Simulating human civilisation and the evolution of society based on factors such as culture, religion, and technology

James Benjamin Brimelow Gorman, 100505349

College of Engineering & Technology, University of Derby

100505349@unimail.derby.ac.uk

**Abstract:** Fictional maps are often unrealistic in modern media. As an avenue for creating more realistic and believable maps, a simulation – named “Juris Civilis” – is designed and implemented as a way to create new maps using artificial intelligence and simulation. The project is able to meet most of its self-defined goals but notes a lot of variance from initial design plans in order to meet these goals and other technical requirements.

Contents

[1 Introduction & Goals 4](#_Toc101281664)

[1.1 Basis 4](#_Toc101281665)

[1.2 Development Cycle 4](#_Toc101281666)

[1.3 Measurement of Success 5](#_Toc101281667)

[1.4 Feasibility 6](#_Toc101281668)

[2 Literature Review – Algorithms & Topic Research 6](#_Toc101281669)

[2.1 Dwarf Fortress 6](#_Toc101281670)

[2.1.1 World Generation 7](#_Toc101281671)

[2.2 Europa Universalis 4 7](#_Toc101281672)

[2.2.1 Provinces 8](#_Toc101281673)

[2.2.2 Artificial Intelligence 8](#_Toc101281674)

[2.2.3 Random New World 9](#_Toc101281675)

[2.3 Sid Meier’s Civilization 5 9](#_Toc101281676)

[2.3.1 Artificial Intelligence 10](#_Toc101281677)

[2.4 Iron Age 10](#_Toc101281678)

[2.4.1 World Generation 10](#_Toc101281679)

[2.4.2 Simulation & Artificial Intelligence 11](#_Toc101281680)

[2.4.3 Tile and National Names 12](#_Toc101281681)

[2.5 GeaCron 12](#_Toc101281682)

[2.6 Guns, Germs and Steel 13](#_Toc101281683)

[3 Literature Review - Environment 13](#_Toc101281684)

[3.1 Language – C# 13](#_Toc101281685)

[3.2 Engine – Unity 13](#_Toc101281686)

[4 World Design & Methodology 14](#_Toc101281687)

[4.1 Project Background 14](#_Toc101281688)

[4.2 World Map 14](#_Toc101281689)

[4.2.1 World Generation 14](#_Toc101281690)

[4.2.2 Provinces 17](#_Toc101281691)

[4.3 World Properties 18](#_Toc101281692)

[5 World Implementation 19](#_Toc101281693)

[5.1 World Design 19](#_Toc101281694)

[5.1.1 Biomes 20](#_Toc101281695)

[5.2 World Properties Implementation 22](#_Toc101281696)

[5.2.1 Cultures & Names 23](#_Toc101281697)

[5.2.2 Population 24](#_Toc101281698)

[5.2.3 Saving 25](#_Toc101281699)

[6 Simulation Design & Methodology 26](#_Toc101281700)

[6.1 Empires 26](#_Toc101281701)

[6.1.1 Ruler 27](#_Toc101281702)

[6.1.2 Culture, Economy and Military 28](#_Toc101281703)

[6.1.3 Religion 29](#_Toc101281704)

[6.1.4 Technology 29](#_Toc101281705)

[6.1.5 Opinions 29](#_Toc101281706)

[6.1.6 Unrest & Rebellions 30](#_Toc101281707)

[6.2 Action Design 31](#_Toc101281708)

[6.2.1 Spawning & Colonies 31](#_Toc101281709)

[6.2.2 Technology Development & Learning 32](#_Toc101281710)

[6.2.3 Religion Spreading 32](#_Toc101281711)

[6.2.4 Opinion Changing 33](#_Toc101281712)

[6.2.5 War Declaration and Attacking 33](#_Toc101281713)

[6.2.6 Rebellion Spawning and Stirring Unrest 34](#_Toc101281714)

[6.2.7 Rebel Suppression 34](#_Toc101281715)

[6.2.8 Rebel Actions 35](#_Toc101281716)

[6.3 Artificial Intelligence Design 35](#_Toc101281717)

[6.3.1 Acting 36](#_Toc101281718)

[6.3.2 Provincial Value 36](#_Toc101281719)

[6.3.3 War Decisions 37](#_Toc101281720)

[7 Simulation Implementation & Balancing 38](#_Toc101281721)

[7.1 Console 38](#_Toc101281722)

[7.2 Representation 39](#_Toc101281723)

[7.2.1 Map Modes 39](#_Toc101281724)

[7.2.2 Province & Empire Viewer 40](#_Toc101281725)

[7.2.3 Notifications 41](#_Toc101281726)

[7.3 Implementation 41](#_Toc101281727)

[7.3.1 Empires & Acting 41](#_Toc101281728)

[7.3.2 Expansion 42](#_Toc101281729)

[7.3.3 Culture & Economy 42](#_Toc101281730)

[7.3.4 Military & War 43](#_Toc101281731)

[7.3.5 Religion 43](#_Toc101281732)

[7.3.6 Technology 44](#_Toc101281733)

[7.3.7 Opinions 45](#_Toc101281734)

[7.3.8 Unrest & Rebellion 46](#_Toc101281735)

[8 Testing 47](#_Toc101281736)

[9 Analysis & Comparison to Goals 48](#_Toc101281737)

[10 Conclusion & Future Developments 52](#_Toc101281738)

[11 Bibliography 53](#_Toc101281739)

1. Introduction & Goals

A map is a representation of the world viewed from an impartial lens. The results of conquests, atrocities, and political disputes - represented by a simple line on a page. This fascinating concept shows one simple truth – the world we live in is forged by our past, the lives of men long gone still alive by the world they shaped. Perhaps this is why fantasy maps often lack a sense of realism, with the borders of nations defined less by their past and more by what suits the medium it was designed for. This is understandable, the time and labour requirements needed to create a highly detailed and refined world often are superseded by other goals – but in the modern day of automation it is possible to forego these requirements by having the task conducted by a computer using a simulation.

This document details the production of a simulation designed to solve this exact task – create a world from scratch with national borders representative of the world and peoples who inhabit it. A believable and dynamic new world where factors ranging from simple geography to complex foreign relations are reflected in the evolution of civilisation. All this will culminate in the artefact, named “Juris Civilis”, solving the project hypothesis: “*Can a simulation be used to believably portray a new world history, using artificial representations of factors that have defined real history, and what algorithms can and should be used to achieve this objective?*”

* 1. Basis

The intention of this project is both to serve as an experiment as well as to create a tool that can be used for a variety of purposes. The subject of causation analysis in history is a small yet controversial field – and the development of a suitable simulation could be beneficial in both educating users in the topic area as well as, if developed to an outstanding degree, a way to test existing concepts in a controlled environment to ascertain the validity of certain theories and ideas.

Additionally, while not a direct goal of the project, the product of this endeavour could later be extended into a library for other projects – allowing developers to make use of the artefact as a way to simply develop their own worlds for projects. Otherwise, if not implemented as a library, the results of the simulation could then be used as a creative reference – with users able to take the produced map as a base for their own projects.

* 1. Development Cycle

The goals this artefact is attempting to achieve are complex – requiring the introduction of different fields of both mathematics and computer science to truly reach its potential. As such, the project will follow a rigid structure of development, comparable to the V-model of software development (Forsberg & Mooz, 1991). The following steps will occur sequentially:

* Literature review
  + Algorithms and similar projects
  + Topic area research
  + Development enviroment
* Objective 1 – world generation (mapping)
  + Design of world generation procedure
  + Implementation of world map generation procedure
  + Testing of world map generation
* Objective 2 – world generation (properties)
  + Design of world properties
  + Implementation of world properties
  + Implementation of world saving
  + Testing of properties generation
* Objective 3 – Simulation (actions)
  + Design of empires
  + Design of empire actions
  + Implementation of empires
  + Implementation of empire actions
  + Testing of actions
* Objective 4 – Simulation (artificial intelligence)
  + Design of artificial intelligence
  + Design of time progression
  + Implementation of time progression
  + Implementation of artificial intelligence
  + Testing of artificial intelligence
  + Readjustment where needed to ensure integrity of the simulation
* Technical testing
* Results analysis
  + Demonstration
* Comparison to hypothesis
  1. Measurement of Success

As the proposed hypothesis represents a subjective question, it is impossible to determine an exact answer to the question from any given product. To alleviate this, the success of the model against the hypothesis is defined by a list of attributes that the model should display, mirroring real world events and concepts. If the artefact can display more than half of the following attributes, it can be considered a successful implementation and proof of the hypothesis.

* Chronological accuracy. The results of the system should provide parallels against key eras of real history.
  + Dawn of civilisation - A number of large powers form while the majority of the world remains in a tribal or dispersed state
  + Bronze age collapse - Through any means, at least one large power should be overtaken or splinter into different states
  + Classical age – Small states, either remnants of a former great power or newly formed, should become more common, and should seek to unify provinces of their cultural groups.
  + Medieval period – Many cultures should be entirely owned by a nation representing their culture; wars should be frequent between states.
  + Renaissance – A large technology gap should form between different regions on the map. The majority of the map should now have nations on it. States with better technology than others may begin to subjugate others
  + Modernisation – The technology gap should begin to even out between nations. All land should eventually become occupied
* Impact of the world. Examples should be provided of certain key world characteristics influencing events.
  + Culture
  + Religion
  + Technology
  + Geography
* Believability. The decisions made over the course of the system should demonstrate believably human qualities. Some examples of the following will be provided in the documentation:
  + Mistakes – in which a nation is shown to have made a decision that has negatively impacted their situation.
  + Collapse – in which a nation is subjected to many negative events (such as war or revolt) in a short period of time – almost as if the people have recognised the weakness of the regime and have taken advantage of this state.
  + Rivalry – in which two or more nations are shown to consistently have low relations with one-another, and conflicts are frequent between the two subjects.
  1. Feasibility

The goals of this project are undeniably high, requiring extensive research and development time in order to meet the specified goals. Even then, due to the nature of simulation, it is entirely possible that the product may turn out wildly different than expected based on some minor flaws in parameters or theory. In order to combat these flaws, the work done on the project will be thorough and consistent – and testing will be conducted throughout in order to ensure at any given stage the project meets expectations.

1. Literature Review – Algorithms & Topic Research

The uniquity of this project comes in the form of its ability to put existing concepts such as world generation and time simulation together – software already exists to complete some of the objectives of this artefact, but little has been done to collate these ideas into one package. By examining these other projects, valuable insight can be gained into notable algorithms and ideas that can be of use during development. Additionally, as this artefact revolves heavily around ideas of real-world history, finding sources and academic works to provide reference for the importance of various factors constitutes invaluable reference for the properties of the simulation.

* 1. Dwarf Fortress

The first similar system identified was the 2006 game Dwarf Fortress (Tarn Adams, 2006). This game, which began development in 2002, is set in a world populated by fantasy creatures such as elves, dwarves, and goblins. The objective of the player is to build a new colony for their chosen dwarven civilisation, trade with nearby populations and accumulate wealth. The game is notorious for its difficulty and attention to detail, particularly in its simulation of the world.

The game world is generated by the player at the beginning of the game and uses an entirely random map fit with its own history and civilisations, as well as records such as that of key peoples and artefacts. This, much like the artefact under development for this project, is done from the beginning of history, starting with an empty map that slowly becomes populated as time progresses. The impact of this approach to simulation allows the world to be truly defined by what occurred in its past: wars fought between nations in the past may affect the player’s ability to trade with others, artefacts of ancient history may be found by the player and can be stolen by enemies, the titular dwarves can even be seen engraving depictions of historical events or peoples in some cases.

This historical simulation system also has a direct impact on gameplay, a testament to the depth of the implementation. Some interesting examples of how the timeline of the world can affect gameplay are as follows:

* When starting a game, the player may choose their civilisation and what resources they will bring to their new colony. The resources available are defined by what the chosen civilisation is capable of – such as what metals they have available or what animals the nation has successfully domesticated.
* It is possible for a player to find themselves in a world that is missing a race, including as the playable dwarves, as a result of past events such as wars.
* The world is able to undergo changes in “age”, as defined by what peoples and monsters exist in the world at any given time. It is even possible for the world to enter a state in which all fantasy elements, such as dragons or monsters, have become extinct, which is referred to by the game as “The Age of Civilisation” (Dwarf Fortress Wiki, n.d.)

While this project includes many features applicable to this project, it is important to note the simulation serves as a backdrop to the main gameplay. Many liberties are taken to ensure the world is playable – a world without dwarves will still allow a player to start a colony and will simply create new dwarves for the player to use. As such, while the game serves as an inspiration for this project, it has very little that can be used as material to work from. The following concepts highlighted are aspects which this project can gain insight from.

* + 1. World Generation

As described by the creator (Adams, 2009), the world generation method centres around elevation, with a location on the map being given a height and the sea level being defined as a result of the elevation of the world. This generated topographical map is then provided factors including temperature and rainfall – by which other factors such as river locations or biomes are built from. This approach to world generation is of particular interest, the concept of building a world by building layers of random content – elevation, temperature, rainfall – could be an appropriate method to be used by this project due to the relevance of these factors and their derivatives to the objective of the artefact.

Figure A world generated in dwarf fortress. In the centre, a dwarven civilisation can be seen, connected by various roads.

The elevation generation seen in Dwarf Fortress makes use of a variant to the diamond-square algorithm (Adams, 2013) (Alain Fournier, 1982), the result of this being results that appear to be more continent-like than other algorithms in this field. Additionally, the developer of the game noted that the game did not use the more common Perlin noise method for world generation, citing it as looking “fluffy and rounded” (Adams, 2010). This assertion has implications for the direction of this project, as the diamond-square algorithm may serve as a better fit for the project than the original intended algorithm – Perlin noise.

* 1. Europa Universalis 4

Europa Universalis 4 is a game developed by Paradox Development Studio (Paradox Development Studio, 2013), which was released in August of 2013. The game presents a map of the real world as it emerges from the end of the medieval era, with gameplay spanning from the renaissance to the end of the age of revolutions. Unlike the aforementioned Dwarf Fortress, Europa Universalis 4 (Henceforth referred to as Eu4) places more emphasis on realism, with the map the game takes place on closely reflecting that of the real world in the time period a player chooses to start from. Additionally, the mechanics of the game strongly influence the human and computer players to act how a ruler in that time would have. This can be seen very clearly when reviewing the history of a game, in which certain real events will almost inevitably occur if the player does not intervene, such as the formation of the Russian empire, the uniting of the British Isles and the expansion of the Ottoman empire. Eu4 presents an interesting study into the field of simulating real world history, and examination into this media yields some interesting results.

* + 1. Provinces

First and foremost, the basic gameplay elements present within the game should be noted. The map in Eu4 is split into different “Provinces” – Polygonal areas representing the borders of a location, often defined by the existence of cities and population centres within the region. These provinces are the foundation on which the game is played – the nations a player represents own a certain number of provinces which represent the borders of the country, and the objective of the game can be said to be to build a large empire by taking as many provinces as possible, either through war or colonisation.

In addition to this, these provinces have a number of properties tied to them which determine how they interact with the gameplay. Some notable examples are the following (Paradox Development Studio, n.d.):

* Provincial development (Split between Taxation, Production and Manpower). This represents the infrastructure present in the area, higher developments corresponding to bigger cities which provide the player with more money, trade power and manpower. The development of a province also determines the value of the location in peace negotiations, the cost to the player to build an administration to the region as well as how the international community reacts to the annexation.
* Terrain. The environment of a province determines aspects such as the speed at which military units are able to travel through a location, the amount of supplies that can reach unit as well as the cost of developing the province. This provides a double-edged sword type scenario – a mountainous terrain is more difficult for an enemy to traverse and survive in, but the player will incur additional penalties for attempting to improve the infrastructure in the region.
* Culture. Each province has a culture that represents the peoples within it and occupying nations will find additional difficulty in attempting to maintain control of provinces with cultures that they do not support. Additionally, the player and the artificial intelligence is rewarded in various ways for occupying locations with their own national culture, incentivising the formation of borders based upon cultural boundaries.

The provincial system presented by Eu4 may be an aspect to consider in the design of this project. Splitting the world up in this way allows for the mapping of populations and population centres onto the model and may provide some advantages in organising the artificial intelligence of the simulation. As the system presented in Eu4 is designed to work on a pre-determined map, some adjustments need to be made to fit the random nature of the artefact.

* + 1. Artificial Intelligence

The Artificial Intelligence of Eu4 is largely classified data belonging to the development studio, however some things about its inner workings are known. As previously mentioned, the computer players will often prioritise the uniting of their culture, but this is additionally modified by aspects such as relations with foreign nations, the difficulty of taking provinces and the decision making of the rulers in charge of each nation.

A good example of how relations impact the decisions made by the AI can be seen in interactions between Spain (Known in the timeframe of the game as Castille) and Portugal. Both Castille and Portugal belong to the same “Iberian” culture group, and yet conflicts between them are minimised. This is partially due to the opinion modifier each country has on the other, citing historical alliances, meaning the two often prioritise other conquests rather than turning against one another.

In terms of rulers, computer-controlled nations will be designated a ruler “personality”, which is changed when a leader abdicates or dies. These personalities impact the actions nations will take during that leader’s reign, with militaristic rulers focusing on conquest and military expansion, and administrative rulers working instead to improve their economic condition. This creates a believable feeling of an evolving diplomatic landscape – players may find that once passive nations will suddenly become aggressive as a militarist takes charge, or former threats become minimised by a ruler less interested in war.

Particularly due to its ability to constantly change the diplomatic landscape of the world, ruler-based artificial intelligence is to be considered when determining the simulated history aspect of the development cycle. This system allows for great rises to power as well as falls due to negligence on the part of the ruler, which has been seen throughout real-world history extensively.

* + 1. Random New World

The first purchasable update to Eu4, named “Conquest of Paradise”, introduced a feature referred to as “Random New World”, advertised as the ability to generate an entirely new continent as a replacement for the Americas. This meant players would have the ability to explore a random map and build a new colonial empire as they explored new terrain. Upon release, this particular feature was met with mixed reception – partially as a result of the algorithm used for the random new world generation.

The generation mapping makes use of set-piece based method – in which the newly generated map is built from a number of premade assets put together to make a map. This method has seen successful usage in games like Spelunky HD (Mossmouth, 2013) and The Binding of Isaac (McMillen & Himsl, 2011), in which it is appropriate for the medium, however this method reflects poorly in a geographic environment. The set pieces present in “Conquest of Paradise” are often individual islands, rather than the large landmass present in the real-world Americas, and veteran players will quickly come to recognise the set pieces as they appear – removing the charm of the pseudo-random geography. Additionally, as the sets are human made, they often reflect themes or shapes – such as the set piece that resembles the corporate logo of the Paradox Development Studio – which does not create a believable environment for the game to be played upon.

Unlike the other aspects mentioned in the discussion of this product, this random generation method should be avoided, as it does not suit the attempted goals of the random world generation feature of this project. As an extension to this, the use of any predetermined structures, such as the polygons of provinces, should be avoided unless necessary – as they represent many of the same issues discussed above.

* 1. Sid Meier’s Civilization 5

Another game that has some relevance to this project is the 2010 Firaxis game Sid Meier’s Civilization 5 (Firaxis, 2010). In this game, the player is prompted to choose a historical nation and leader, before being presented with a hexagonal world map for them to build a new nation upon. Other nations, led by civilisations and leaders that the player may have selected, exist on this map concurrently, and the nations of the world will compete to reach a number of arbitrary “Victory” conditions. This game is far more gameplay focused than titles like the previously mentioned Europa Universalis 4, forgoing much realism in favour of creating an entertaining product. This would largely make it inapplicable to this artefact, which focuses almost entirely on realism, aside from the inner workings of the AI within the game.

* + 1. Artificial Intelligence

Each leader in Sid Meier’s Civilization 5 has a number of integer values representing their focus towards different tasks – set on a scale of 1 to 10 (With one glaring exception) – this in turn influences the actions the civilization in question will take. For example – the nation of France, led by Napoleon Bonaparte, has a “Loyalty” score of 3 – representing the real-life actions of the man, who was notable for violating various treaties and acting aggressively towards former allies. These values in turn then provide weighting to the decisions of the computer players, allowing them to act in pseudo-random ways which reflect the personalities assigned to them. This system allows for the nations of the world to act as if they have their own personalities and goals, and the actions of certain leaders can be predicted based on these hidden values.

This works as an extended implementation of the “personality” system present in Eu4. Whereas the Eu4 system works to provide a ruler a focus in areas such as diplomacy or economy, the Civilization approach instead provides weighting to specific areas – allowing for the characters within the game to have more of a “personality”. This could see use in the artefact described in this document, with random rulers receiving random values that weight decision making. This, in addition, could permit the rulers to make mistakes – a ruler with a high weighting towards war could start conflicts with larger nations, potentially having significant consequences on the nation as a whole. The ability for a leader to make mistakes is one that should see implementation in the simulation, as misguided decisions have changed history countless times.

* 1. Iron Age

Iron Age (Gorman, 2018) was the working title of a previous attempt at a concept similar to that of this artefact conducted by myself in 2018. The software produced as a result of this endeavour was of amateur quality and did not meet the technical expectations it had set out to achieve, but yet managed to display two aspects relevant to the current artefact – World generation and time simulation. Despite the flaws of this project, some valuable research can be gained from investigation into the inner workings of this system.

* + 1. World Generation

Throughout development, Iron Age underwent no less than three different implementations of world generation algorithms – each uniquely designed to fit the project.

The first, referred to internally as the “Snake method”, involved a tile-based approach, in which a random index would be selected in a two-dimensional array as a “Snake”. This “Snake” block would then expand to an adjacent index, which would then repeat the process, expanding to another index location. Once this process was repeated a number of times defined by the parameters of the generator, each existing “Snake” block would generate 3 “land” tiles in each orthogonal direction, before becoming a “land” tile itself. This method allowed for the generation of long continents, as well as permitting the generation of features like inland seas and coves. However, in the end this method was not used due to its unpredictability and tendency to develop unrealistic geography.

The second method, the “Fill method”, was developed and rejected in less than a day. The concept was simple, populate a two-dimensional array with a number of tiles in random indexes with a number representing their elevation, then populate all tiles in a 3x3 area around said tile with a random modification to the elevation of its creator. Eventually this would produce a mountainous terrain map which would then be designated as sea level below a certain elevation, corresponding to the roughly 70:30 ratio of sea to land seen in real life. This system, while similar to those in use in similar projects, was not implemented in a way that benefitted from the strengths of elevation-based terrain – the nature of the expanding elevations meant that the landmasses produced were often of the same shape, and constructs like islands or inland seas simply did not appear.

Finally, the implemented “Continental method” was inspired by the real-world separation of Pangea into the modern continents. The model would produce a circle with a radius almost equivalent to the height of the map, which would be entirely filled with land tiles. The outermost tiles of the circle would then become “Active” tiles, which could move anywhere within a 3x3 area – switching the specified index to an active tile and replacing the previous tile with an ocean tile. These active tiles would then propagate simultaneously, creating random lines in the land, until they reached a point where there were no possible land locations to “jump” to in the 3x3 Area. Once all “Active” tiles had been removed, a flood fill algorithm would be utilised to identify each landmass and place it randomly in the ocean. This algorithm produced favourable results, producing random continents with believable proportions.

A computer screen capture

Description automatically generated with low confidence

Figure An example of an archipelago generated by the Iron Age project

For this artefact, none of these methods should be used, as despite the relative success of the “Continental method” there were numerous flaws that could be alleviated by the use of other algorithms. Additionally, the continental method as implemented in Iron Age was particularly inefficient, using far more time and processing power than would typically be necessary for a world generation algorithm.

* + 1. Simulation & Artificial Intelligence

As discussed, Iron Age attempted to meet many of the same objectives as outlined in this artefact, including the development of nations across time. The simulation aspect of Iron Age was flawed in many ways, first and foremost in its scalability, a mistake which cannot be repeated in this artefact.

Iron Age implemented a system in which all nations generated a certain amount of “military units” per season, the amount of which would be modified by the technology of the owning civilisation, the number of tiles owned by the civilisation and the number of resources in the capital of the owning civilisation. These units would continuously generate as time progressed and would only be expended when the computer players attempted to take land from another nation. This resulted in an inevitable overflow in which a nation would exceed the bounds of the integer data allocated as their military size – an issue that was compounded by the already extensive use of memory by the software.

Additionally, the artificial intelligence implemented for Iron Age was very simplistic, and the only concerns of nations revolved around taking more land – with little regard for any properties of the land in question. This was sufficient for the small scale of Iron Age and its usage as an experiment into simulation but prevented the system from depicting a believable world. Additionally, the nations of the world fit a rigid decision-making structure – only declaring war if they had a numerical advantage over their competitor of above a certain constant value. This prevented the nations from taking risks, or developing any real form of personality, an aspect which will be emphasized in the development of this project.

* + 1. Tile and National Names

As a final note on Iron Age, there is one method that may see recreation for the purposes of this artefact, notably in the naming system for locations. Upon generating a world, the user would be prompted to select a method of name generation – either by random selection or random generation. Upon selecting random generation, the system would create a series of new location names based on existing names of cities.

This was done by first randomly selecting an index within the dataset containing the names of various cities, the first number of characters from this name would then be stored in memory. The algorithm would then find another random index, and compare the last stored character to the character in the same position in the character array (For example, if the memory stored the words “Lond”, from “London”, a comparison against the city of “Leeds” would return true, as the fourth character is shared), after which the mechanism would pull a number of characters from this word to append to the original memory (In the listed example, this would produce “Londs”). This process would continue until the system reached a blank character or no names that fit the condition were found (after an iteration through the source material starting from the first failed comparison). The produced name would then be stored with an index representing the location it applied to.

While this system had some flaws, the results given were often believable location names that fit the simulation well, as such it is possible that this algorithm may be reworked to fit this artefact – though with better efficiency measures and potential quality checking systems implemented alongside it.

* 1. GeaCron

While not a simulation or game, the GeaCron project (geacron.com, 2011) provides significant insight into world history by providing a national map spanning from 3000 BC to the modern day, at yearly intervals. Additionally, the premium edition of the product displays analytics that provide valuable insight into the reasons for border changes – while providing sources where applicable. This resource presents a much-needed insight into the historical aspects that will be simulated within the artefact, as well as references for how long aspects such as wars should last in the simulation.

One notable aspect that can be inferred from the map is the importance of water access in the development of early civilisations – all ancient nations can be seen developing in regions with river or ocean access (most often, both) and often expanding in the direction of regions that satisfy these conditions. Another observation that can be made is that instability of nations correlates negatively as time progresses – with early civilisations often collapsing multiple times across their reign, especially in the case of vast empires.

* 1. Guns, Germs and Steel

One significantly enlightening source for the simulation comes in the form of the book “Guns, Germs and Steel” (Diamond, 1997a). The non-fiction text takes a critical stance on racial-superiority theories of world history – specifically in relation to why some societies were able to dominate others. Diamond asserts his belief, known as environmental determinism, that the scenery of populations determined their future.

Early in the document (Diamond, 1997b), the author refers to the idea that regions more suitable for cultivation allowed for centralised communities, in which less citizens were required for producing food. This is in contrast to hunter-gatherer societies in which individuals were required to find their own food, and therefore had to be more dispersed. In the former populations, the lesser need for food-producing citizens allowed for more specialisation – meaning these societies could afford for some residents to specialise into other fields and thereby develop new technology and ideas. This source demonstrates a clear need for the simulation of access to food.

1. Literature Review - Environment

As a software development project, one key aspect unrelated to the topic area itself is the discussion of what development environment is most suitable for the artefact. This primarily includes two aspects – language and engine. Owing to prior knowledge, each of these items had been predetermined before this literature review had commenced, and therefore much of this segment will serve as justification for the language and engine used.

* 1. Language – C#

C# is a programming language developed by Microsoft and released in the year 2000. Allegedly intended as a competitor to the then-monopolising java language, the owners of which Microsoft was involved in a legal dispute with at the time (SUN MICROSYSTEMS INC v. MICROSOFT CORPORATION, 1999), C# quickly formed into its own entity as part of the .NET framework (Microsoft, 2000). Today, C# is used for a variety of fields due to its nature as a compiled language and incorporation of intuitive syntax – allowing for quick development of products and ability to release said products in an executable format.

The compilation-based nature of C# is one of the largest justifications for the use of C# in the artefact. ‘Juris-Civilis’ is large in scope – by nature it will require significant processing power and optimisation to function as intended. By building the project into an executable via compilation, much of this overhead processing cost can be minimised – practically a necessity of a project of this size and scope. When accounting for the support for compilation, ease of use and prior experience with the language, C# is justifiably appropriate for the purposes of this software.

* 1. Engine – Unity

As formerly mentioned, experiments into this subject have been conducted before – and they highlighted one significant detail – the need for an engine. These former projects made use of .NET framework windows forms (Microsoft, 2021) – which quickly proved insufficient for its needs, especially with regards to graphical display. As part of the time-based nature of the simulation, frequent graphical updates are needed to accurately display information, and the use of simple images that update individual pixels manually each frame chafed against this requirement. Compounded with the need to develop hit detection within the map in place of traditional buttons, the environment proved itself unsuited for the task.

Learning from prior mistakes, this project will make use of a more appropriate environment – the Unity engine (Unity Technologies, 2005). Designed primarily as a game development environment, Unity provides support for many of the problems encountered previously – support for constant graphical updates and buttons, as well as more complex constructs like cameras that were not implemented in the former project.

In addition to this, unity incorporates a vital component – meshes. Meshes (Unity Technologies, n.d.) are a polygonal structure capable of quick rendering and hit detection – a perfect match for the provincial system the project will incorporate. By using these meshes, graphical updates can be localised entirely on areas that need updating – rather than updating the entire interface each frame, individual polygons can be updated when needed – an invaluable boon for the performance of the simulation.

1. World Design & Methodology

As a result of the collated information and comparison in the literature review, the following design plans have been developed to specify how the system itself will function – in particular what features will exist and what algorithms will be used where appropriate. It should be noted that this design segment is subject to change during development, as direct exposure to the impacts of each decision may demonstrate unforeseen consequences necessitating a change in direction from these plans.

* 1. Project Background

To reaffirm that which was discussed in the environment literature review – the system will use C# and the unity engine throughout development. As an amendment, the C# LINQ library (Microsoft, 2021) will also see frequent use as a way to query system data quickly and simply. Said library includes significant improvements to the initial C# lambda functions and querying capabilities, as well as its own pseudo-language similar to SQL designed for dataset management. Some form of dataset management is essential for a project of this scope, and the LINQ library fits the system requirements outright.

* 1. World Map
     1. World Generation

The intended algorithm for the world generation was originally the Perlin noise method, but due to the results of the literature review the “Diamond-square algorithm” has been identified as a better fit for the goal of generating a landscape – and will therefore take the place of Perlin noise in this area of the system. This is not to say that Perlin noise will see no use in the project, as review into systems like dwarf fortress have identified the necessity of different “layers” of randomly generated data – it is not enough to simply create a landscape. The following “layers” will need to be produced in any single use of the map generator:

|  |  |  |
| --- | --- | --- |
| **Layer** | **Layer Name** | **Algorithm** |
| 0 | Elevation | Diamond Square |
| 1 | Temperature | Random gradient from the equator |
| 2 | Rainfall | Perlin Noise |
| 3 | Flora | Perlin Noise |

These layers are directly inspired by the prior literature review – especially that of the rainfall and flora fields which were included due to insights provided by the GeaCron project and the book Guns, Germs and Steel[[1]](#footnote-1). These factors in turn should make the following adjustments and additional values:

|  |  |  |  |
| --- | --- | --- | --- |
| **Order** | **Name** | **Relevant Factors** | **Description** |
| 0 | Land/Sea Adjustments | Elevation | All values over the 70th percentile of elevation will be defined as land tiles, while all below will be defined as sea. All values relating to the sea locations will be discarded, and any following operations will only occur on the land tiles. |
| 1 | Mountain Adjustment | Elevation, Temperature | The top 20% of elevation values will have their temperature values lowered proportional to their deviation from the 4th quintile elevation. |
| 2 | Forestry | Flora, Rainfall | Locations are given a forestry level based on the overlap between above-average flora values and above-average rainfall values. |

All these collated factors will then be used to specify a biome type for each pixel, using a system by which the locations “fit” value to a biome type is a real number between 0 and 1 - calculated based on the weighting each biome assigns to a specific property. While exact values for weighting are not yet set, the rough impact each property will have on the “fit” value can be seen in the table below. For each property, the difference from the median value for the field is taken as the input and multiplied by the weight (specified here as either high or low) – therefore a negative difference will count as the opposite for the weighting (for example, a biome with negative forestry rating will return a negative number for a highly forested region, but will produce a positive value for a less-forested-than-average region)

|  |  |  |
| --- | --- | --- |
| **Biome** | **Value Weighting** | **Colour** |
| Temperate Forest | Forestry (High Positive Weight)  Temperature (Low Negative Weight) | Dark Green |
| Tropical Forest | Forestry (High Positive Weight)  Rainfall (Low Positive Weight)  Temperature (High Positive Weight) | Very Dark Green |
| Taiga | Forestry (High Positive Weight) Temperature (High Negative Weight) | Light Green |
| Grasslands | Forestry (Low Negative Weight)  Rainfall (High Positive Weight)  Elevation (Low Negative Weight) | Green |
| Savannah | Forestry (Low Negative Weight)  Temperature (Low Positive Weight)  Elevation (Low Negative Weight) | Orange |
| Tundra | Forestry (High Negative Weight)  Temperature (High Negative Weight) | Very Light Green |
| Desert | Forestry (High Negative Weight) Rainfall (High Negative Weight)  Temperature (High Positive Weight) | Yellow |
| Mountain Range | Elevation (High Positive Weight)  Forestry (High Negative Weight) | Grey |
| Forested Plateau | Elevation (High Positive Weight)  Forestry (High Positive Weight) | Green-Grey |
| Shrubland Plateau | Elevation (High Positive Weight)  Temperature (High Positive Weight)  Forestry (Low Positive Weight) | Orange-Grey |

These biomes and their weighting-based structure are derivative of the works of Leslie Holdridge and Samuel Charles Kendeigh. In the writings of the former (Holdridge, 1964), the author introduces the idea of “life zones” – classifications for areas of the world based on the rainfall, humidity, and temperature of the region – factors which parallel the rainfall, forestry, and temperature properties previously defined. As seen in the attached figure, the exact classifications of a “life zone” are dependent on the weight of each of the properties – inspiring the described system to be implemented in the artefact.

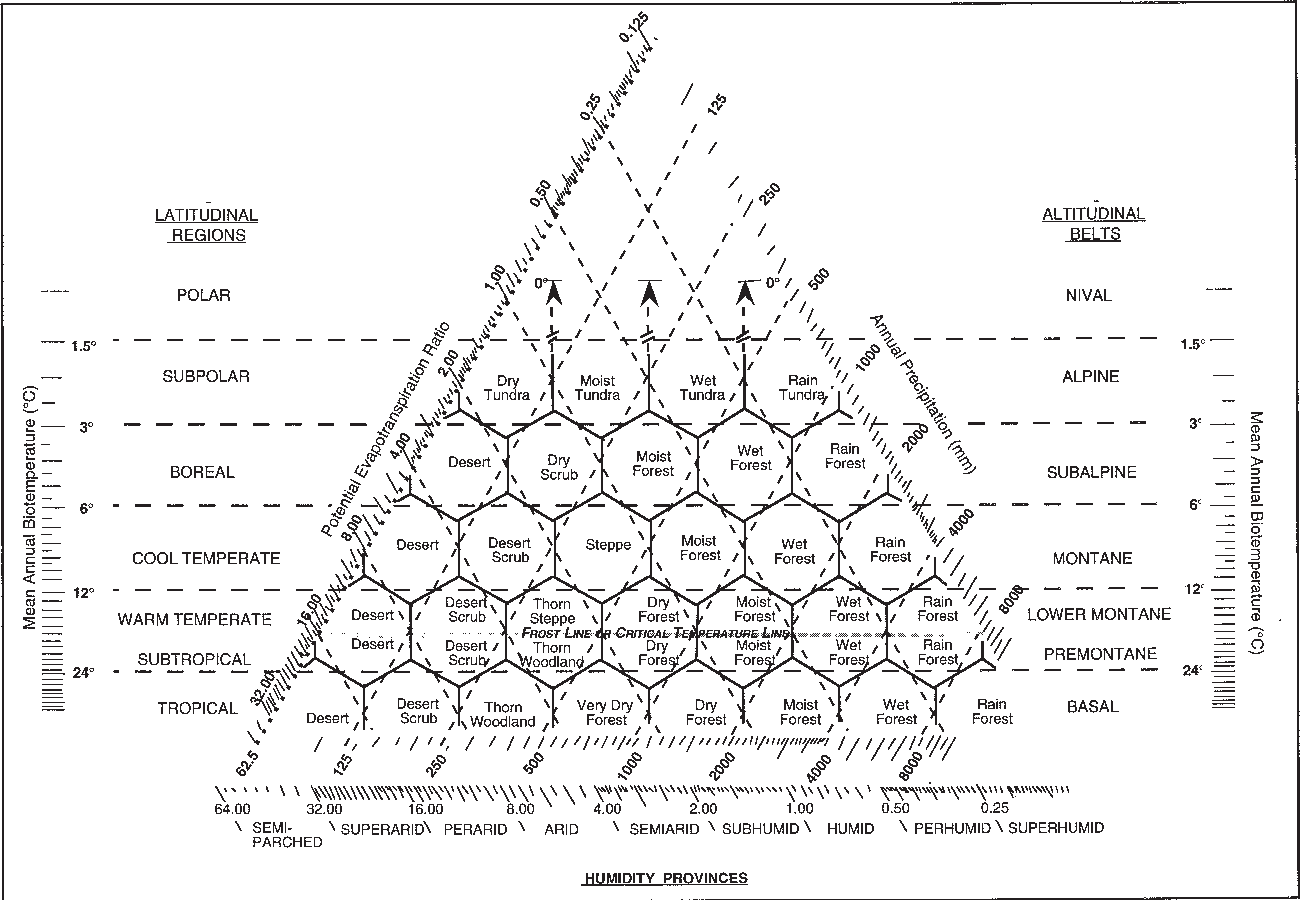


Figure The Holdridge life zones chart (Lugo, et al., 1999)

Kendeigh’s work (Kendeigh, 1961) builds on the ideas presented by Holdridge to reduce the number of discreet “biomes” down to a plethora of more inclusive classifications. In his writings, the author identifies nine terrestrial biomes: Temperate deciduous forest, Woodland, Coniferous forest, Chaparral, Tundra, Desert, Grassland, Savanna, Tropical forest. These items serve as the main inspiration for the described biomes – though with some modification to better diversify the world.

These works and the biome system as a whole were chosen as the basis of the environmental system for two primary reasons. Firstly, the biome model of environmental classification is simplistic but retains a lot of value – by describing a location as a desert, a user does not require significant knowledge to understand the environment, and yet the qualifying properties of a desert environment are sufficient to build further simulation from. Secondly, the classification system lends well to a mathematical environment – the Holdridge model is based on quantifiable properties, rather than opinions or speculation, and therefore can be implemented into the simulated environment with little variance or difficulty.

* + 1. Provinces

Once all the geographical information is produced, the world generation algorithm can move towards the next stage of human history – the dispersion of humanity across the world. In this stage, the existing map will be divided into a number of polygons – each representing the geographical boundaries of a population of people, the rough borders at which the residents of a location live their lives.  
  
The first stage of this process is to divide the map into a number of equally sized squares, each of these squares that contain land is then split into two triangles, randomly selecting the direction of the division (from top left to bottom right or from top right to bottom left). Each of these locations is referred to as a “chunk” and represents a fraction of a province. Each chunk will then store the following information based on the average values of its containing pixels: Elevation, Temperature, Rainfall, Forestry and Biome.

When all the chunks have generated, a random chunk will be randomly selected from the set of all chunks to form the first component chunk of a “province” – the base measurement of land to be used by the simulation. The first chunk in the province will return its biome which will be polled for its “spread” value – a rough indication of the population density of the region (biomes with more dispersed populations, such as those of tundra or desert regions, have a high spread value. Those with more concentrated populations, like those of forested regions, have lesser spread values).

Upon receiving the spread value as dictated by the province, the province will then proceed to spread across multiple chunks – choosing any chunk with two or more connecting vertices to be added to its components (upon which the adjacent chunks to the new component chunk will be accessible by the province) and repeating this process for as many times as the spread value allows. A province will choose the next chunk to be appended to its components set by randomly selecting a chunk from a compiled list of all valid chunks, with the following rules determining what can be considered a valid chunk:

* Only chunks with two or more connecting vertices are able to be added as a component chunk. When the set of all adjacent chunks is compiled, the software will check for any repeated chunks within the set. If the set contains multiple of the same chunk, specifying that it has multiple adjacencies, all chunks with less instances than the highest number of repeated chunks will be discarded.
* Provinces that start in an ocean biome (dictated by having no non-ocean tiles in the chunk boundaries) may only claim adjacent chunks of other ocean biomes. Similarly, land biomes may not claim ocean biome chunks.
* Any chunk appended to a province (either the first chunk or any connected chunks) may not belong to any previously generated province – preventing any overlaps in claims.
* A province with no valid adjacent chunks will cease appending new members to the set of component chunks regardless of the remaining spread value.

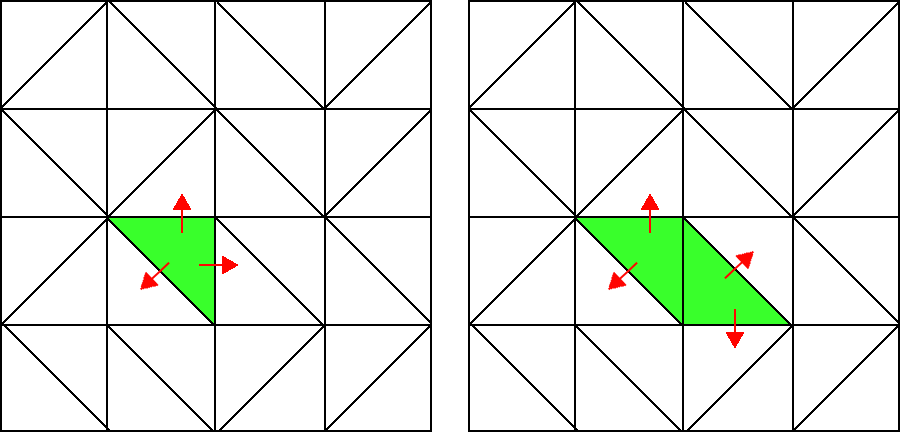
The generation of new provinces will proceed until all chunks generated are a member of a single province. The process of generating provinces allows for the world to be split into polygons according with size dependent on their biome – creating an environment split between the populations that live upon it.

Figure The chunk spreading method, in which one chunk spreads randomly to another, giving it access to more chunks as it proceeds.

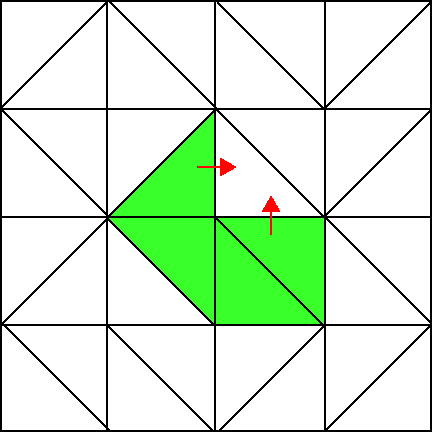
* 1. World Properties

Figure Chunk spreading is limited when there is a chunk with more than one adjacency

As mentioned in the prior segment – each location on a map will be sorted into different “provinces” which are the basic “unit of measurement” upon the map. A province represents not only the borders of a location but the borders of the people that inhabit it – and what conditions they live in. Like the chunks and the individual tiles on the map, each province will contain its own list of properties that describe it – first and foremost the biome defining properties (Elevation, Temperature, Rainfall and Flora) as well as the biome itself.

Each province will also require its own unique name to serve as a descriptor of the lands. This name generation algorithm will use the algorithm described in the literature review for “Iron age”[[2]](#footnote-2), with variations to improve performance and fit towards the project. This means the project will need to implement a dataset of numerous city names to allow for this property generation. Unlike the referenced software however, this procedure will be expanded to provide the ability to generate names for other constructs, such as oceans. In addition, the performance of this algorithm will be amended by rebuilding the algorithm from the ground up, using the concept rather than the code to allow for modifications to be made which produce a better performing product.

Additionally, each location will require its own “Culture”, a property that will be shared across a large variety of different provinces. A culture refers to a group with similar principles and ideas, effectively representing a “block” of people. Provinces in a similar area should share the same culture, and each culture should be provided a name to identify the people within, much like how the name of a location is generated.

Finally, each province should contain a property defining the starting population metric within its borders – using the previously generated parameters to do so. These population metrics will specify the size of the capital city within each province, either as a “Village”, a “Town”, or a “City”. This is an important step as it serves as a model for the population density of a region – how dispersed or centralised a set of people are. This factor is arguably the most vital of all the simulated properties – as it will determine both where civilisations may develop as well as how much value is ascribed to a region.

1. World Implementation
   1. World Design

The world generation of the artefact met most of the goals outlined for the project within the allocated deadlines but presented some unexpected challenges throughout development which necessitated changes in direction and procedure to be made. While most of the implemented features remained faithful to the original designs, some (with much deliberation) were modified to better fit the technical and creative requirements of the system.

First and foremost, the previously undefined parameters of the system – particularly the size of the map were determined. After testing, the best fitting width and height were determined as 6000x4000 (in pixels) – creating an image with a height to width ratio that both resembled that of a Mercator projection map as well as giving space to the map generator to operate effectively. While implementing all subsequent procedures, special consideration was made towards keeping all operations designed for different map sizes – with the intention of potentially permitting different map sizes and ratios in the future.

For the generation of continents, the originally specified diamond-square algorithm was discarded – replaced by a Perlin noise algorithm making use of octaves to generate jagged terrain. This algorithm, when compared with the previously mentioned method, demonstrated a better fit towards the direction of the artefact. Due to this change, review was made into the relevance of other described algorithms within the design plans – with little changing in aspects other than that of the temperature generation. While originally intended to use purely a curved line algorithm, the implemented system uses a mixture of a Perlin noise generator as well as a gradient function from a randomly generated line across the centre of the map. This change better reflects the variance in heat across the world, which is not a consistent line but rather a more varied mesh.

When the first results of generating terrain were made it became apparent that the noise generation procedures were liable to produce continents across the map borders. In most modern maps, the map is orientated in a way that cuts the map borders through the sea, rather than through land – this change allows maps to be more easily readable, as consideration does not often need to be made into which side of the world a location falls under. This meant that for the map to properly resemble a Mercator projection map, either the map had to orient itself in a way that cut through the seas, or the map had to generate in a way that did not step into map border territories.

To fix this issue, a new algorithm was produced to reduce the intensity of generated properties when they approached a certain boundary of the map. The values of properties within a certain distance from a map boundary were divided by greater numbers as they approached the border – allowing for a sloped decline in values. While this procedure was originally only designed for the elevation, it also found use in generating the polar regions of the map, with the temperature of a location being rapidly decreased if it fell within designated polar regions.

One unexpected trial of development came in the form of memory usage – which first appeared after the implementation of the four primary world generation factors (elevation, temperature, rainfall, flora). In accordance with the original design plans, each pixel of the map stored its own integer values for the four properties. With the map size of 6000x4000, this resulted in a set of twenty-four million elements storing 16 bytes of memory each. Additionally, the next step was to be the compilation of four sets containing a sorted list of each property to determine the decile values for each property. This procedure took too much memory to process, and therefore was modified in two ways; Firstly, the pixel class was changed to store only enumerators referring to the relative position of each property in relation to the set (in essence, instead of storing elevation as a number, it stored the value as “Low” or “High”). The enumerators generated by this procedure are as follows:

|  |  |
| --- | --- |
| **Property** | **Enumerator / Decile** |
| Elevation | |  |  | | --- | --- | | Low | Less than 6th decile | | Medium | Between 6th decile and 9th decile | | High | Greater than 9th decile | |
| Temperature | |  |  | | --- | --- | | Low | Less than 3rd decile | | Medium | Between 3rd and 7th decile | | High | Greater than 7th decile | |
| Rainfall | |  |  | | --- | --- | | Low | Less than 4th decile | | High | Greater than 4th decile | |
| Flora | |  |  | | --- | --- | | Low | Less than 5th decile | | High | Greater than 5th decile | |

Secondly, the decile calculation was changed from using a median based procedure to using a range-based formula, removing the need for the sorting of the properties lists entirely. The new function used the following calculation to determine the rough value for each decile.

These changes, while making the results of the generation algorithm less accurate and refined in places, consistently cut down memory usage from the software by a fourth – a huge change that made up for the minute decrease in quality as a result of the changes. In addition to this, the described property adjustments[[3]](#footnote-3) were cut from the function entirely – as the switch from exact integer values towards enumerators demonstrated a lack of need for these changes – as a result of this, the forestry property was cut entirely, being replaced by the previously generated flora values.

* + 1. Biomes

As these changes reflected large modifications to the design, the biome system, previously dependent on the replaced integer system, was overhauled too. These changes were mostly to better fit the new enumerator variables into the procedure, but these changes largely made the map produce better results overall. In the new system, biomes each had a “expected” enumerator value for each property, as well as an integer value which scored their “fit” for each biome. For each matching value, the “fit” score would increment by one. The highest scoring biome would be applied to the pixel. Two exceptions existed for this method however – the elevation of a province applied for ten “fit” points rather than one – ensuring ocean biomes and mountainous regions would be set appropriately – and each biome may have an enumerator value of “NA” which would automatically apply half-a-point of “fit” score to the biome score. The following table describes the newly formed biome properties.

|  |  |  |  |
| --- | --- | --- | --- |
| **Biome** | **Value Weighting** | **Colour (RGB, Normalised to 0-1)** | **Provincial Spread** |
| Ocean | Low Elevation  NA Temperature  NA Rainfall  NA Flora | (0.04,0.08,0.58) | 400 |
| Temperate Forest | Medium Elevation  Medium Temperature  NA Rainfall  High Flora | (0.01,0.39,0) | 16 |
| Tropical Forest | Medium Elevation  High Temperature  High Rainfall  High Flora | (0.06,0.34,0.05) | 36 |
| Taiga | Medium Elevation  Low Temperature  NA Rainfall  High Flora | (0.06,0.22,0) | 36 |
| Grasslands | Medium Elevation  Medium Temperature  NA Rainfall  Low Flora | (0.02,0.54,0) | 24 |
| Savannah | Medium Elevation  High Temperature  High Rainfall  Low Flora | (0.78,0.55,0.15) | 16 |
| Tundra | Medium Elevation  Low Temperature  NA Rainfall  Low Flora | (1,0.78,0.78) | 48 |
| Desert | Medium Elevation  High Temperature  Low Rainfall  Low Flora | (0.77,0.61,0.23) | 36 |
| Mountain | High Elevation  Low Temperature  NA Rainfall  Low Flora | (0.5,0.5,0.5) | 8 |
| Forested Plateau | High Elevation  NA Temperature  NA Rainfall  High Flora | (0.5,0.5,0.5) | 8 |
| Shrubland Plateau | High Elevation  High Temperature  NA Rainfall  Low Flora | (0.5,0.5,0.5) | 8 |

Map

Description automatically generatedThe completed biome system is able to produce results depicting a believable continental structure and demonstrates many geographic principles such as the concentration of temperature based on distance towards polar regions or the equator. The implemented biomes create diverse ecosystems that are later used in the simulation to demonstrate how the factors that make up a biome can affect the development of a world.

Figure A world generated using the completed world generation algorithm

Once the world generation procedure finishes, the software quickly moves onto splitting the world into provinces. The provincial system has no variance from the way it was designed – splitting the map into right angled triangles and then connecting them to form provinces. As shown in the biomes table above, each biome has an appropriate spread value which dictates how far the borders can spread across the world map. These spread distances are determined by the initial starting location of a province, but once generated, the properties of a province are determined by the most prevalent of each property in each chunk – meaning a province with 6 chunks, with 4 high elevation chunks and 2 medium elevation chunks will be considered a high elevation province.

* 1. World Properties Implementation

After generating the world map, the software begins the procedure of generating values for each province, allowing for the simple polygonal locations to have uniquity, giving more credence to the idea that these shapes represent a group of people and their lives. In order to better represent these properties on the map, new “map modes” were implemented, allowing the map to show users different relevant statistics at a single glance. These map modes can be selected at any time using the dedicated map mode panel.

Figure An example of the map mode window. Each icon representing a different displayable property on the map. In this screenshot, the elevation map mode is selected.

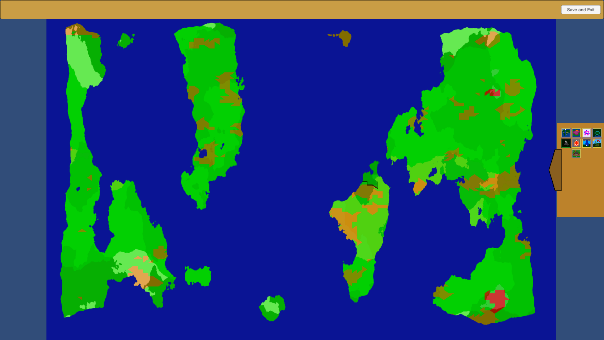
The first property to be implemented was the inclusion of hills and craggy terrain. The original map generation stores the elevation of a location as a trinary set – “Low”, “Medium” or “High” – with low referring to anything below sea level, medium as anything on land, and high as any mountainous location. This initial procedure generated relatively flat land, with only minor inclines in the form of mountains and plateaus. To fix this, the hills procedure first sets all “Medium” elevation land to “Low” (Excluding oceans, which fall under a new property, “NA”) before selecting a random assortment of these “Low” provinces to be incremented to “Medium”. The implementation of hills provides a new avenue for the eventual decision-making procedure to consider – as heightened terrain provides a defensive advantage to any owner of the province.

Figure An elevation map. Showing green land as flat land - yellow land as hills and red land as mountains.

* + 1. Cultures & Names

Secondly, each province on the map was given a unique name – generated randomly using the naming procedure introduced in the “Iron Age” project. By selecting random amounts of letters from a real city name, and then appending another random number of letters from a city name with a letter in the same position – realistic provincial names can be created to give each location on the map a unique “capital” to represent the location as a whole. The data set from which these names are pulled contains a listing of over one hundred and thirty thousand unique city names from across the globe. This procedure, while sometimes tending to generate inappropriate outputs, often produces realistic sounding city names, with the matching of characters property meaning that cities from similar cultures and languages are often paired together. Some examples of names produced by this algorithm are the following:

* Krasonvi
* New Marti
* Attastia
* Endalu
* Bajakh
* Vantelier
* Ameraya

Map

Description automatically generatedThis algorithm was then built upon to produce two variants – the cultures procedure and the ocean names. In the terms of the simulation, a culture refers to a group of provinces that share similar ideas and ethics. In the simulated environment, provinces that share a culture with a nation are the main focus of a nation – as it seeks to unify its people and handles complex relations with potential allies or rivals.

Figure The cultures map mode

The culture procedure uses a variant of a flood-fill algorithm. At the start, a number of random provinces are selected to form new cultures. One of these provinces is then selected to spread its culture to all adjacent provinces that do not yet have a culture. Once this is completed, the procedure starts again, a random culture is selected, and a random province with adjacent provinces without a culture is selected within that culture to spread to all adjacent provinces. This procedure repeats until all cultures have no more applicable adjacent provinces, causing the software to check that all provinces have been granted a culture. If this condition is not met, a new number of provinces with no assigned culture are granted new cultures to spread to their adjacent – repeating this procedure until every province has a culture assigned to it, no matter how small said culture may be. The results of this represent a varied world of different groups – mainland provinces will often belong to large contiguous culture blocks, whereas islands will often have cultures that extend only to the borders of the island.

These cultures, like the provinces that they encompass them, are granted names. This name generation algorithm, as previously mentioned, uses a variant of the province naming procedure. A dataset included with the software contains a list of roughly seven-hundred different adjectives to describe cultures, nations and ethnic groups. Each name in this set is then reversed before the prior described name generation procedure is applied – ensuring that the suffixes are maintained. When the desired number of culture names are generated, the names are applied randomly to each culture. As the sample size of names is relatively low, there is a high risk of duplicate names, therefore the algorithm queries the set of generated names for duplicates before appending any new culture names to the set. Some examples of culture names generated are the following:

* Ippian
* Digarhi
* Canese
* Lusian
* Lilandian
* Aquenian

Finally, each ocean is also provided its own name, using the same algorithm as the culture naming procedure but with a different dataset. The dataset in question in this instance includes almost five hundred unique eponymous adjectives, describing concepts such as philosophies and religious figures. This produces a set of names that are noticeably distinct from cultures and lean more towards concepts rather than descriptions. After this procedure is complete, each name is provided a suffix to that refers to its position relative to land, creating names like the following:

* Ersonian Ocean
* Censian Gulf
* Vaudian Sea
* Bertite Ocean
* Aurian Ocean
* Ysonian Sea
* Nesian Cove
  + 1. Population

After generating these properties, the arguably most important metric can be determined – the population of a region. This property will later determine aspects like when nations will rise and fall, and so must incorporate most of the properties generated to create an estimate for where people are most likely to live. For this, there are five values used – Culture size, Elevation, Temperature, Rainfall and Flora.

Each of these properties are given numeric values to allow their usage in a formula to determine a “score” for the population of a region. Culture size, referring to the number of provinces on the map that adhere to a specific culture, is given a decimal value from 0 to 1, normalized according to the highest culture size and lowest culture size on the map. A provinces elevation is given a value of -1 if in the lowlands, 0 if on a hill and 1 if on a mountain, likewise, temperature receives a -1 for a cold climate, 0 for a temperate climate and 1 for a hot climate. For the two Boolean properties, rainfall, and flora, they are each provided a 0 for a “Low” value and 1 for a “High” value. These numbers are then used in the following equation:

This equation makes use of each property in a way that reflects the way a property would impact the population of a region – for example, the flora property is significantly important to the formula – allowing for dense populations to generate in locations with easy access to food. The produced results of this demonstrate a world defined by its surroundings, with populations mostly generating in close proximity to one another.

Map

Description automatically generated

Figure A world population map. In this world, civilization has largely centered around the upper right corner of the map due to its moderate temperature, high rainfall, and high forestry. Other populated regions can be seen scattered across the map

* + 1. Saving

Once a world has been generated, it is automatically saved to a user’s disk in the directory from which the artefact is being executed. If one does not already exist, a “Saves” directory will be generated to contain the worlds generated. The world name can be determined on saving, but will default to “World(1)”. When a duplicate world name is detected, a suffix is applied to distinguish it from the other file – so if a default world name is selected while “World(1)” already exists, the new world will be called “World(2)”, with the same logic applying to custom names as well. The save data folder contains a number of relevant data to be loaded in the simulation stage, most notably the provincial properties and component vertices and the world map image.

Each save file within the document is written in an XML format – with nodes representing concepts such as provinces and cultures. When first generating a world, five files are written to – “World.sav”, “WorldData/Provinces.xml” and “WorldData/Cultures.xml”, “Simulation/Empires.xml” and “Simulation/Religions.xml”. The first file, the world save file defines the basic properties for a world, such as the height and width of the map, and its data is loaded first to ensure all world configuration settings are maintained.

The provinces xml file contains all the relevant province object data – including its name, ID, properties, culture, and mesh vertices, allowing for the province to be loaded in the same format it was originally generated in. When loaded, the map procedure will automatically reconstruct the province meshes from the vertices provided in the correct order to ensure the vertex faces align.

The cultures xml file contains all relevant data to cultures, such as the name and ID. Provinces only store the ID for their culture, and therefore it is imperative to load the list of cultures to serve as a reference for this data. The final two files serve as placeholder files for any data to be saved during the course of the simulation produce – each providing data for the nations of the world and the religions of the world respectively.

Within the save directory, there are two image files – “Map.png” and “Mask.png”, each with their own important purpose to help with the presentation of the simulator. Map.png stores the map texture, the backdrop for all the provinces to be rendered in front of, while Mask.png stores only the ocean texture. This masking image is placed in front of the provinces, creating an effect in which the provincial borders fit to the land borders, while still maintaining their initial interaction hitbox. By using both these image files, the map can be stored and loaded in a way that presents its best Map

Description automatically generatedaspects.

Figure Map.png and Mask.png. Map.png is placed on the back layer when displaying the world, followed by the provinces layer, and finally Mask.png on the frontmost layer. This means that the provinces appear to fit to the map, regardless of if their polygonal structure overlaps with the ocean or not.

1. Simulation Design & Methodology

After completion of the world generation and properties segment of development, the artefact may move onto the development of its primary feature – the simulation of a world upon the generated map. The artefact is intended to produce a believable world map, influenced by the actions and mistakes of its denizens - the “empires” that will form upon the world.

* 1. Empires

An empire in this artefact is the representation of a collection of people, united under one ruler. These nations will be the focus of the artificial intelligence – each having the opportunity to expand, develop, conquer, and fall dependent on its reigning monarch. An empire consists of the following:

* A name, defined by the province it forms in – the capital
* A colour, once again inherited from its capital province
* A list of owned provinces – which, when reaching zero, causes the empire to cease to exist
* A culture
* Five “technologies”
  + Military technology
  + Economic technology
  + Diplomatic technology
  + Logistics technology
  + Cultural technology
* A reigning monarch
* A state religion
* A military force – consisting of a maximum value and a current value
* A list of opinions of other known nations
* A list of any active rebellions
  + 1. Ruler

First and foremost, each nation will be designated a ruler, the representation of the artificial intelligence for the nation. A ruler consists of a name, an age, a birthday, two “technology focuses” as well as a list of weighted personality values. The personality values of a ruler are a numeric equivalent of their likelihood to undertake certain actions – a ruler with a high score for colonising land will be more likely to colonise land than others. These personality traits and their associated actions are as follows:

* Declare war – Start a war with a neighbour
* Develop technology – Attempt to increase one of the technology values for the nation
* Learn technology – Increase a technology value by learning from a neighbouring nation
* Spread religion – Either adopt a primary religion or convert provinces within the empire
* Increase opinion – Increase a peer nation’s opinion of this empire
* Spawn rebellion – Increase unrest in some provinces, or spawn a rebellion if there is too high unrest in a region
* Colonise – Attempt to gain control of a province with no current occupier
* Attack – Either attempt to gain control of a province owned by a warred nation or suppress an active rebellion using military force
* Stir unrest – Increase unrest in the provinces of a neighbouring nation
* Idle – Take no action and end the action state

When a ruler is able to act, they will order these personality traits in a random-weighted order, attempting to do the action pertaining to each trait in turn, ending their “action” state if they either successfully complete an action or reach the “idle” action. In this way, each ruler will act with their own agenda, though notably some caveats exist to ensure rulers consider the impacts of their actions, rather than just randomly selecting actions.

Firstly, if an action cannot be completed, either due to having no applicable targets (such as with colonisation or spreading religion) or having insufficient military power to complete the action, the next action in sequence will be selected. Secondly, when evaluating some actions, a ruler will consider the opinion penalty said action will incur, and how it will impact their standing with their peers. Finally, decisions will be impacted by other “unpolled” personality traits; “Risk” and “Insult”. The “Risk” personality value determines how much military power a ruler is willing to divert to an action. This presents an avenue for mistakes to be made – a high “Risk” ruler may take actions that use up large portions of their military power, putting them at risk of foreign aggression, whereas a low “Risk” ruler may not take actions that would benefit their nation. The “Insult” personality value determines their willingness to take actions that will sour relations with enemy nations – with high “Insult” value rulers potentially even prioritising tasks that hinder their enemies.

In addition, there are two final personality traits that are both unpolled and do not contribute to actions directly. These are the “Calm” and “Teach” personality values. Each year, a ruler has an opportunity to decrease unrest across their nation – with the likelihood of taking this action being determined by the “Calm” trait. The “Teach” trait only impacts the ascent of a new ruler.

When a ruler dies, a new ruler will replace them, most likely from the same family as the former ruler. This ruler will receive a new set of personality values, though (if this new ruler is from the same family as the former ruler) these values will be biased by the “Teach” personality value of the former ruler. A high “Teach” value ruler will produce heirs with little variance from their own values, whereas a low “Teach” rulers heirs have a higher maximum difference of personality from their predecessor. If the ruling dynasty is replaced, either through random chance or through rebellion, the new ruler may inherit from a ruler within the same culture group or be an entirely random new ruler.

* + 1. Culture, Economy and Military

When an empire forms, it will inherit the culture of its capital province. A culture determines a number of aspects of the empires, most notably the foreign relations, economy, and stability of a nation. Each culture has its own economic system, comprising of the economic output of each occupied province within the culture.

When occupied, a province will contribute to economy of its culture based on its properties, including:

* The population of the province
* If the province is coastal
* The amount of adjacent occupied provinces
* The economic technology value of their nation
* If the province is owned by a nation that does not share the provinces culture, the province will contribute a reduced amount towards its original culture, with the majority being diverted to the culture of its occupier
* The total technology value of its occupier nation

The scores of each province within a culture will then be totalled to produce the economic score of a culture, and each nation within that culture will receive the proportion of the economic output they contributed towards the economy, with some notable caveats:

* A province will produce less economic output for its nation dependent on its unrest value
* A rebellious province will produce no economic output for its nation
* A province with a culture different from the occupier’s primary culture will produce reduced economic output
* A province with a religion different from the occupier’s primary religion will produce reduced economic output

The economic output of a nation is a vital statistic, as it determines the influx of military power received by a nation. Each nation has a “military power” score, representing not just their army but the technologies held by said army. Every three months, a nation will receive new military power dependent on their economic output, with the military power unable to exceed the maximum military power as defined by its military technology and population-adjusted size of the nation. The military power of a nation acts as the de-facto currency of a nation, determining how much they can field in battles, as well as being the “cost” of colonising unowned provinces. To maintain a successful military, an empire must carefully manage its own economic score.

* + 1. Religion

When forming, a nation will either inherit the religion of its capital or, in the case of there being no such religion present, act as a pagan state. Religions, which form and spread across the map over time, determine much of the opinion and unrest statistics of neighbours and provinces respectively. A ruler may choose to adopt a religion once per lifetime, taking the most-populous religion within their borders and designating it as the state religion. After this action is taken, subsequent calls to said action will allow the ruler to spread the religion to both pagan and heretical provinces – attempting to convert their nation and thereby maintain stability. The religion of a nation also highly influences its foreign relations – nations will often find themselves at odds with one another over their respective faiths, meaning wars are more likely to break out between these ideologically different empires.

* + 1. Technology

The technology aspect of the simulation consists of five major technology groups, representing different aspects of real-world technology and having different impacts on the simulation itself. The first, military technology, represents the development of both weapons of war and national tactics, serving to increment the maximum military value of a nation. The second, economic value, represents the development of trade and industry across the nation, and increments the economic output of each province within the empire’s borders. Diplomacy technology represents both the foreign policy of governments as well as the speed of their communication methods, causing it to both impact the opinions of a nation held by other nations as well as increase the chance of a nation being to act on each day. Logistics technology represents the coordination of a nation’s infrastructure and supply convoys – with higher logistics technology allowing nations to take lesser losses in battles. Finally, culture technology represents the cultural and religious unity of a nation – both in terms of national identity and adherence to a nation’s laws – and as such increments the maximum unrest that can be accrued before a rebellion may spawn.

Technology may be increased through a number of different methods, as discussed in the action design. A newly formed nation will spawn with the lowest technology for each category displayed by other nations with a shared culture – meaning if all nations in a culture have a minimum value of 2 for military technology, any new nation that forms will inherit this baseline of military technology. In cases where there are no other nations of a culture, newly formed empires will inherit the default values of 1 for each category.

* + 1. Opinions

Each nation will hold an opinion of all nations within its culture as well as any empires they share a border with, a value which will update each month. These opinions represent the relations between countries and allow for more complex decisions to be made dependent on the political climate of a region. There are a number of factors that will impact the opinions of nations, most notably the presence of a shared culture or religion – which will by default cause an opinion bonus between nations, whereas a lack of shared religion or culture will cause a negative opinion penalty. In addition to this, a shared dynasty between monarchs will also cause a large increase in opinion between the two nations.

Opinions may also be modified by the titular “modifiers”, which present time-sensitive changes to opinion, be it positive or negative, dependent on the actions of a nation. For example, when a nation increases a peer’s opinion of it, enemies of the targeted nation will gain a short-term opinion penalty of the empire that initiated the action, whereas the target will receive a longer-term increased opinion bonus of the acting nation. Of note are two war related modifiers – the first, the war modifier itself, allows nations to attack one another while also providing a large opinion penalty. The latter represents a peace treaty between nations, giving a small opinion penalty for a large period of time, during which time neither nation may attack the other.

After calculating opinions, a nation may categorise a peer under four labels. The first of these, “ally”, represents a high opinion between two nations which means they are very unlikely to act in ways that would ruin relations with the other, and entirely unable to attack their ally.

The second, “rival”, represents a particularly strained relationship between two nations. Most often forming when two nations have similar relative economic scores within a shared culture group, rivals will have low opinions of one another, causing them to act competitively towards one another – even antagonising each other based on the “insult” personality traits of their respective monarchs. Rival nations have a high chance of going to war with one another, further lowering their relations until one becomes powerful enough to no longer consider the other a threat.

The third, “feared”, is a one-way relationship which is characterised by one nation feeling at risk of aggression from another nation. Nations that fear others will often stray from actions that would anger the others and may attempt to improve relations in order to minimize the risk of war between the two. This opinion type is most often formed when an empire has a large difference in military power compared to the other, an aspect which is modified heavily by the “risk” personality value of a ruler.

The final category, “unimportant”, represents a one-way relation between a nation in which they consider another nation entirely unimportant, stopping them from considering the others opinion when determining actions. This is most likely paired with a “feared” opinion from the opposing empire and is common in nations with significantly higher military power than their peers.

* + 1. Unrest & Rebellions

Each occupied province stores a value of “unrest”, representing the dissatisfaction of the populace with the administration. Unrest can be incremented in many ways, including:

* Aggravation by the ruler using the “spawn rebellion” action
* Forced conversion by the “spread religion” action
* Capture in war
* Loss of provinces in war, especially the capital
* New rulers taking power
* New dynasties taking power
* Other nations using the “stir unrest” action

While the rulers “calm” personality value decreases unrest over time, if the unrest of a province exceeds a culture technology dependent maximum unrest value, use of the “spawn rebellion” action will cause a province and any nearby dissident provinces to erupt into open rebellion. Rebellions will remain the property of the nation at large, while providing no economic output to the empire. During a rebellion, unrest in nearby provinces will increase, and the rebels will gain some military power over time. A rebellion can come in one of four forms:

The first kind of rebellion, a cultural rebellion, refers to a rebellion vying for independence due to a difference in culture from the empire at large. Cultural rebellions may only spread to provinces that share their culture, as their objective is to unify their group under the banner of a new empire. When gaining enough military power, the rebel-held provinces will split into a new independent state, starting an immediate war with its former sovereign, and reducing unrest across the provinces significantly.

The second, religious rebellion, attempts to separate from its owner empire much like the cultural rebellion. While the mechanics mirror that of the cultural rebellion, the primary difference between the two is that a religious rebellion can only spread to provinces with the same religion as the rebel group.

The third rebel type is revolutionary rebels. Revolutionary rebels are the most common rebel type and seek only to overthrow the ruler and replace them with another. When meeting the desired military power limit, they will start a battle targeting the capital city, during which either side will field a much larger military size than would normally occur. Unlike other rebel types, revolutions will not always decrease unrest when activating, and may in fact cause more unrest regardless of if they are successful or not. This allows revolutions to cause widespread rebellions – potentially collapsing entire empires as the nation spirals into instability.

The final rebellion type is separatist rebels, which, much like both the culture and religious rebels, attempt to split the realm by creating another independent state. Separatist rebels may only spread to the same culture as the province they spawned in, most likely the primary culture of the empire, and will only spawn if there is a concentrated group of unrest in one region.

Rebellions may only spawn if the empire has extended over its maximum national size, a value dependent on the cultural technology of the nation. This means that early empires are much more susceptible to rebellion than modern nations, and therefore may be subject to collapse if mismanaged.

* 1. Action Design

As mentioned prior, each ruler has a set of personality traits that determine their willingness to take certain actions. While the specifics of the action-selecting procedure are topics covered in the artificial intelligence design, the actions each personality trait represents will be developed first – with a debug console allowing the usage of commands to force national actions in lieu of automatic action selection.

* + 1. Spawning & Colonies

The most important thing in the development of a nation is its formation. As referenced in the literature review, the earliest nations formed from agrarian communities – people would concentrate around areas of stable food production, and social hierarchies would form, eventually leading to the development of simple governments and foreign relations. In this project, the flora and fauna of a region is used to determine the prehistory populations of a region, and therefore the existing population metric can be used to determine where these nations form.

Nation formation is largely handled by the simulator itself – rather than any natural action. An empire can spawn at any time in a high population or medium population region – with significant weighting to the former and coastal regions. When an empire owns a province in a cultural region, either through formation or expansion, the likelihood of a new nation developing within the region increases. The chance of a nation spawning is largely dependent on the year – with more nations forming as time passes, though as the simulation is intended to begin at the start of human civilisation, there is a significantly increased chance for a nation to spawn when no nations exist.

When an empire does exist, it may expand using the “colonise” action. This action represents the expansion of a nation’s borders by settling new communities in unclaimed regions, and as such a colony target must be unowned before a nation can expand into the region. To colonise a location, an empire must expend military power relative to the difficulty of expansion into the region, meaning nations must choose colony targets wisely as the creation of a colony may leave them susceptible to foreign influence. Colony cost is modified by the following factors:

* Cultural technology level (decrement)
* High or low population (increment) – as areas with pre-existing communities may be more resistant to occupiers whereas unpopulated land requires the development of infrastructure
* High elevation (increment)
* Low rainfall (increment)
* Culture of which the empire has no provinces with the same culture (increment)
* High flora (decrement)
* Coastal access (decrement)
* Number of adjacent provinces held by an empire (decrement)
* Number of provinces owned by an empire that share the biome of the target province (decrement)

When the colonise action is selected, a ruler will evaluate all potential targets of the action, comparing the cost and value of the province as well as any diplomatic impacts of colonising new land. The targets are then sorted by the ratio of value and cost, and each option sequentially evaluated before the ruler selects the first acceptable province, or otherwise rejects the action if no targets are found to be applicable.

* + 1. Technology Development & Learning

Each ruler has the opportunity to increase their technology level through two means – either through development or learning from peers. Attempting to develop a technology rolls a random number, with a match against a pre-set value determining if the ruler is able to develop a technology level. When developing a technology level, a ruler will consider both their personal preferred technologies as well as the lowest technology value held by their empire, choosing pseudo-randomly to determine the development target, which will then be incremented by 1 at no cost.

When attempting to learn a technology, a nation will instead compile a list of all technologies held by those within their culture as well as any adjacent nations. The nation will then find the highest disparity between its own technology and its peers and increment the value of the technology with the highest difference. This action is not tied to any chance-based mechanic and will always be successful if a nation has lower technology than its peers.

* + 1. Religion Spreading

Rulers who value spirituality may choose to spread their religion within the nation. When a ruler ascends to power, they gain the opportunity to determine their state religion – accepting the religion with the highest population-adjusted acceptance as the new official religion. If no religions exist within a ruler’s domain, they will not select a state religion but will maintain their ability to set a state religion. Each ruler only has one opportunity to change the state religion, regardless of if it changes during their reign, allowing for dissent to develop against peoples of differing faiths.

If a ruler has selected a state religion, or otherwise reaffirmed the previous state religion, they may attempt to spread the religion within their borders, randomly selecting a number of provinces with different beliefs to convert to the accepted faith. This procedure accumulates unrest within targeted provinces, which may spark revolt in instable realms. While rulers may choose to manually spread religions, the changing of religions is also conducted randomly, in which a province may gain a new religion by proximity to provinces of differing religious values. This may cause friction between a particularly spiritual ruler and their nation, especially so if the majority religion is to change when power transfers to a new ruler.

* + 1. Opinion Changing

Rulers may choose to send resident diplomats to peer nations to improve their relations for a period of time, potentially forming alliances or attempting to appease feared nations. The positive opinion modifier provided by this action lasts for a period of ten years, and applies to each nation, incrementing the opinion value of each against the other. The exact value of this change in opinion is determined randomly, with a minimum of twenty opinion points and a maximum of 25 through 45, depending on the diplomatic technology of the acting empire. The envoy modifier may be applied multiple times to a single empire, meaning repeat actions will allow for high opinion bonuses to be accrued between nations. However, the sending of envoys may have the consequence of reducing relations of the acting nation against rivals of the target, an aspect considered by the ruler when determining the chance of the action.

* + 1. War Declaration and Attacking

When opinions become strained, or a nation sees potential value in conquering land held by a rival, a war may be started between two empires. When at war, nations will attempt to wrest control of provinces by starting battles over territory. Each battle starts when an empire at war polls the “attack” action, after which they will choose a province to attack. Each side of the conflict will field a proportion of their current military power towards the battle, and the chance of the attacker succeeding will be calculated using these values as well as other factors. The following formula will determine the chance of each side:

Modifiers for the attacker side are:

* -0.2 by default, due to having to take an offensive position
* +1 if the province is in a high flora environment, in which there are more natural supplies for an attacker
* +0.2 if the target province is at a low elevation
  + +0.8 if the any adjacent attacker held province is at a medium elevation
* +0.5 if the target province is of a different religion from the attacker, or either side has no religion

Modifiers for the defensive side are:

* +0.2 by default, due to prepared fortifications
* +1.5 if the target province has medium or high elevation
* +0.5 if the province has coastal access, as this would allow for resupply during a siege
* +0.5 if the province is in a low-temperature region
* +0.5 if the province is in high temperatures and the attacker has less than half provinces in high-temperature regions, as attackers with less experience in warm environments will struggle in the heat

Reinforcement bonus refers to the number of provinces adjacent to the target province held by each side. While the side with less adjacent provinces receives no bonus, the side with more adjacent provinces receives a bonus of the following (at a maximum of 2):

After this is done, the final chance of attacker success can be calculated (though it should be noted the maximum chance is 95%, and the minimum 5%):

This value will be computed before any battle actually occurs, allowing rulers to determine the risk vs reward of attempting to siege a province. When the battle does occur, each side will take losses dependent on their logistics technology and their success in the battle. When an attacker is successful, they take control of the targeted province, raising its unrest. Regardless of the results of the conflict, each empire will accrue some “war exhaustion”, a value that represents both the happiness of an empire with captured objectives, as well as the willingness to make peace due to losses. When both nations reach their maximum war exhaustion, as determined at the start of the war, they will make peace, maintaining any possessions they acquired during the war. When not at war, empires will slowly lose war exhaustion, and will not attempt to start new wars when they have remaining war exhaustion. If war is declared on them during a period in which they have war exhaustion, the maximum war exhaustion for the new war will be incremented by their existing war exhaustion – meaning while they will not instantly surrender, they will have a longer waiting period before they attempt to start future conflicts.

Upon the ending of a war, each side will receive the “peace treaty” opinion modifier as previously mentioned, both preventing wars from occurring between the two combatants for a period of time, as well as reducing opinions of one-another during this time.

* + 1. Rebellion Spawning and Stirring Unrest

The unrest system allows for nations to splinter and fall depending on the impacts of the actions they take, most notably in the cases of the rebellion spawning and stirring unrest actions. The “spawn rebellion” action refers to two separate events – either the ruler acting in a way that sparks controversy or the actual development of a rebellious group in response to actions by the government. The former event occurs whenever this action is polled, in which a number of random provinces increase their unrest levels in response to the modelled slight made against them by their monarch. The latter only occurs if any provinces reach over their maximum unrest threshold, determined by the cultural technology of the empire, in which a new rebel group will form in select locations. By design, these actions are heavily detrimental to the stability of a realm, and the repeated ascension of rulers with high precedent for this action may cause the downfall of an empire.

On the other hand, nations may also disrupt rivals purposefully, raising unrest in random provinces in a targeted nation much like how the spawn rebellion action would - with the exception that rebellions cannot spawn as a direct result of this action. In the setting of the model, this represents the sowing of discontent by spies or other operatives to destabilise a realm for political gain, allowing rulers to fell their rivals without using any military power.

* + 1. Rebel Suppression

When a rebel group is active within a nation, a nation will gain the ability to attempt to suppress the revolution in a province of their choice, simulating a battle between the two forces. In this endeavour, the empire almost always succeeds, as a rebel group with sufficient military power to oppose the nation at large would likely already have started their attack. When winning a suppression battle, the unrest of a suppressed province will be heavily decreased, meaning rebellion is unlikely to strike in the province for a long period of time. On the other hand, an empire losing will signal weakness to other provinces, increasing unrest across the realm dependent on the chance of victory held by the empire – in which high victory chance losses increase the unrest gain more than the alternative.

* + 1. Rebel Actions

While not an empire in their own right, active rebellions too have the ability to take actions to expand their cause – though at a rate much slower than that of normal empires. First and foremost, rebels may attempt to add unstable provinces to their controlled lands, even influencing provinces in ways that increase their unrest in order to gain allies for their ambitions. When a rebel group gains enough military power over their empire, a value accrued over time that represents the military power gain that would have been applied to the nation from a province were there no rebel group in place to nullify the economic output of the province, they may attack with an action dependent on their rebel type.

When a revolutionary rebellion attacks, a battle will be simulated between the two groups, using the same metrics as regular battles with the capital of a nation being the targeted province. The military power fielded for either side during this process is significantly higher than usual, as the attack is purely centred on one location. When this battle concludes, regardless of the result, the rebel group will disband, though the unrest held by provinces will largely remain the same, paving the way for future conflicts.

With all other rebel groups, meeting the military power threshold means the formation of a new nation, at which point the original empire immediately declares war. The conflict caused by this is notably different from other wars, in which war exhaustion is not accrued by the original empire when winning battles, but heavily increased when losing battles. Due to this, war between rebel states and their former masters can only end in the extinction of one nation or an uneasy peace between the two. When ending a war in the latter format, resulting in a rebel victory, land will be exchanged based on the conditions of the rebel group; Cultural rebels and separatists will only gain control of land with their culture, whereas religious rebels will only maintain control of provinces that share their religion.

* 1. Artificial Intelligence Design

When the actions referred to prior are implemented, the simulation segment can begin development. Each day, every empire rolls a random number between zero and one, and may act if their roll is lower than their acting chance, determined by the following formula:

If the chance is successful, then a ruler may generate a set of actions, ordered by their likelihood to occur. As mentioned prior, each action is tied to a specific personality trait of a ruler, represented by a value between zero and one, relating to the chance of the action occurring. The actions taken by a ruler are determined randomly by the weighting of each personality trait.

To determine the action sequence, the set of actions is first shuffled, and a new random number generated, with a maximum value of the maximum personality trait value of a ruler. Each item in the shuffled set is checked in sequence, and the first that has a value greater than or equal to the random number is appended to the action sequence. This process is repeated with new random numbers each time to create a set of all the actions biased towards the ruler’s personal preference. When a set is created, all items after the “idle” personality trait are removed, as calls to the idle command will always result in the use of the empires “turn”.

* + 1. Acting

When the action queue is generated, the first action in the set is polled to simulate an “attempt” of the act. When selecting an action, the ruler is still able to opt out of committing to the selection, regardless of action sequence, in which case the next act in the set is selected and attempted, only ending the procedure when a successful action occurs, or the idle action is reached. The chance of a ruler to decline an action in their queue is determined by a number of factors.

Firstly, if the action is impossible for any reason, such as a colony action when all adjacent provinces are occupied, the action will automatically fail. Secondly, any action to which the military cost requirement exceeds the ruler’s maximum military expenditure value (determined by their “risk” personality trait) will fail, though it should be noted that in an action to which there are multiple targets, such as colonisation, the action will only fail if all targets exceed this value.

Finally, nations may choose to forego an action if the diplomatic consequences would be too heavy. When considering an action that may cause negative relations on important peers, the empire will compare the changes in values as positives or negatives. Positives include the increasing of relations with allied or feared nations, as well as the decreasing of relations with rivals. Negatives include the decreasing of relations with feared nations and allies. The exact impacts of the positive and negatives of each are determined by miscellaneous personality values, such as the “insult” value which determines how much the decrease of relations with rivals impacts the positive score.

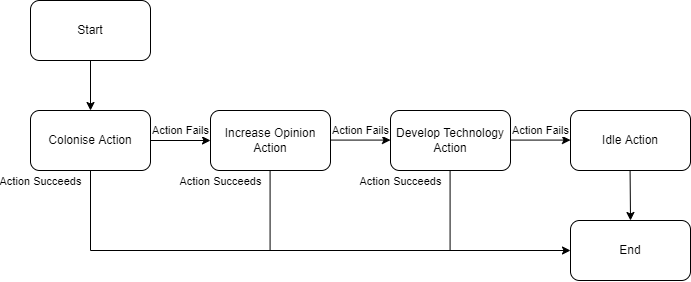


Figure An example of the action queue. In this, "Action fails" refers to when diplomatic or other concerns cause a ruler not to take an action.

* + 1. Provincial Value

Each province has an economic value determined by its properties and location, which contributes to the economic score of the occupying empire. While this score is considered when taking actions, specifically in situations like colonisation where the value and cost of a target is determined, there exists an additional factor to these calculations called “personal value”.

The personal value of a province is a nation-specific value determined by the summation of the existing economic score and the personal factors surrounding the province. The most notable of these factors is the adjacency bonus. For stability, empires will attempt to create clean borders that can resist attacks from aggressors, as reinforcement bonuses are dependent on the number of adjacent provinces to a battleground. To facilitate this, the personal value of a province is highly influenced by the number of adjacencies to other owned provinces, forcing empires to construct structurally sound shapes to define their borders.

Additionally, personal value is also increased by the presence of a shared culture or religion, meaning nations will attempt to unify people who share similar ideas. This also means that nations will be more likely to value “safe” provinces, ones with less chance of developing rebellions that could tear the nation apart.

Personal value is also modified by the presence of coastal access as well as the comparison of the biome of the target and the capital. Coastal access means better trade and defence in conflict, and therefore nations will value it for stability, whereas biome sharing means that nations will naturally seek to encompass areas of similar climate, if possible.

Finally, personal value is hugely incremented when the province in question is fully surrounded by the acting empire, allowing nations to eliminate enclaves that may threaten their nation, as well as form rigid structures as previously described.

* + 1. War Decisions

While an empire is at war, numerous changes are made to the AI to ensure excessively disadvantageous and unrealistic decisions aren’t made. For example, the colonisation and religion spreading actions are temporarily unavailable during this period due to said actions requiring the use of military power and raising unrest respectively – two modifiers that could change the outcome of the war. Empires will also not declare another war while already in a war, though this does not extend both directions – nations may have war declared on them despite already acting in a conflict.

Declaring a war is one of the most heavily monitored actions, with rulers having numerous decisions to make that ultimately decide if they believe that war is necessary or not. First of all, war targets may only consist of the owners of provinces the empire shares a border with and said empire may not be an ally or feared nation. Additionally, the nation cannot declare war on nations it has the twenty-year “treaty” opinion modifier active with – as a truce declared by nations cannot be broken for the purposes of the simulation.

When the list of war targets is compiled, nations will next evaluate the “score” value of a prospective war. The average personal value of the provinces held by the target from the perspective of the acting nation is calculated and appended to the score. This score is then incremented by the sum of the personal value of each enemy border province, modified by the chance of an attack action successfully taking the province – with the condition that if the chance of victory is greater than the rulers “risk” personality, the province value will negatively impact the score value. After the de-facto “potential gain” of a war is added to the score, nations will then use the opinion value ascribed to the target to generate a multiplier for the value – with low opinions granting a higher multiplier to the score. This multiplier is then once again modified by the difference in military power (both current, maximum, and expected gain) between the two nations, with the acting monarchs “risk” value referenced to determine what they believe to be a successful war. Finally, the target with the highest multiplier-modified-score is selected as the war target, and war is declared between the two nations.

During the course of war, the chance of an action occurring is greatly increased for both combatants. When a nation gets the random chance to act, they will first check if peace is possible – a condition satisfied when each side exceeds their war exhaustion capacity. If the conditions for peace have not yet been met, the acting nation will compile a list of all attackable provinces – those that are adjacent to lands held by the acting nation – and determine the battle statistics for each – including the cost and victory chance. These generated statistics will then be used to remove from the set all provinces in which the cost exceeds that of which the ruler is willing to risk, as well as any that have chances lower than the risk personality value of the ruler. All remaining items are then sorted by their personal value to risk ratio, and the battle will be simulated.

In the event of a nation having no possible targets in the war, most commonly caused by other empires taking land in a war against the same target nation, then each side will accrue war exhaustion automatically, meaning peace will be made regardless of if the war is able to continue. Additionally, if either side no longer exists – as all provinces have been removed – wars will be ended automatically, regardless of if the acting nation is the one who caused the fall of the opposing empire.

1. Simulation Implementation & Balancing

The implementation of the described simulation features of the artefact took considerably less time than expected – with much of the time being spent considering the approach taken and how it would change the progression of the simulation. A considerable number of designed systems required adjusting as the natural results of their parameters and algorithms caused problems, most notably in the form of the technology system. This segment will discuss any notable parts of the implementation and changes made from the design plan, as well as why these were necessary.

* 1. Console

As mentioned in the design document, the first step taken when developing the simulation was the creation of a simple console – allowing the forcing of certain actions and examining of different properties. The commands implemented were as follows:

* DEBUG – allowing for the opening of the province viewer “debug” menu, which listed internal aspects such as the ID values of provinces and empires
* ECHO – a simple console testing command able to return the parameters of a request to the console
* EMPCOUNT – returns the number of existing empires in the simulation at the current moment
* SPAWN [Province ID] – allows the generating of a new empire at a position, regardless of the actual chance of nations forming at said location
* ADD [Province ID] [Empire ID] – appends a province to the component provinces of an empire, without any costs being applied or war requirements necessary
* ECOUPDATE – forces an immediate call to the economics system to reorganise its data, ignoring the regular time requirements of the system
* TECHUP [Empire ID] [Type] – increments a technology value by one in a specific empire, without any chance mechanics or costs associated
* SPAWNMIL – much like the ECOUPDATE command, this forces all nations to gain their expected military power increase without reference to the date said action would normally occur upon
* SPAWNREL [Province ID] – generates a new religion at the specified province, allowing it to spawn as any other religion would
* GRANTREL [Province ID] [Religion ID] – changes the religion of a specified province regardless of proximity to the religion in question
* STATEREL [Empire ID] [Religion ID] – changes the state religion of a nation to any specified, notably without modifying the rulers “changed religion” flag
* COLONY [Province ID] [Empire ID] – forces the colonise action to occur, while also checking that the action is possible given the targeted empires proximity and military power. This also removes the military power cost from the targets military power balance as would a normal colonisation action
* ATTACK [Province ID] [Empire ID] – forces a nation to attempt an attack on the province in question, provided the province is accessible and held by a nation that the specified empire is at war with. This simulates the battle as usual, and therefore can fail, while also reducing military power and gaining war exhaustion as normal
* ADDMOD [Receiving empire ID] [Sender empire ID] [Days] [Opinion change] – adds a new modifier to the receiver empire’s opinion of the sender, with the parameters set to the specified day and the amount of opinion change. This command will fail if one empire has no held opinion of the other, and when successful will have the opinion type of “MAGIC”, separating it from other modifiers
* CHECKOP [Empire ID] [Empire ID] – returns the opinion values of the two empires in either direction
* ADDUNREST [Province ID] [Unrest change] – changes the unrest value of a province, provided it is owned by an empire, by adding any specified value
* FORCEACTION [Empire ID] [Personality type] – forces the specified empire to attempt to use the specified action as if it had been conducted naturally. This means that the attempt may fail due to factors such as diplomacy or the personal risk of the ruler. Additionally, the original personality value chance of the action occurring is printed to the console

A picture containing graphical user interface

Description automatically generatedIt should be noted that the console actions described are intended for testing purposes only, and while each command has input validation in the form of checking the number of parameters and the value of supplied parameters, it is not thoroughly tested to the same degree as the simulation as a whole. In the case of an error occurring due to a console input, the simulation will continue as usual but print the error message to the console. It should be noted that any impacts caused by the action before the error occurred may still be applied.

Figure The updated map modes system

* 1. Representation
     1. Map Modes

As Juris Civilis seeks to demonstrate the impact of various factors onto national borders, these factors must be readily accessible throughout the run-time of the simulation. To do this, the pre-existing map modes interface received new modes to properly display characteristics of relevance to the current state of the simulation. The map modes, ordered from left to right and top to bottom in the included figure, are as follows:

* Geography – displays only the map with no nations or borders
* Nations – displays the nations on the map, each represented by their own colour.
* Cultures – displays the cultural regions of the map, represented by the culture colour
* Languages – displays the language of choice that defines the ruler names for cultures on the map
* Elevation – shows the height of each province, with lowlands as green, hills as yellow and mountains as red
* Temperature – shows the temperature of each province, with cold regions as red, moderate regions as yellow, and hot regions as red
* Rainfall – shows the access to rainfall for a province. Low rainfall provinces are highlighted in red, high rainfall areas are highlighted in green
* Flora – shows the flora access in the region, with low flora as red and high flora as green
* Population – the relative population density of a region, with low population regions in red, medium population regions in yellow and high population regions in green.
* Provinces – shows an overlay of all the province borders on the map, regardless of the presence of an owner empire
* Technology – depicts the normalised technology total of each empire, with the current technology leader in green and the lowest technology nation in red
* Military – the normalised current military power of a nation, demonstrating the military might comparisons between empires
* Economic – the normalised economic score of each culture, where the culture with the highest economic score is green and the lowest is red
* Contribution – the relative contribution to the cultural economy score presented by each nation. The colour of an empire refers only to its percentage of the contribution, and so one green nation may not have the same economic output as another from a different culture
* State religion – the accepted religion held by each empire, displayed in black if there is none applicable
* Local religion – the majority religion for each province, coloured to match the colour of the religion
* Rebellions – Shows any active rebellious provinces, in a colour representing their rebel type
  + Red refers to revolutionary rebels
  + Yellow refers to separatist rebels
  + Cultural and religious rebels are shown by the colour of their culture or religion respectively
* Unrest – shows the unrest of each province in reference to the maximum unrest of the empire. Red provinces are regions close to revolt, but may be more or less rebellious depending on the cultural technology value of its empire

One final button exists on the map mode panel – an eye button. This button is a quality-of-life feature designed to make the software more accessible. When clicked, the camera is moved back to its default position and zoom – preventing the possibility of a user getting “lost” while using the camera controls.

* + 1. Province & Empire Viewer

Graphical user interface

Description automatically generatedWhen selecting a province, a province viewer panel will appear on the right-hand side of the screen. This panel displays relevant information to the user about both the province in question and the empire that owns it – if applicable. To do this, the panel consists of two primary modes – province and empire.

Graphical user interface, application

Description automatically generated

The province mode shows information solely applicable to the province itself – details such as the biome, culture, and religion. On the other hand, the empire panel shows empire specific data. Seen in the attached figure, the empire panel shows the national culture, as well as the percentage of the economy said nation makes up, the military power, the ruler, the tech scores, and the diplomatic status of the nation. When selected, the details button opens a new menu, detailing specific properties of the empire, allowing for a user to carefully examine the situation of an empire.

Figure The empire view tab

Figure The province view tab

A picture containing graphical user interface

Description automatically generatedAs can be seen in the attached figure, the detailed menu provides far more information on specific factors pertaining to the nation. Of note is the ruler personality field – displaying an overview of the ruler’s personality traits. The personality name is defined by two descriptors, chosen by selecting the two traits with the highest variance from the baseline 0.5 personality value – meaning both extremely low and extremely high values are chosen. The first word in the personality name is the adjective attached to the second most extreme personality, and the second the noun attached to the most extreme personality trait. In the supplied figure, the ruler in question has the personality “Adventurous Conservative” – meaning they have a high likelihood to take the colonisation action, but a low chance to take the learn technology action.

Figure The detailed empire viewer

* + 1. Text

       Description automatically generatedNotifications

The final representation improvement added to the artefact was the notification system – in which on-screen text shows actions as they occur – keeping users up to date on what has occurred without requiring in-depth knowledge of the workings of the simulation. Notifications disappear within seconds, preventing them from overloading the user with information.

Figure A notification showing a colony being formed

* 1. Implementation

Throughout the development and testing of the artefact numerous changes had to be made to maintain the integrity of the simulation. For the most part, these changes do not stray too far from the original designs, but instead consist of modifications to existing formulas in order to create an overall better product. This segment will detail the developed features and any differences made from the initial design, as well as any challenges that were found across the development period.

* + 1. Empires & Acting

Empires largely remained unchanged from the initial design, implementing all the required features as expected, with only minor changes to parameters for various fields. One notable change from the design came in the form of the empire formation chance and the empire action chance. Upon the completion of the first working build, nations spawned at a rate that far exceeded expectations, causing the map to expand at an alarming rate. The chance of nations forming was eventually set to the following:

This system allowed for the formation of nations in the early stages of the simulation, while slowing the procedure as time progressed. The empire action chance, as previously mentioned, was also modified to better fit the time period of the simulation. After a period of testing and balancing, the final daily action chance was set to the following:

* + 1. Expansion

The rate of colonisation and expansion also proved a difficult subject, as the rate at which nations have grown in the real world is too variable to be accurately expressed as a specific formula. To best model the expansion rate of nations, the initial military cost of a province is defined as:

After this, a modifier value is created, with the initial value of 0. The modifier is then subjected to a number of conditions before it can be used to change the cost of a province.

|  |  |
| --- | --- |
| Condition | Modifier Change |
| High or Low Population | +2 |
| High Elevation | +2 |
| Low Rain | +2 |
| Not Same Culture | +3 |
| High Flora | -1 |
| Coastal | -1 |
| Number of Held Adjacent Provinces | -Min(3 , 0.5\*AdjacentsCount) |
| Same Biome as Empire Average | -1 |

This modifier then serves as a multiplier for the cost, with a base cost of 12 times the monthly military gain of a nation multiplied by the modifier. When this cost is calculated, the following final conditions are applied:

|  |  |
| --- | --- |
| Condition | Impact |
| Small Province (3 Vertices) | Half Cost |
| All Adjacent Provinces Held by Empire | Set equal to Military Gain |
| All Adjacent Provinces Held by multiple Empires | Set equal to Military Gain \* 1.5 |

* + 1. Culture & Economy

The culture and economy system maintained roughly the same design as originally intended, with the glaring exception of the language system. The language system was implemented after the creation of the rulers system, allowing for each ruler to have a name to identify themselves by. To keep these names similar within a group, each culture is now assigned a language by a process similar to the creation of the cultures themselves – meaning languages are often shared by multiple nearby cultures. Each language consists of a predefined list of real-world first and last names from various cultures and time periods. The language name groups are:

* Asian
* Latin American
* European
* Indian
* Islamic
* Latin
* Pacific Islander

In addition, some languages have specific rules towards their naming conventions, these are:

* Asian – last names are ordered before first names
* Pacific Islander – last names are not displayed
  + 1. Military & War

While initially undefined in the design process, suitable parameters for the military system were implemented across the course of the development process. This meant the algorithm underwent a variety of changes in order to meet the desires of the simulation. The maximum military size of a nation is defined by the following formula:

This system provides a large gap between nations – allowing larger and more advanced nations to rise in power, while not being so insurmountable that weaker empires have no fighting chance. Consistent with the design, each nation gains some military power every three months. This military gain is calculated by the formula:

It should be noted that for these two equations, the province value has a variable meaning. The first equation refers to the non-economic value of a province – calculating value based solely off the population score, access to water and adjacency bonuses. The province economic value seen in the second equation is the unrest-adjusted economic value experienced by a single province.

In terms of the war system, there are only small deviations from the original design. First and foremost, battles now have a cooldown system, in which a nation may not attack for a short period of time after launching an assault – preventing wars from happening too quickly, an issue that was prevalent in early builds of the simulation. Secondly, the peace treaty system now allows for longer peace treaty lengths dependent on the diplomatic technology score of a nation – meaning in the later years of the simulation, wars will begin to slow down considerably.

* + 1. Religion

The implemented religion system builds on the outlined concept considerably. Religions now are generated on world generation – in which twenty randomly-selected faiths from a predefined list of real-world religions are selected to be the major religions of the world. Every day, if the world does not generate a new empire, a religion may form. The chance of a religion spawning is set as the following:



When a religion does form, it may do so in any province of high or medium population as long as no religion exists in the region. After this point, every day there is a 1/2000 chance of religions gaining the ability to spread, in which a list of all provinces with an adjacency to a province of the religion is compiled and a random subset selected to gain the religion as their primary belief. This spreading of religion is distinct from the empire action religion spreading, as this “natural” spreading of religion can only occur between adjacent provinces and does not accumulate any unrest for the inhabitants.

Religion plays a large part in almost every aspect of the simulation – ranging from economic value, unrest, and diplomacy. A province that finds itself at odds with the primary religion of the state will automatically produce less economic value for its sovereign, and if forced to convert by the ruler will accrue a large amount of unrest. In terms of diplomacy, nations of differing religion automatically start with a -25 opinion penalty, preparing the two for future conflicts. If a nation is to switch religion, it will also gain an additional -20 opinion penalty against any nations loyal to the former religion, and a +10 opinion bonus to all peers who share the empires new religion – both of which are opinion modifiers that last for 10000 days.

* + 1. Technology

Figure The state religion (Top) of a continent vs the provincial religions (Bottom)

The most consistently changed simulation construct was the technology system. In the original design, technology was incremented when a ruler took the appropriate action, after which a random number was generated to decide if they could increase their technology or not. In practice, this system consistently caused a snowball effect – in the early stages of the simulation, nations would struggle to gain any technology points. However, as the diplomacy level slowly increased and therefore nations gained more opportunities to attempt to develop technology, nations could suddenly develop up to thirty levels of technology in the span of a number of years. This, combined with the fact that the learning system applied to all members of the culture, meant a single culture could dominate within the first hundred years.

Initially, this was changed by decreasing chance proportionally to the technology total of the nation, however, this caused the opposite problem – the chance of technology developing was too low and therefore technology failed to improve over a long period of time. In order to fix this, the next build used a different system in which nations accumulated a random number of “technology points” each time they selected the develop technology action – an integer number that would be set as the baseline for the random generator. When the technology development action fired, the nation would accrue some technology points, and a number between zero and one hundred would be generated. If the number was less than that of the technology points, then the technology would be developed, and the technology points set to a negative number proportional to their technology total. This system once again failed to meet expectations – ending up too slow to allow for accurate development. The final technology system reuses the existing technology points system, but rather than accruing points when selecting the develop technology action, the points are incremented each month by a variable value tied to various factors.

Map

Description automatically generated

When the number of points reaches 250 or above, a technology can be developed, and the tech points are set to the negative of the nation’s technology total. This system allows for a slower rate of technology development, in exchange for a better rate of technology learning between nations. To improve this, when learning technology, a nation may also gain multiple levels of technology at once, allowing for technology to spread quicker – however, as a counterbalance to this, learning was made to only apply between adjacent nations – rather than the original method of within a culture as well as between any adjacent nations.

* + 1. Opinions

Figure The technological gap between two disconnected landmasses at the year 1140

The national opinions system remains largely unchanged from the design – with the glaring exception of the recalculation time period. Each nation holds an opinion of all its adjacent empires and nations with a shared culture, ranging from the values -150 to 150. This value is calculated from two factors – the base opinion and the active modifiers. The base opinion is calculated each month based on the following factors:

|  |  |
| --- | --- |
| Factor | Modifier |
| Both pagan religions | -5 |
| Different non-pagan religions | -25 |
| Same religion | +15 |
| Same dynasty (last name) | +75 |
| Same culture | +10 |
| Was ally on last opinion calculation | +50 |
| Was rival on last opinion calculation | -25 |
| Is currently at war | -200 |
| Provinces captured in last war (decreases over time) | -(Number of captured provinces \* 10) |

After this calculation occurs, existing modifiers are then appended, and the value is limited to the range of -150 to 150. With this final opinion value calculated, the nations then may categorise peers into categories as described in the design – unimportant, ally, feared, rival and neutral. These categories are then used to determine how nations will act when presented with a situation in which opinions may be changed as a result of an action. To do this, the nation weighs the potential opinion modifier against the impacted nations, in a positive and negative system. If the negative value outweighs the positive value, then the action will be cancelled. The following lists the positive and negative score modifiers.

|  |  |
| --- | --- |
| Positive Score | |
| Factor | Score increment |
| For actions that would give a positive opinion modifier | |
| Is allied nation | (Opinion change) \* (ruler “increase opinion” personality value) |
| Is feared nation | (Opinion change) \* 1 - (ruler “risk” personality value) |
| For actions that would give a negative opinion modifier | |
| Is rival nation | (Opinion change) \* (ruler “insult” personality value) |

|  |  |
| --- | --- |
| Negative Score | |
| Factor | Score increment |
| For actions that would give a negative opinion modifier | |
| Is feared nation | (Opinion change) \* 1 - (ruler “risk” personality value) |
| Is allied nation | 0.2 |

Because of the immense number of calculations made in this system, it contributes to the majority of performance issues present in the simulation. In an effort to improve this, numerous fixes have been made to improve the performance speed of the system – making it considerably faster than previous iterations. First, as previously mentioned, the recalculation time period of the opinion system has been modified from every month to every three months, minimizing the number of times the method is used in a period of time. Secondly, the culture system now automatically compiles a sorted list of the economic scores of all of its constituents – as originally the system had sorted the list itself for each empire in the opinions recalculation procedure. While the current implementation still has flaws, it shows a remarkable improvement over previous versions.

* + 1. Unrest & Rebellion

Map

Description automatically generatedThe unrest and rebellion system closely follows the proposed designs, creating an environment where nations may break apart if they fail to manage the stability of their nation. Unrest may be gained through a variety of different actions, but the amount gained depends on the unrest multiplier. The unrest multiplier is decimal value which defaults to 1, and is incremented by:

* +0.25 if the nation has more provinces than its culture tech value, with a minimum of 5
* +0.25 if any provinces in the empire are at or over their unrest limit
* +0.1 if there are any active rebellions

Figure Varied unrest across a continent

This multiplier is then used in the calculation of the gained unrest for each applicable action, as such:

|  |  |  |
| --- | --- | --- |
| Condition | Applies to | Unrest Gain |
| National religion changed | Religious provinces that do not follow the new primary religion | Multiplier \* 1 |
| Province forcibly converted to national religion | Targeted provinces | Multiplier \* 1 |
| Primary culture of the empire changes | Provinces not of the new primary culture of the nation | Multiplier \* 0.5 |
| New ruler ascends to the throne | All held provinces | Multiplier \* 0.5 |
| New dynasty ascends to the throne | All held provinces | Multiplier \* 1 |
| Province occupied during war | Targeted province | Multiplier \* 0.5 |
| War ends with the empire losing more land than gained | All held provinces | Multiplier \* 0.5 |
| Empire fails to suppress a rebel uprising | All held provinces | Multiplier \* 0.1 |
| Capital city changed | All held provinces | Multiplier \* 2 |
| Spawn rebellion action taken by empire or stir unrest action taken by peer | Random set of provinces | Multiplier \* (1 + (ruler spawn rebellion personality value – ruler calm value)) |
| Culture technology increased | All held provinces of primary culture | -0.2 |
| Culture technology increased | All held provinces not of primary culture | -0.05 |
| High war exhaustion | All held provinces | Multiplier \* 0.1 |
| Unrest spread from nearby provinces | Adjacent provinces to active rebellion | 0.1 |

These conditions, as well as their multipliers and areas of effect, were determined primarily by manual testing – noting how much impact individual factors had and modifying their attributes until the results were satisfactory. Of particular note is the province occupation condition – which has no restrictions on how often it can occur. If a single province is repeatedly captured and lost in a war, it will quickly gain a lot of unrest and be susceptible to rebellion regardless of which nation controls it after the end of the war.

A picture containing text, clock

Description automatically generatedWhen a rebel group exceeds the unrest capacity of the empire – 5 + the cultural technology value of the empire divided by 5 – it will become active and choose a rebellion type[[4]](#footnote-4) – after which it will become able to spread and revolt as described in the design. When a rebellion is active, it can be seen on the national map as a darker shade of the empire it is rebelling against – allowing for easy review of what areas are in active rebellion without needing to switch map mode.

Figure A rebellious segment of the larger turquoise nation

1. Testing

After finishing the implementation of all specified simulation features, the simulation can now produce its intended output – a complete world map. This means that the required technical testing can be conducted to ensure the software is not prone to errors or significant faults. In the included testing document, a substantial number of tests were conducted to make sure the simulation is functional and not excessively error prone. The completed test document[[5]](#footnote-5) shows a notably low number of failed tests, so few in fact that the existing errors can be considered negligible.

It should be noted however, that the tests conducted only concern the technical requirements of the system, and do not refer to any non-fatal performance issues or user interface concerns. There are a number of known issues that fall into this category, moreso the latter, but none that are so impactful that they prevent the use of the software. These known issues are as follows:

* Opinion updating consistently causes software speed issues which gradually become more prominent as the years progress
* Selecting an ocean province may cause notification text to appear behind the targeted province
* Different resolutions are not yet fully tested. It is possible that some screen resolutions make interface elements appear off screen or overlap

1. Analysis & Comparison to Goals

Map

Description automatically generatedThe completed artefact shows a clear success at meeting the set goals, often meeting more than half over the course of a single simulation. One particularly notable example of this came in the form of a world henceforth referred to as Pangaea. As seen in the attached figure, Pangaea generated as a single great landmass, with a number of mountain ranges and deserts separating the world. Upon generating, population centres began to form around savannah biomes – notably in the easternmost savannah and desert as well as in the regions below the centre-left mountain range.

Figure Pangaea before the simulation began

The first nations developed in the west, right of the aforementioned mountain range, in a cultural group called “Lonian”. These early civilisations formed on coastal provinces, and remained relatively small and disconnected, preferring to expand towards coastal provinces. It took roughly 100 years before civilisation spread to other regions, forming in the eastern savannah based “Potamian” cultural region. While the nations in this region originally fell behind their western counterparts, a golden Map

Description automatically generatedage of technology allowed them to grow and develop.

Figure The Bellmon empire

In the year 543, this culminated in a series of conquests taken by the technological leader – the nation of Bellmon. Over time the nation was able to unify its cultural region, before even expanding into the lower “Luscan” territories. In less than a century, the Bellmon empire had created a sizeable hegemony spanning from coast to coast. Seen in the attached figure, Bellmon stood as the greatest empire in history. Of note is the way the geography shaped the empire. While the capital of Bellmon fell in the central-eastern provinces, they favoured southern military expansion rather than eastern colonialism. This can be attributed to the mountains and deserts present within the eastern region – making the Map

Description automatically generatedcosts of developing a permanent population much higher than the southern alternatives.

The Bellmon empire reigned supreme in the region for two centuries – before a string of weak rulers, monarch deaths and succession disputes caused their fall. The first revolts broke out in the northern provinces – cultural lands of the “Potamian” people, who, after building up resentment for the state after a number of conflicts with the green-coloured northern nation, began a cultural revolt. This conflict was undoubtably a success, with the revolution crushing their former sovereign and taking their homeland. After this defeat, a small separatist movement began to brew in the southern peninsula, eventually consuming much of the eastern empire. The empire was unable to protect their capital and was forced to relocate to the fertile southern regions. This allowed the eastern rebels to overwhelm their former masters – only being unable to overwhelm the eastern coast of the peninsula.

Figure The collapsed Bellmon empire

A map of the world

Description automatically generated

Figure The difference in technology as of 1069 – the start of the renaissance period in the history of this world

Map

Description automatically generatedThis empire collapse only exacerbated regional disputes, with the new splintered kingdoms fighting until they too fell to revolution. Throughout this whole period, the technological difference between the region and its neighbours only increased, and eventually one of the medieval nations, Petto, came into contact with less advanced western nations. Within fifty years the nation had expanded its influence greatly, taking the lands of less well-equipped states and even making their way to the cradle of civilisation. Eventually, a failed suppression of rebels in the west caused the “Cidian” and “Salano” provinces to raise the banners of revolution, and declare their own state. The Petto empire, already weak from conflict, was unable to hold their own against the revolution, and was forced to lose these holdings, an act that would only increase revolutionary spirit in other regions. The new states, holding the technology of their former masters, quickly found itself able to outcompete the other regional leaders. Both quickly asserted themselves as the great powers in the region, devouring weaker nations in the process.

Figure The Petto Empire

Shape

Description automatically generated with low confidenceA picture containing text, vector graphics

Description automatically generated

Figure The former Petto held states easily dominating other regional powers

Figure The 1100 Cidian and Salano revolutions

This rapid expansion westward spread not only the influence of the east, but the technology it held too. By 1200 the western kingdoms had began to learn from their oppressors, so much so that they began to be able defend against foreign powers. This technological difference allowed the west to rapidly create their own new power structure, and within two centuries the region had become an economic competitor to the east.

Map

Description automatically generated

Figure The western modernisation

Map

Description automatically generatedDuring this time, the eastern kingdoms, now finding significantly more difficulty in overpowering the central and western nations, turned their attentions to the north, which had not developed like the west. Within a short period of time, the nation of Lasm had established a great empire – one that remained relatively stable due to the greater increase in cultural technology since the days of Bellmon and Petto.

Figure The Lasm empire

A map of the world

Description automatically generatedBy the 1960s, the world had become largely complete. Wars began to become less common and the technology gap between nations fell sharply. The included map of the world during the 1960s shows the results of the simulation generating a modern world. The map shows a diverse number of nations, each shaped by the conflicts of the past.

Figure The map as of 1969

This simulation example overwhelmingly meets the goals set for the simulation, and during the course of this single simulation instance, was able to display the following goals as set in the methodology:

* Chronological accuracy
  + Dawn of civilisation – the slow formation of nations in the east and west, followed by the creation of great hegemonies like Bellmon
  + Bronze age collapse – the collapse of the Bellmon empire, and shattering into disputing states
  + Classical age – the disputing states form into new nations
  + Renaissance – the east begins to outclass its western neighbours technologically, and spreads west unopposed
  + Modernisation – Many of the former western occupied states gain independence and technology spreads across the world
* Impact of the world
  + Culture – The cultural difference between the Petto empire and its colonial possessions was a vital part in its collapse
  + Technology – The rise in eastern technology played an invaluable role in its ability to conquer the west
  + Geography – As can be seen in the early figures, the borders of Bellmon were defined by the mountains and inhospitable terrain surrounding it
* Believability
  + Mistakes – The overextended Petto state was destined to fall due to its indefensible structure and poor leaders
  + Collapse – Multiple times throughout the simulation

These noted items more than constitute the original intention of demonstrating at least half of the goals, and while this is a specific example, without fail all tests have shown a trend of completing the majority of the milestones – including those not seen here. In fact, all the different goals have been shown through testing to be possible – with only one having less impact than the other – religion.

Religion proved to be less prevalent than expected – with most of the conflicts over religion owing moreso to other factors. This is not to say that religion is not important in the simulation – the presence of religious unrest and rebellions is a decisive counterargument to this assertion – but due to the fact that cultural groups often contained only one religion, much of the conflict in which it played a part can be better attributed to cultural differences over religious issues.

1. Conclusion & Future Developments

In conclusion, the produced artefact meets its goals, and even exceeds expectations in areas such as rebellions and diplomatic relations, and therefore can be said to have met the hypothesis. The worlds that “Juris Civilis” generates are built by their history, and the ability to observe the factors that built the worlds provides insight into real history and civilisation. The simulation outright surpasses all its goals, and even incorporates some aspects not initially required. That said, were development to continue, there are some areas unrelated to the initial goals that could have seen significant improvement. These areas are as follows:

Opinions. As noted in the opinions segment, the updating of opinions causes a significant amount of slowdown, especially in the later stages. Were this project to be extended in the future, the first priority would be the fixing of this flaw – either by spreading out the opinion updating procedure or having it work on a separate thread entirely.

Enclaves, referring to disconnected land within the borders of another nation, are a common occurrence in the simulation – a product of the focus of nations on expanding into vital territory over maintaining clean borders. While this is often punished by the simulation – granting the enclaved nation the ability to use said enclave as a jumping off point for future conflicts, the AI has no concept of preventing this from occurring, and therefore later maps contain significant amounts of enclaves. In future developments, it is likely that enclave provinces would gain a new type of rebellion, able to defect to their encompassing nation, in order to maintain strict borders.

Finally, naval procedures are entirely unmodelled by the simulation – aside from aspects like coastal regions having more economic value. At the start of development, naval systems encompassed a form of “stretch goal”, in which it was intended to be implemented if the project was completed quicker than anticipated. This unfortunately was not the case, and so navies were never truly implemented into the simulation, though this would be another priority if development were to continue. The groundwork for naval simulation does exist in the final product – oceans borders are defined properly and even have their own names, but do not currently have any specific purpose.

1. Bibliography

Adams, T., 2009. *Dwarf Fortress Talk #4* [Interview] (3 November 2009).

Adams, T., 2010. *Nullpointer Dwarf Fortress Interview* [Interview] (1 July 2010).

Adams, T., 2013. *Dwarf Fortress Talk #21* [Interview] (23 May 2013).

Alain Fournier, D. F. C., 1982. Computer rendering of stochastic models. *Communications of the ACM,* 25(6), pp. 371-384.

Diamond, J., 1997a. *Guns, Germs and Steel.* s.l.:s.n.

Diamond, J., 1997b. A Natural Experiment of History. In: *Guns, Germs and Steel.* s.l.:s.n., p. 68.

Dwarf Fortress Wiki, n.d. *Dwarf Fortress Wiki - Calendar.* [Online]   
Available at: https://dwarffortresswiki.org/index.php/DF2014:Calendar

Firaxis, 2010. *Civilization 5 Home Page.* [Online]   
Available at: https://civilization.com/en-GB/civilization-5/

Forsberg, K. & Mooz, H., 1991. The Relationship of System Engineering to the Project Cycle. October.

geacron.com, 2011. *Interactive World History Atlas since 3000 BC.* [Online]   
Available at: http://geacron.com/home-en/

Gorman, J., 2018. *Github - Iron Age.* [Online]   
Available at: https://github.com/JaVonox/Iron\_Age

Holdridge, L., 1964. *Life Zone Ecology.* s.l.:Tropical Science Center.

Kendeigh, S. C., 1961. *Animal Ecology.* s.l.:s.n.

Lugo, A. et al., 1999. The Holdridge life zones of the conterminous United States in relation to ecosystem mapping. *Journal of Biogeography.*

McMillen, E. & Himsl, F., 2011. *Steampowered Store - The Binding of Isaac.* [Online]   
Available at: https://store.steampowered.com/app/113200/The\_Binding\_of\_Isaac/

Microsoft, 2000. *Microsoft Unveils Vision for Next Generation Internet.* [Online]   
Available at: https://news.microsoft.com/2000/06/22/microsoft-unveils-vision-for-next-generation-internet/

Microsoft, 2021. *Language Integrated Query (LINQ) (C#).* [Online]   
Available at: https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/

Microsoft, 2021. *Windows Forms .NET 6 Documentation.* [Online]   
Available at: https://docs.microsoft.com/en-us/dotnet/desktop/winforms/

Mossmouth, 2013. *Spelunky World.* [Online]   
Available at: https://spelunkyworld.com/

Paradox Development Studio, 2013. *Paradox Plaza.* [Online]   
Available at: https://www.paradoxplaza.com/europa-universalis-iv/EUEU04GSK-MASTER.html

Paradox Development Studio, n.d. *Europa Universalis Wiki.* [Online]   
Available at: https://eu4.paradoxwikis.com/Province

*SUN MICROSYSTEMS INC v. MICROSOFT CORPORATION* (1999).

Tarn Adams, Z. A., 2006. *Bay 12 Games.* [Online]   
Available at: http://www.bay12games.com/dwarves/

Unity Technologies, 2005. *Unity.* [Online]   
Available at: https://unity.com/

Unity Technologies, n.d. *Unity Documentation.* [Online]   
Available at: https://docs.unity3d.com/ScriptReference/Mesh.html

1. See Segments 2.5 and 2.6 [↑](#footnote-ref-1)
2. See segment 2.4.3 [↑](#footnote-ref-2)
3. Referring to the design document property adjustments [↑](#footnote-ref-3)
4. See segment 6.1.6 [↑](#footnote-ref-4)
5. See attached testing document [↑](#footnote-ref-5)