It’s the year 2868 on the Galactic Calendar

and War. Is. Coming.

# Summary

Play as the Entente, a coalition of three empires and play out war simulations to come up with strategies which can be used in the coming conflict. In the game, Play Singleplayer, Or Multiplayer with friends and gather resources in order to build units to destroy the enemy base in this Real Time Strategy game entitled, Empires of the IV!

# The JourneY - difficulties & Solutions

## Cross Platform

Being a cross platform game, Empires of the IV initially started with the high difficulty point of needing to figure out the best way to have the game play exactly the same across all my platforms.

To solve this I could have gone the route of programming the game once for each platform, but instead I settled on an approach which involved the creation of a singular toolkit, my Killerrin Studios Toolkit, which wraps platform calls to my own specific API and cross platform implementation. By doing this, I can take a miniscule hit in terms of performance in favour of allowing me to not have to ship multiple builds of my game each with different implementations of the platform specific APIs.

To help with cross platform, I decided to go with MonoGame for the Framework and Json.Net for quick cross platform serialization and deserialization and built my Anarian Game Engine with the aim of further improving the cross platform abilities

## Multiplayer

Utilizing the Networking wrapper of my Killerrin Studios Toolkit, I was able to quickly implement cross platform multiplayer in Empires of the IV. To handle Multiplayer in game, I chose to utilize TCP (due to lack of time to develop a proper UDP reliance system) and a Lockstep pattern which utilizes pre-set Commands which can influence almost all portions of the game.

To solve the problem of games going out of sync, I gave authority to the Host whom is the sole provider of specific types of commands, such as Killing a Unit or Destroying a Factory.

To keep from flooding the network, commands are separated into outgoing and incoming lists which are properly synchronised across threads and aggregated into the main Command List at the end of each frame; the outgoing being sent over the network during aggregation

# Technology

* C# + MonoGame
* Killerrin Studios Toolkit Custom Made Toolkit for easy Cross Platform development
* Anarian Game Engine Custom Made Game Engine with several built in tools!
* JSON.net Cross Platform NUGET Package for easy serializations