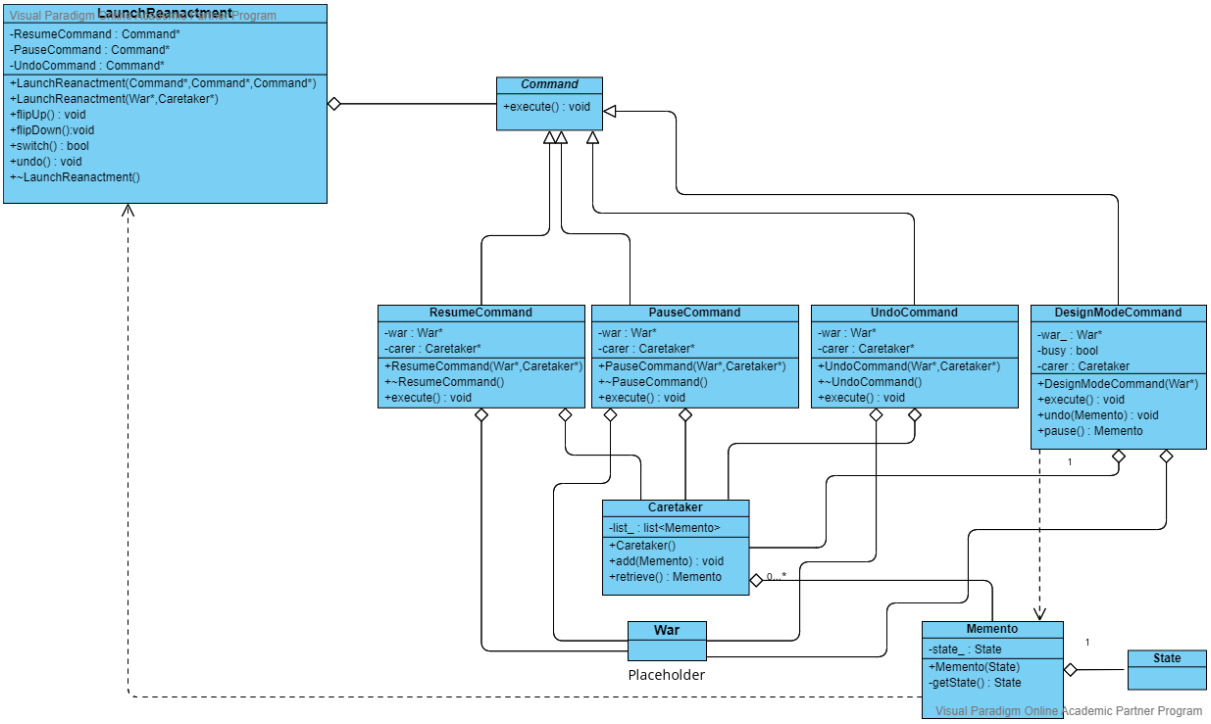

Figure: Class diagram showing Command and Memento patterns.

The Concrete Commands can all be seen in the middle of the figure, and the Memento pattern is positioned on the lower right of the figure.

The following sequence diagram shows how the system reacts to a user requesting an undo operation.

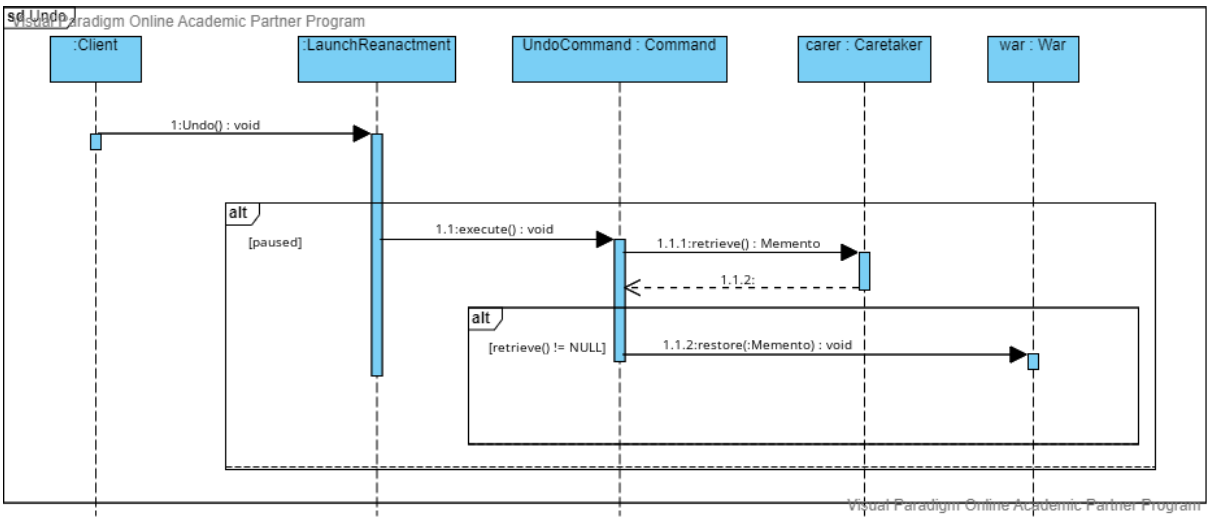

Figure: Sequence diagram showing undo operation being processed.

State & Strategy

In the following diagram, both the state and the Strategy class diagrams can be seen. When attacking or being attacked, a country will have the option to act in either a Strong, Medium, or Weak way. These actions are facilitated by the Strategy pattern. The attacking or defending is managed by the State pattern allows the object's behaviour to modify based on their current state, whereas the Strategy pattern allows the user to change the behaviour of the object.

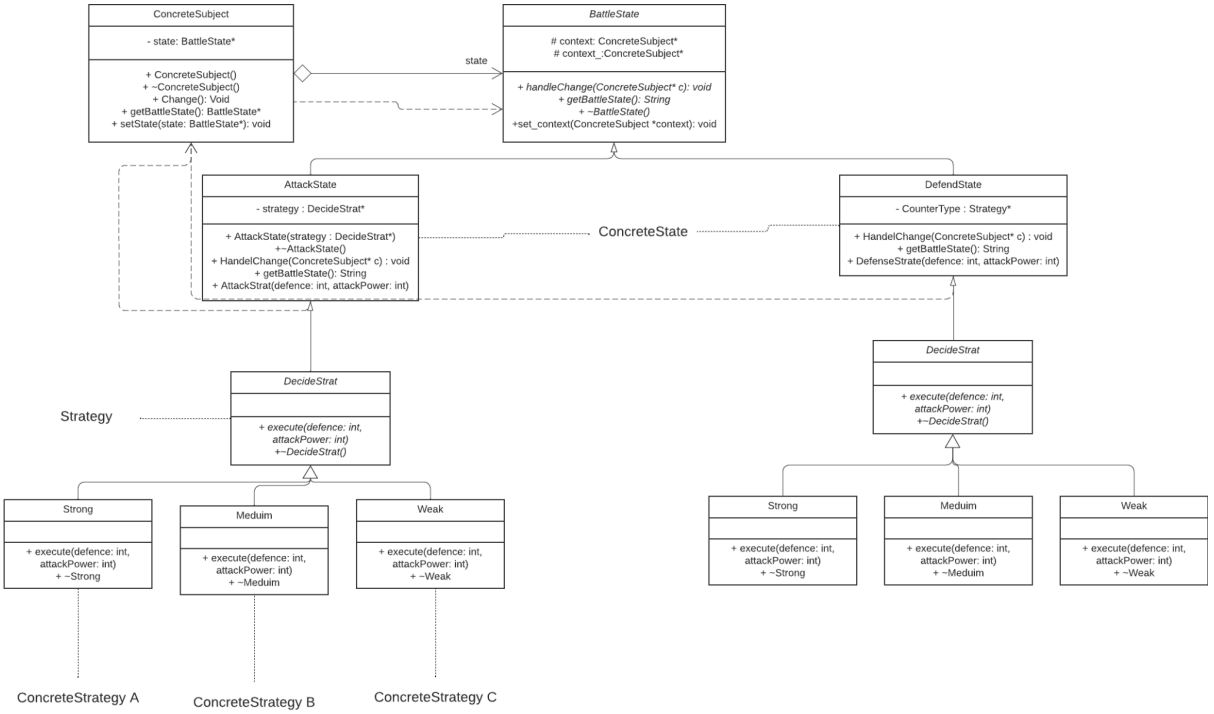

Figure: Class diagram of both Strategy and State patterns.

Observer & Mediator

The Observer and the Mediator work together to ensure that the countries involved in the conflict stay up to date at all times. The Concrete Observers act as the equivalent of army generals, watching over the performance of their respective militaries. When the number of active troops change, they relay their observations on to the mediator pattern which acts as a central object routing all of the communication to the correct participants. The mediator pattern also initiates the phase changes in the war theatre. These changes are determined by how much damage has been inflicted.

The class diagram showing how these patterns work together can be seen in the figure below:

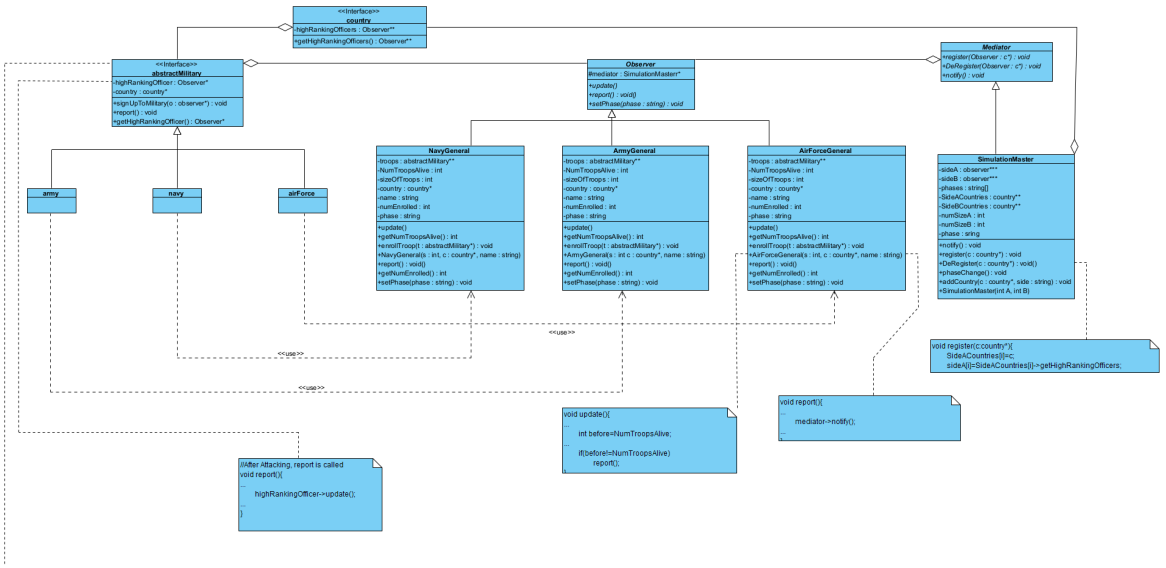

Figure: Mediator and Observer UML class diagram.

To illustrate the timing of these interactions, the sequence diagram for this subsystem has been included below:

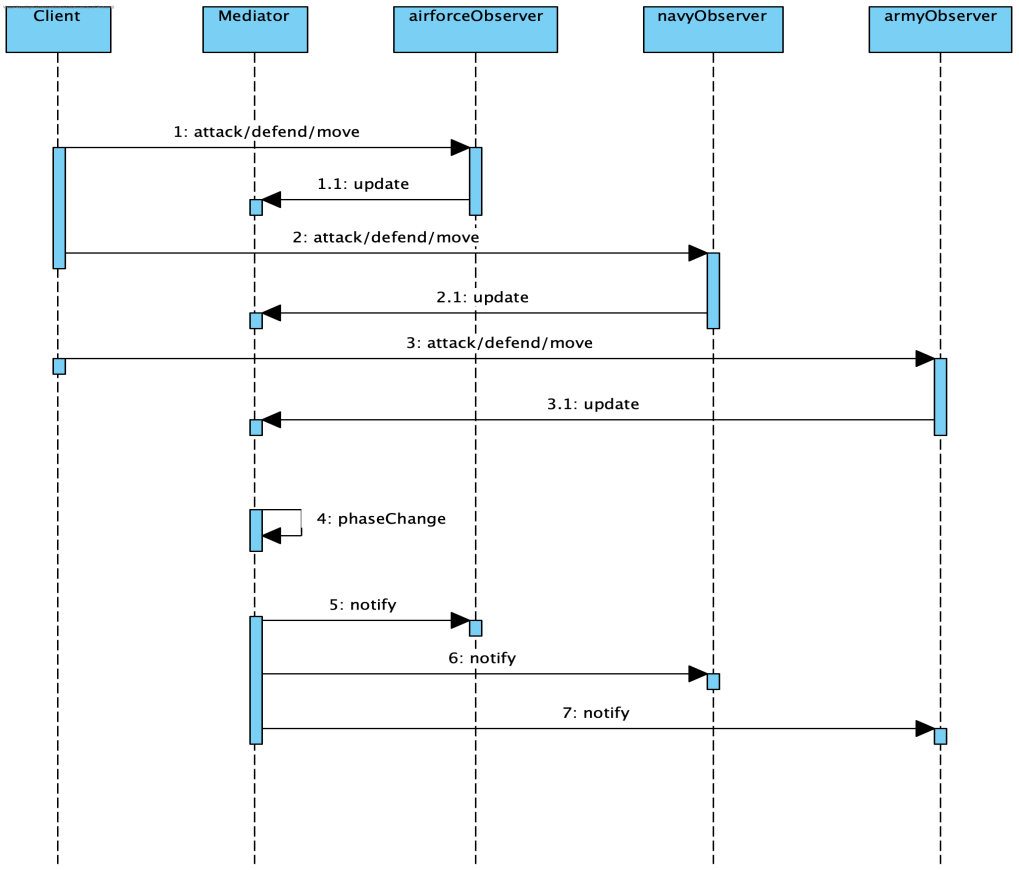

Figure: Sequence diagram for Mediator and Observer subsystem.

The communication diagram below illustrates the interactions between these patterns even further. This diagram shows how the phase may change depending on the results of an attack on a country.

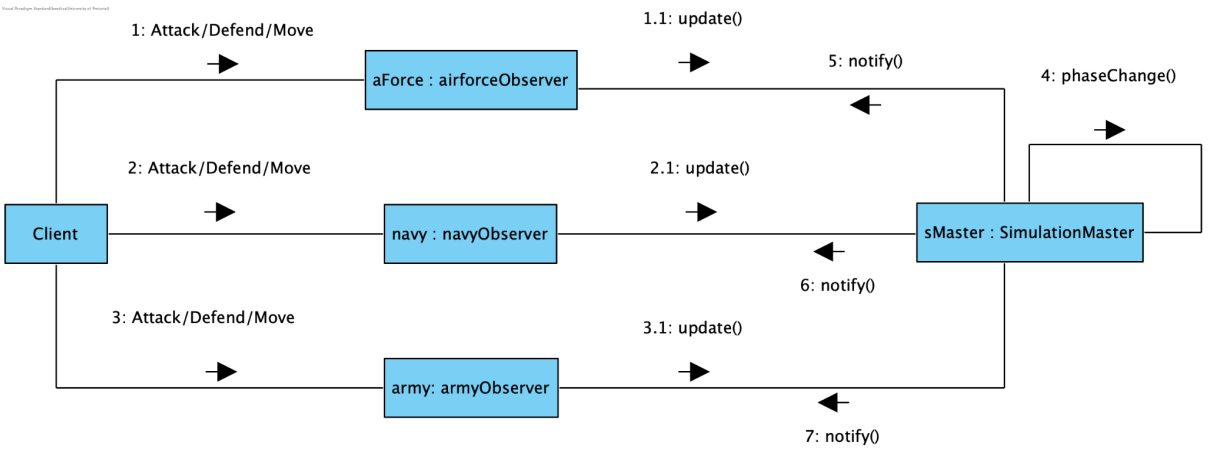

Figure: Communication diagram for Mediator and Observer patterns.

Composite

The Composite pattern is responsible for keeping track of and printing details relating to the alliances present in the war simulation. It allows the alliances to change and modify their details without causing issues in other parts of the simulation, as the alliance details are worked out every time the Composite pattern is used and they take the form of a tree, allowing the pattern to deal with simple and complex elements in the same way.

The UML class diagram can be seen in the figure directly below. Further on, a diagram showing the communication between the objects involved in the Composite pattern is shown, and further below, a sequence diagram depicting the same scenario as the communication diagram is displayed.

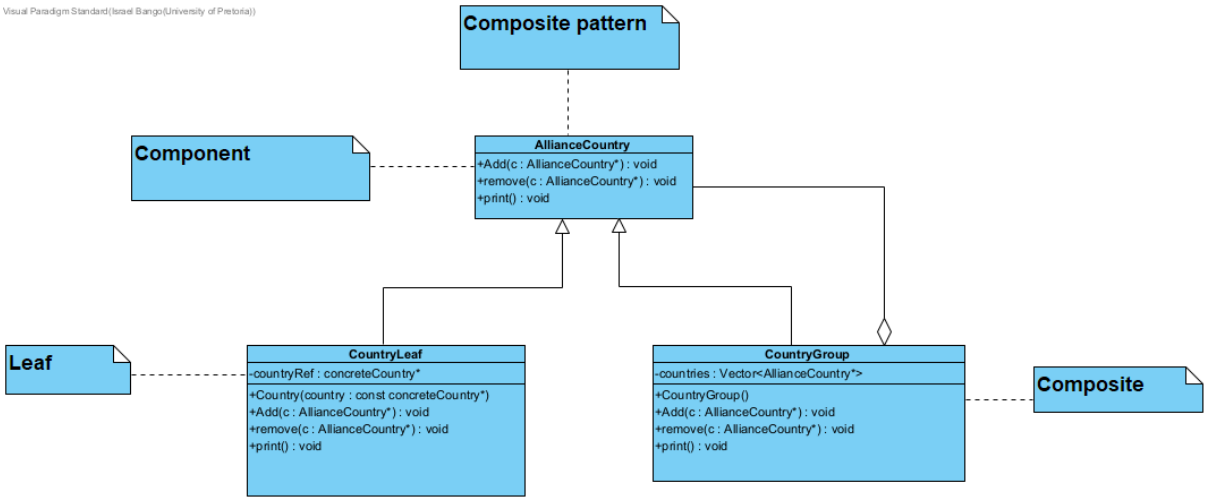

Figure: Class diagram for Composite design pattern.

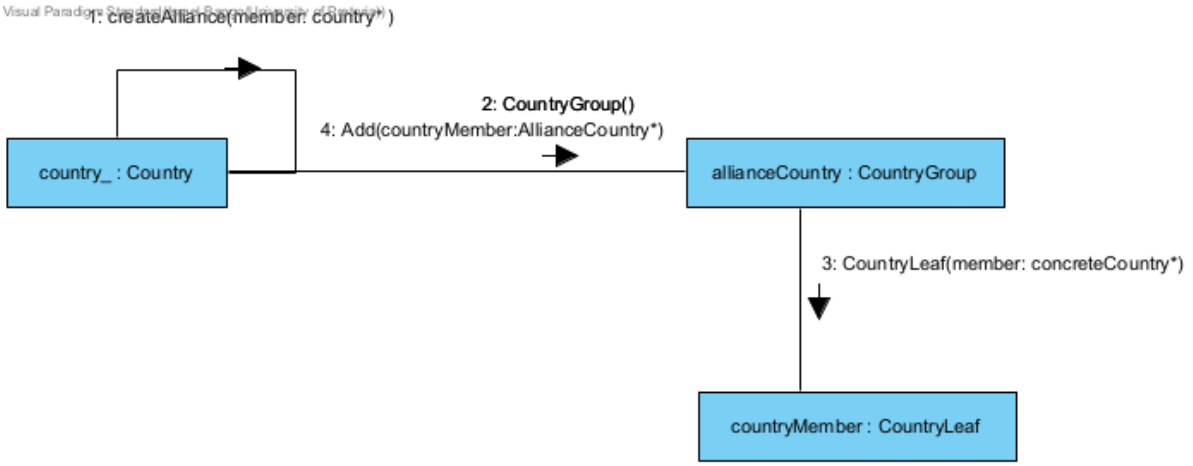

Figure: Communication diagram showing the creation and addition of a new alliance member

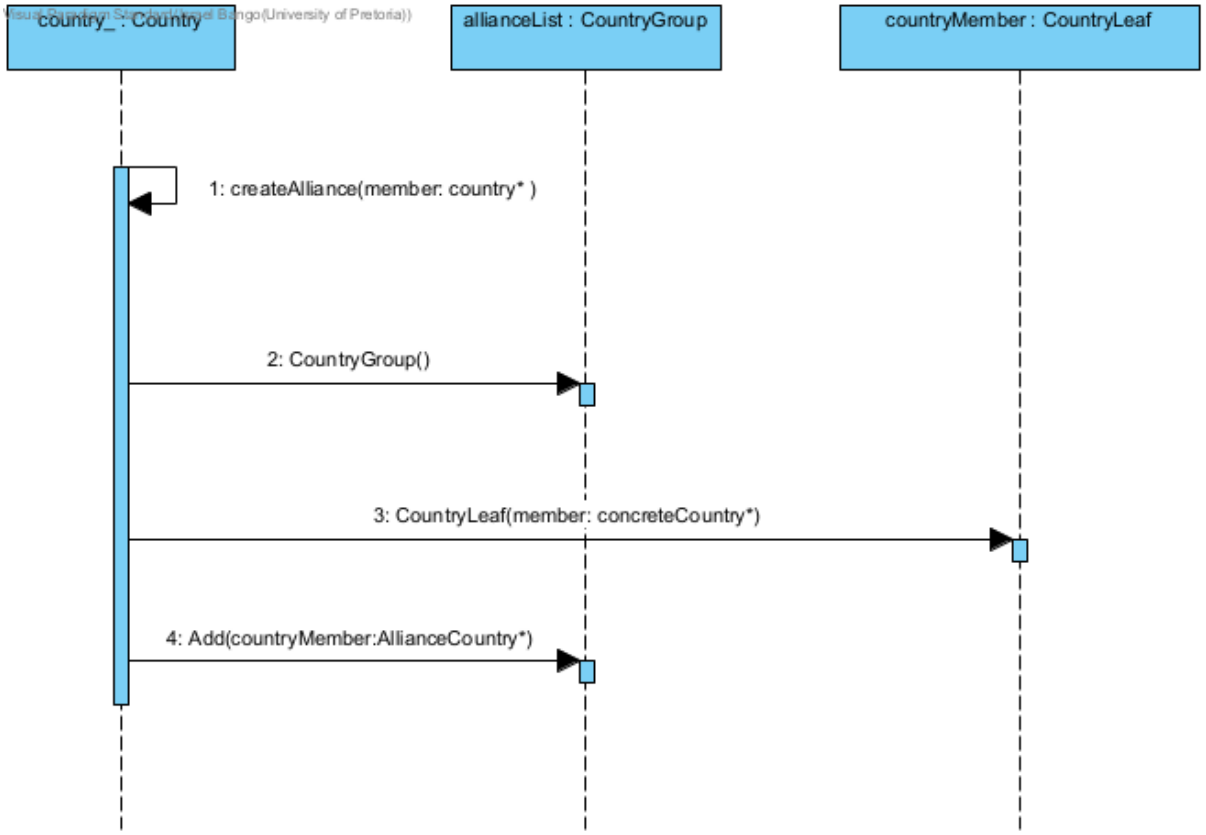

Figure: Sequence diagram of the Composite design pattern.

Chain of Command

Used to pass the phases of the war through different handlers and to facilitate transport, the Chain of Command pattern can be seen illustrated in the diagram included below:

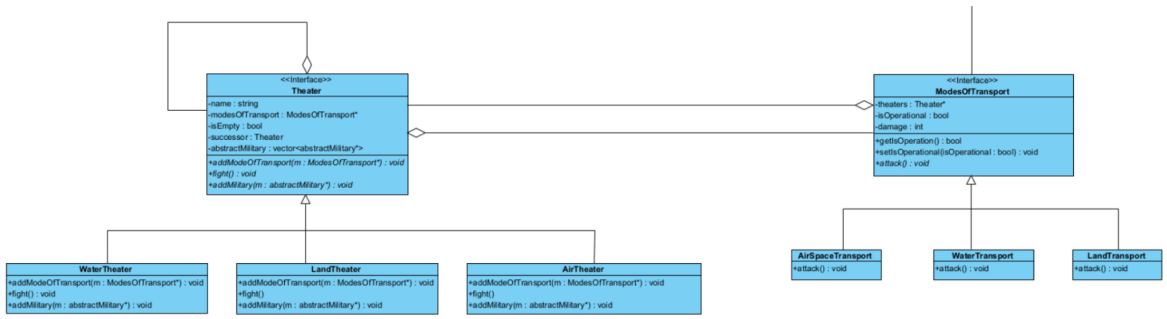

Figure: UML class diagram showing the chain of command pattern implemented in the context of the simulation software.

Facade

The facade pattern covers up the complex subsystems and presents the user with a simple interface from which they can run the entire simulation.

Further Applications

This section contains UML diagrams which illustrate more subsystems within the simulation.

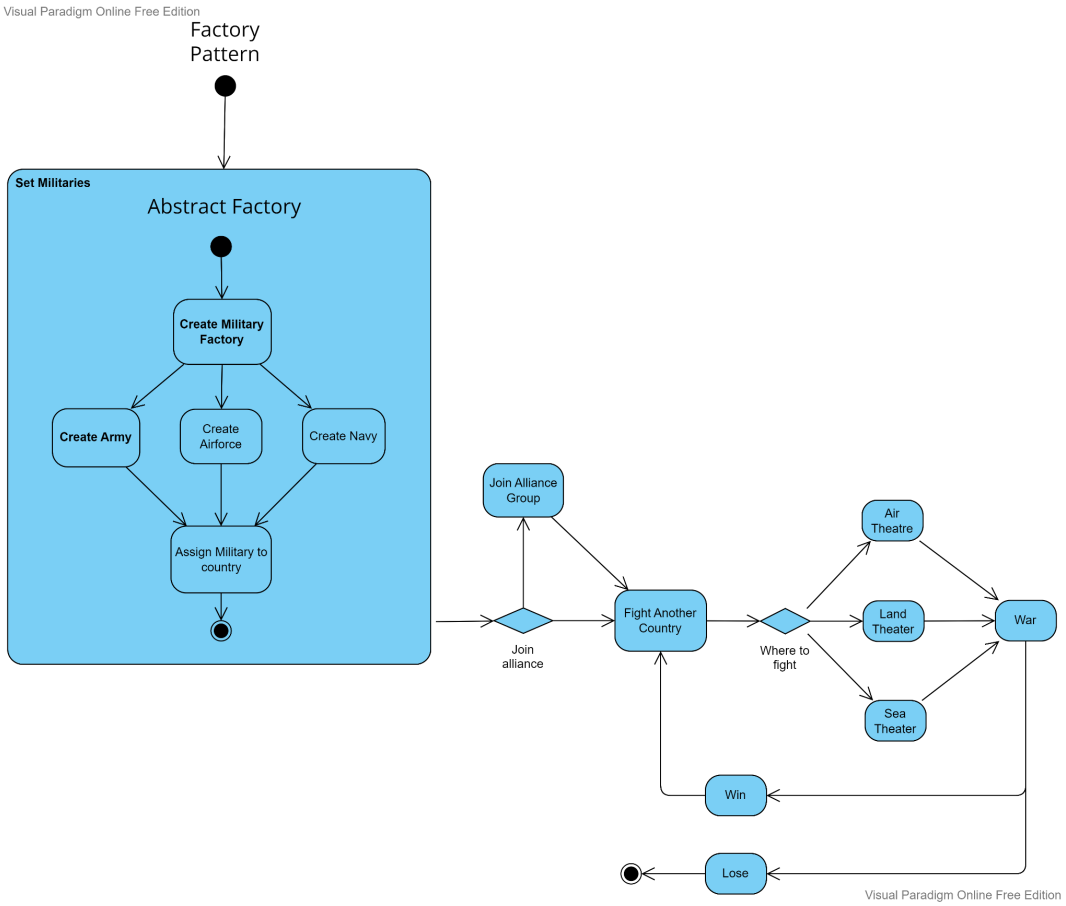

Figure: Activity diagram showing the creation of a country, the alliance subsystem, and the state and strategy patterns.

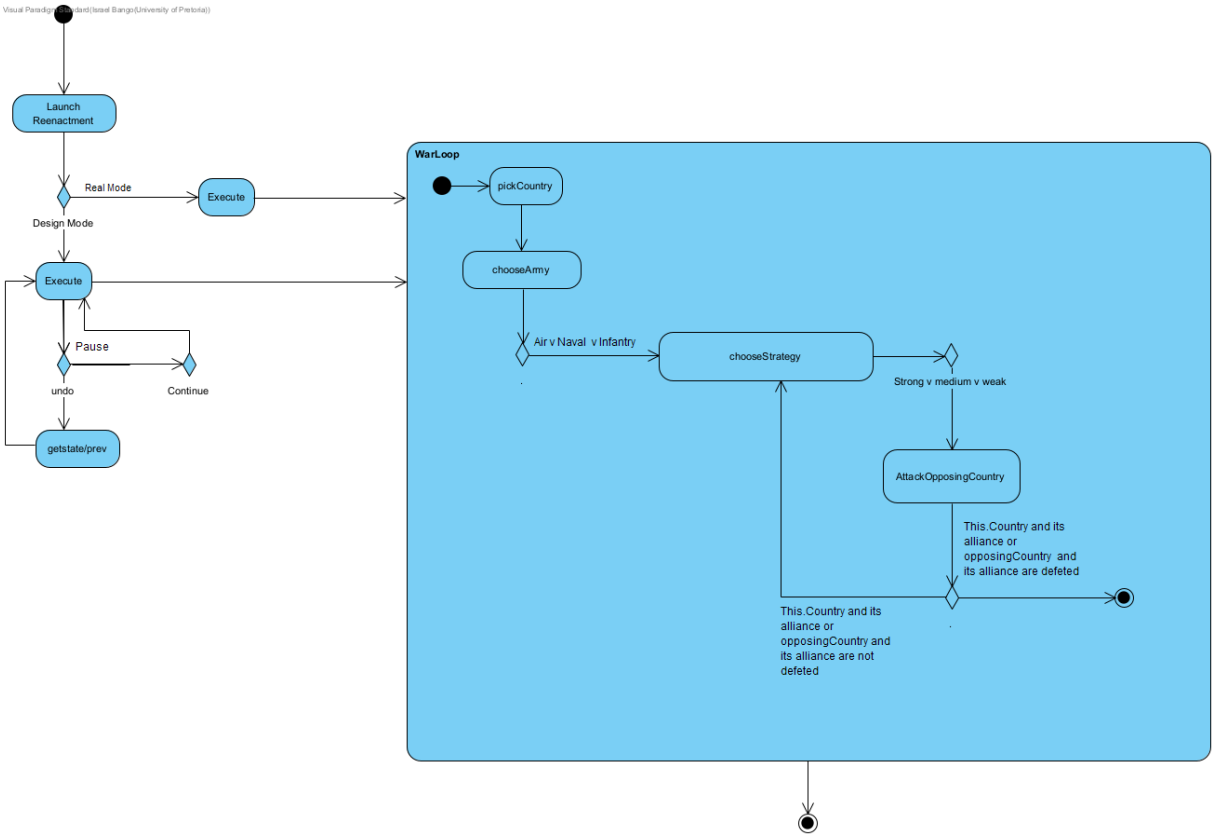

Figure: Activity diagram detailing the operation of the war execution loop.

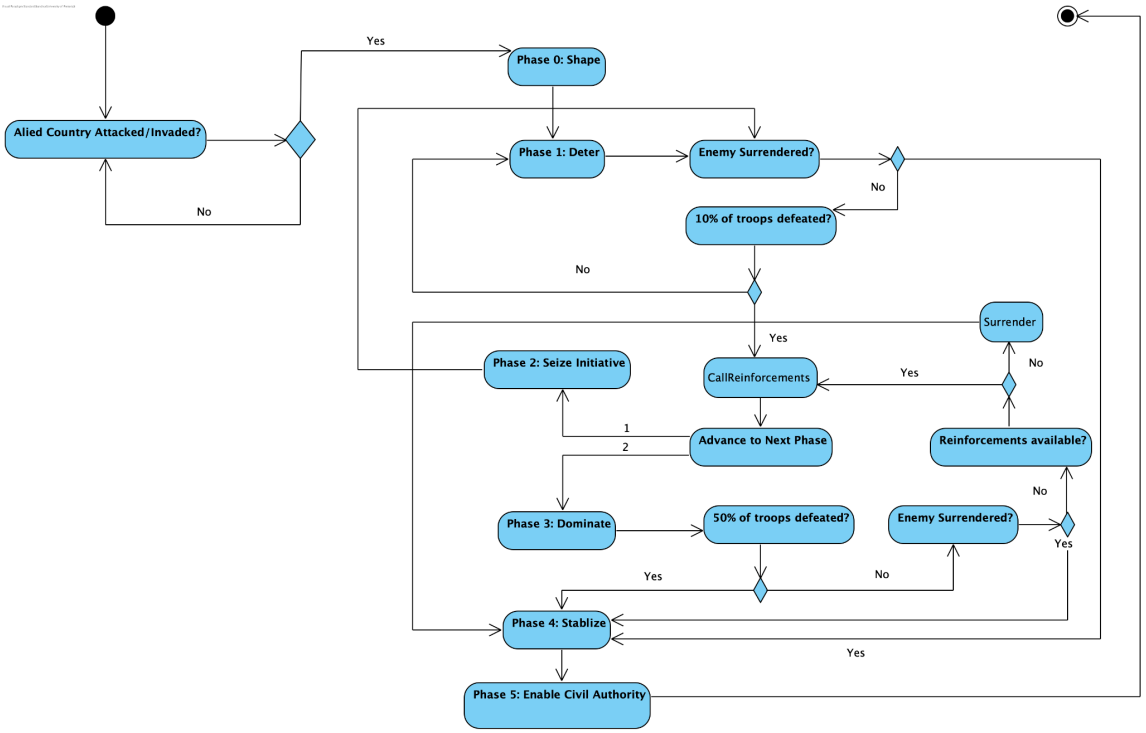

Figure: The various phases of war, first discussed in the research section, are shown within the context of the simulation here.

Links

Please refer to the following link to view the comprehensive system class diagram:
https://lucid.app/lucidchart/a7cc8ad8-fe4c-408c-945c-6d93ac6fb242/edit?page=HWEp-vi-RSFO&invitationId=inv_8e77e7c3-87cf-4233-ab70-e9a14a0c898e#

GitHub Link:
https://github.com/Jameson4/Optimal_Circuitry_COS_214_Project

References

Brown, I.M. (2018) *Transportation and Logistics*. International Encyclopaedia of the First World War. Available at: https://encyclopedia.1914-1918-online.net/article/transportation_and_logistics (Accessed: November 7, 2022).

Clausewitz, C.von and Graham, J.J. (2017) *On war*. Germany: Jazzybee Verlag.

Rummel, R.J. (1975) *Understanding conflict and war*. Sage Publications. Available at: https://www.hawaii.edu/powerkills/UCW.HTM (Accessed: November 7, 2022).

Scharre, P. (2016) *American strategy and the six phases of grief*, *AMERICAN STRATEGY AND THE SIX PHASES OF GRIEF*. War on the Rocks. Available at: https://warontherocks.com/2016/10/american-strategy-and-the-six-phases-of-grief/ (Accessed: November 7, 2022).