Beware of Strange Warp Points

A 7-Day Roguelike inspired by Space Empires

Overview

Beware of Strange Warp Points (or BOSWP) is my (Ed Kolis') entry in the 2014 7-Day Roguelike Challenge. It's based on the Space Empires games by Malfador Machinations.

System Requirements

BOSWP requires the .NET Framework v4.0, which can be downloaded from Microsoft's website. If you're not using Windows, you should be able to run it using Mono, but I haven't had a chance to test that yet.

Backstory

Captain's log, stardate 24XX…

After completing our refit at Earth Orbital Shipyard, we received a priority communiqué from UNE central command:

"Captain! The fleet has been decimated! You're our only hope to stop the Jraenar invasion! You must destroy all the cloaked Jraenar shipyards that have been deployed throughout Terran space so that the Jraenar war machine grinds to a halt! Use minerals salvaged from destroyed Jraenar vessels to upgrade your ship with new weapons, engines, and shields at allied colonies! Good luck and godspeed!"

What?! The Jraenar have been our longtime allies! Has Master General Jar-Nolath gone mad? Those Jraenar shipyards… if they're not destroyed immediately, we could suffer the fate of the Eee! Phong colonies scattered throughout Eee space… stealthy military buildup… surprise attack… no more Eee… Has no one learned from the dastardly deeds of Emperor Eorg?

Gameplay Mechanics

The objective of BOSWP is to find and destroy all of the 10 cloaked Jraenar shipyards scattered throughout Terran space. Because they are cloaked, you can only find them if they are very close to your ship, though you can extend your sensor range by upgrading your ship with Tachyon Sensors. Once you find a shipyard, it will stay visible, as it is immobile. The shipyards will periodically produce warships to defend themselves. Due to maintenance costs, shipyards will build slower the more shipyards and enemy ships exist; this serves as a difficulty balancing mechanism.

Weapons automatically fire when reloaded; you don't need to bump enemy ships or press a fire key. Shipyards are prioritized first; then the closest ships. When you destroy a shipyard or enemy ship, you will salvage resources from the wreckage. The amount of resources salvaged depends on the price of

the intact components (for ships) or the amount of saved build resources (for shipyards). Shipyards have a fixed HP pool, while ships (both enemy ships and your ship) are destroyed when any of the following conditions occur:

- All components are destroyed
- The number of crew on the ship falls below the ship's hull mass
- The ship's speed falls to zero (i.e. the amount of thrust produced by the engines falls below the ship's hull mass)

Using the resources you salvage, you can buy upgrades at allied colonies. You can find allied colonies by bumping into planets. Only a few planets have colonies, so explore early and often! Different colonies have different components available for sale, so even if you find one colony, it is worth searching for more. Colonies will also repair your ship's hull damage for free, though components that are completely destroyed are lost permanently; you will need to buy new ones to replace them.

Travel between star systems is accomplished using warp points. Bump a warp point, and you will be transported to the adjacent star system. Like in Portal, velocity in equals velocity out, so if you enter a warp point from the northwest, you'll come out on the southeast. If you try to move through a warp point and find yourself unable to, it probably means an enemy ship is blocking the way. Wait a bit and try again, or approach it from a different angle.

Controls
The controls in BOSWP are fairly simple:

- Numpad: Move (press 5 to sit still for a turn). Or, if you don't have a numpad:
    - Arrows: Move orthogonally
    - Home/End/PageUp/PageDown: Move diagonally
    - Space or .: Sit still for a turn
- Click system map: Scan clicked ship or shipyard (requires that it be within scanner range, indicated by a white square)

The rest of the UI (e.g. the shop interface) is navigated using the mouse.
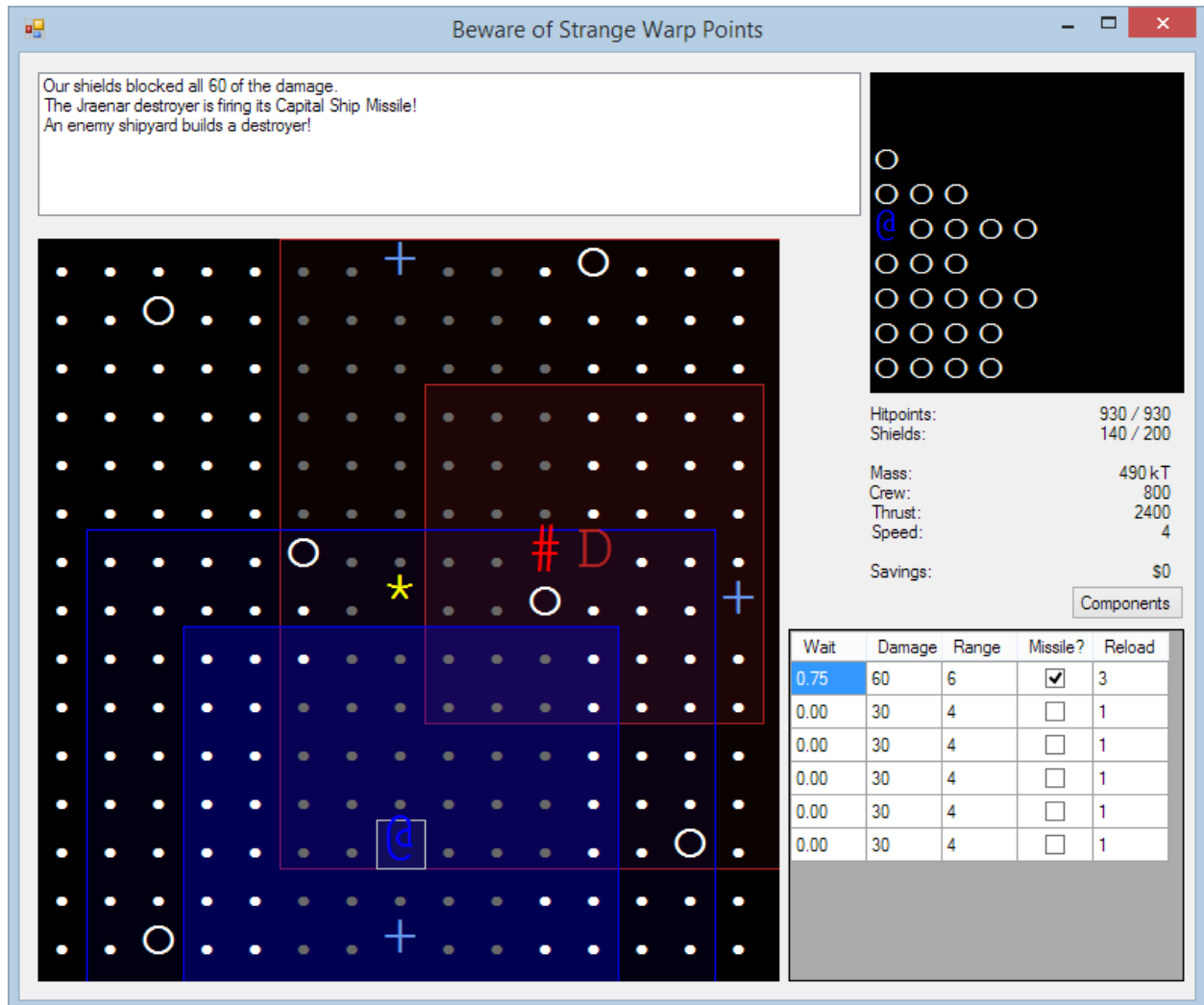
UI Layout
The main UI is divided into 5 sections:

- Message log (upper left) – Displays any messages that you might need to read this turn. Newer messages are displayed on top.
- Galaxy map (upper right) – Displays a map of the galaxy. Each symbol represents a star system.
- System map (lower left) – Displays a map of the current star system. Each symbol represents a sector.

- **Stats (center right)** – Displays basic stats about your ship. More detailed stats and a list of components can be brought up by clicking the "Components" button, or clicking your ship on the system map.
- **Weapons list (lower right)** – Displays information about the weapons equipped on your ship.

You can get information on almost any UI element by hovering your mouse over it.

Galaxy Map Symbols

On the galaxy map, each star system is represented by a symbol indicating the "most interesting" object contained within. Objects are listed below in order of interest:

- Blue @: Your ship
- Red #: Enemy shipyard
- Blue o: Allied colony
- White o: Unexplored planet

- Gray *: Sector that has not yet been swept with sensors
- Dark gray .: Nothing of interest

Enemy ships are not shown on the galaxy map because they can move about.

System Map Symbols

Various objects are also represented with symbols on the system map:

- Blue @: Your ship
- Red #: Enemy shipyard
- Dark red letters: Enemy ships
- Aqua +: Warp point
- Yellow *: Star
- Blue o: Allied colony
- White o: Unexplored planet
- Gray o: Uninhabited planet
- White .: Sector that has not yet been swept with sensors
- Dark gray .: Sector that has already been swept with sensors

Additionally, the ranges of your ship's weapons are indicated on the map with blue squares, while the ranges of enemy weapons are indicated with red squares. Brighter red or blue indicates that more weapons are in range. Your scanner range (note that scanners are distinct from sensors) is indicated with a white square. You can click a ship that's in scanner range to scan it – you get the same report you get by pressing the "Components" button, but for that ship instead of your own ship.

Stats and Abilities

There are a variety of components in the game, all with their own stats and abilities. Here's an overview of what all those stats mean:

- Hitpoints: The basic durability of the component. When this reaches zero, it's destroyed, and is no longer usable. The mass of the ship is also reduced by the mass of the component when this occurs.
- Shields: Additional protection (i.e. bonus hitpoints) provided by the component. All shield points on a ship must be eliminated before the ship's components can be damaged.
- Evasion: Chance to evade direct fire (non-missile) weapons. If a ship has multiple components which grant evasion, they stack with diminishing returns. So a ship with two components granting 10% evasion apiece, the total evasion is 19%, not 20%. The maximum evasion for a ship is 99%, but this is practically impossible to reach with the standard set of components!
- Point Defense: Behaves identically to evasion, except it works against missile weapons instead of direct fire ones.

- Emissive Armor: Blocks (i.e. completely prevents) a random amount of damage per hit up to the ability amount, provided the shields are penetrated. So if a ship has 40 points of emissive ability, then it will block 0-40 damage per hit.
- Scanner Range: Extends the ship's scanner range. Ships within this range can be scanned in detail by clicking on them. You also get detailed reports of damage against ships within your scanner range; otherwise you don't know exactly which components got hit.
- Sensor Range: Extends the ship's sensor range. This is the range at which cloaked shipyards can be detected.
  - Note: Your ship gets 1 point of sensor range for free, so you can still find enemy shipyards without bumping into them even if your tachyon sensor is destroyed.
- Mass: The component's mass, in kT. The heavier your ship is, the more crew it needs, and the more engines it needs per point of speed.
- Crew: Crew granted by the component. Ships require crew greater than or equal to their mass. If this requirement is not met, the ship is destroyed.
- Thrust: Engine thrust granted by the component. Ships require thrust equal to their mass for each point of speed. If a ship's speed falls to zero, it is destroyed.
- Weapon stats: Weapons have their own special set of stats:
  - Damage: The amount of damage inflicted by this weapon.
  - Range: The range of this weapon, in sectors.
  - Reload Rate: Amount of time required to reload this weapon.
    - Note: This is in game time, not player turns, so if you have speed 4, and a weapon has reload rate 3, then you will need to move or sit still 12 times for the weapon to reload.
  - Missile?: Is this weapon a missile, or direct fire? Missile weapons can be shot down with point defense; direct fire weapons can be evaded.

Modding

BOSWP can be modded using JSON files. You can define new components and enemy ships, and configure the player ship and general game settings by editing the four provided mod files. The structure of the files should be fairly self-explanatory, but there are some helpful tips to keep in mind.

Components.json

This file defines all the types of components in the game. It contains an array of objects, with each object representing a type of component. Abilities and stats are represented by properties of the objects. Weapon stats are grouped together in a sub-object. It's not necessary to limit a component to only one function; you can certainly make a component that is both an engine and a weapon, for instance. (Want to make a Thraddash afterburner for your Star Control mod? Go for it!) Be careful that no two components have the same name, or EnemyShips.json and PlayerShip.json will have trouble finding them!

EnemyShips.json

This file defines all the types of enemy ships in the game. It contains a dictionary, with keys being strings (names of ships) and values being arrays (lists of names of components, from Components.json). The symbol used for each ship on the system map will automatically be taken from the first character of the ship's name. Be careful that all the components referenced here are actually present in Components.json! Also be careful that your ships have enough crew quarters and engines, or they will be destroyed instantly when constructed.

PlayerShip.json

This file defines the components found on the starting player ship. It contains an array of strings, each string being the name of a component from Components.json. The same caveats from EnemyShips.json apply here.

Settings.json

This file defines general game settings. It contains a dictionary with 7 entries:

- GalaxyRadius: The radius of the galaxy (0 => 1x1, 1 => 3x3, 2 => 5x5, etc.) Each star system takes up one square on the galaxy map.
- Systems: The number of star systems in the galaxy. Make sure that GalaxyRadius is large enough to accommodate this many star systems!
- SystemRadius: The radius of each star system, in sectors. This must be at least 2 to leave room for a star, planets, and warp points.
- PlanetsPerSystem: The number of planets per star system. For pathfinding purposes, planets will never be generated directly adjacent to the star, other planets, or warp points, however, so if you set this too high, you might not get that many planets.
- ColonyChance: The percentage chance of finding a colony when exploring a planet.
- EnemyShipyards: The number of cloaked shipyards that exist in the game. Destroy them all to win!
- EnemyShipyardBuildRate: The base build rate for enemy shipyards. This will be reduced by a factor dependent on the number of enemy shipyards and ships present in the game, so as to provide a smooth difficulty curve. The factor is equal to 10 * Shipyards + Ships, so if there are 5 shipyards and 20 ships, the build rate will be reduced by a factor of 70.

Credits

- Ed Kolis: Game development
- James Newton-King: JSON.NET library
- Nick Dumas: Difficulty curve concept
- Aaron Hall: Original Space Empires games