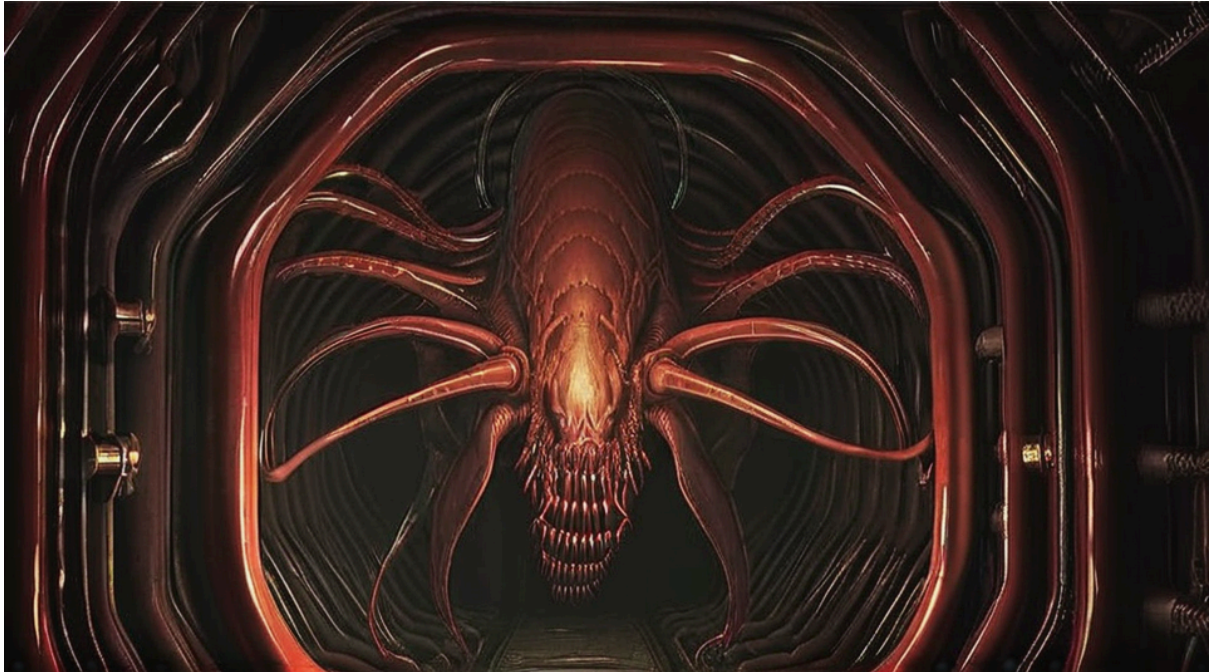


# Alpha Release



# Abyssal Isolation

09.01.2023

## **Submechanophobia**

Kim Niclas Simon

Leonard Keil

Pascal Neubert

Ruben Pfeiffer

## State of the Game

In the following we (once again) show our layered task breakdown and comment on our current implementation for each feature. We omitted any features that were already completed for the interim report and did not change much since then:

### Functional Minimum

- Reactor Mechanics

Reactor Mechanics have been implemented using a simplified, physics based simulation system. Water and reactivity from the reactor are turned into steam, which can be converted into electricity for the various devices around the submarine.

Currently the reactor can be controlled from multiple different consoles, which can be freely placed in the world, connecting automatically to the reactor manager.

Alternatively machines can be placed in a similar fashion, with the major change being that they only affect one of the reactor stats at the time. Some machines might also be passive in nature, serving the purpose of displaying the reactor state diegetically within the game world. One example of this would be a whirring machine or bursting pipe.





## Low Target

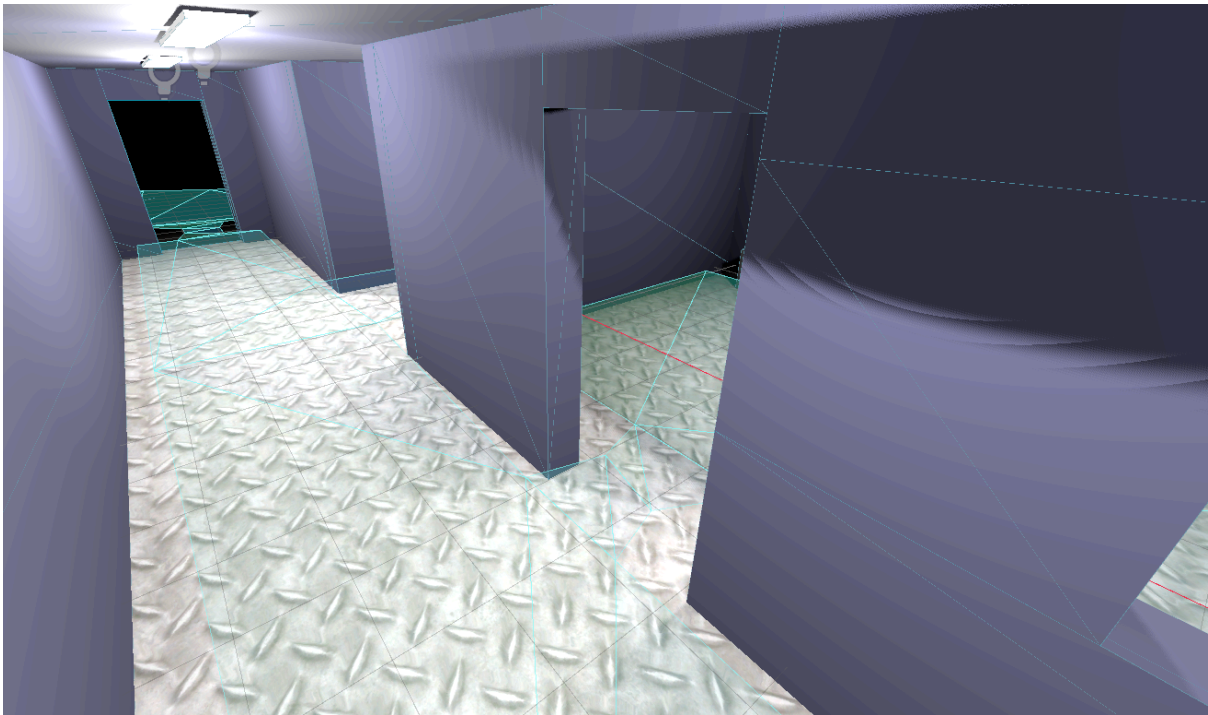
- Basic Creature Movements
- Creature Design with Spring System
- Lights
- ~~Oxygen~~ Sanity
- SFX

The creature moves throughout the world using mesh based navigation, utilising the A\* algorithm to find its targets. Nav-meshes can be pre-generated for every individual section of the UBoat, or generated at runtime from the randomly assembled model.

Currently it has a number of different behaviour states, which it switches between based upon its perception of outside influences

- Hiding:
  - The monster stays at its current location, not moving, waiting for something to happen.
- Roaming:
  - The monster randomly travels to different waypoints around the map.
  - This state is randomly triggered after a certain time has passed to make it more challenging for the player.
- Searching:
  - Similar to roaming but with a faster movement speed.
  - Triggered when an event like a loud sound is telegraphed to the creature.

- Attacking
  - If the creature is close to the player and has a direct line of sight, it will rush towards them, killing them if they come too close.

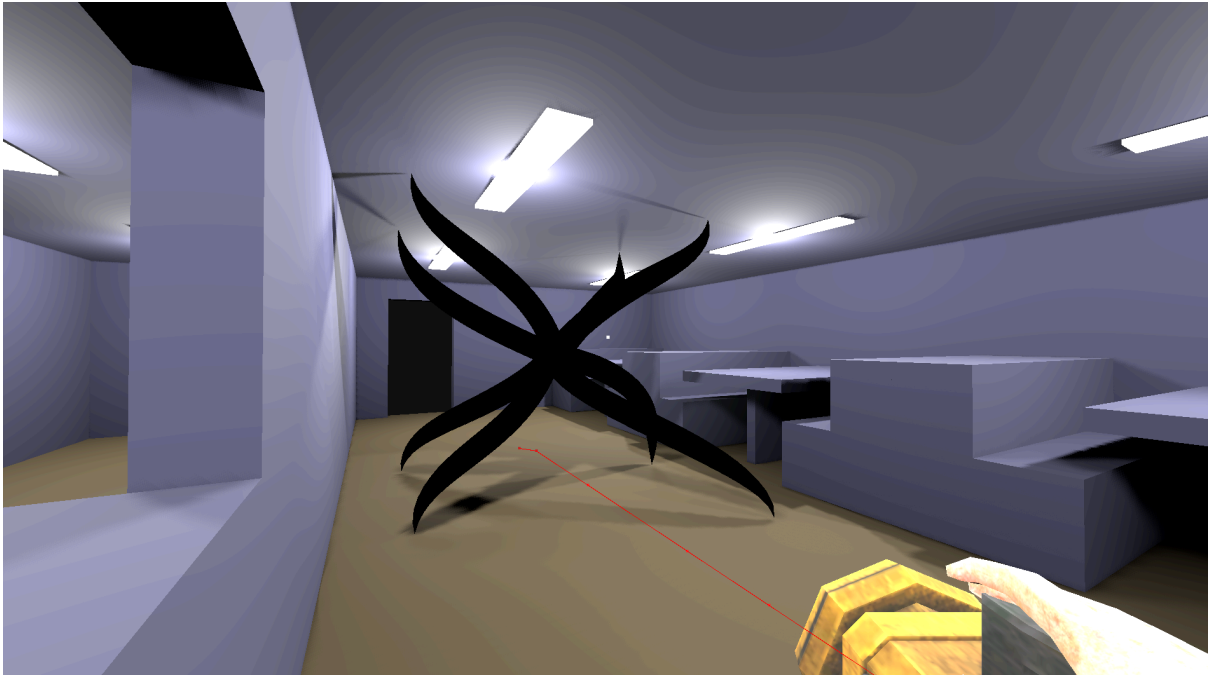


The creature's highlight is its signature tendril look, which is achieved through a complex dynamic animation system.

All tendril meshes are generated on startup based upon preset parameters like width, resolution and vertex count. A skeleton bone structure is then generated from the same parameters to fit the mesh and rigged to the mesh. Bones can either be entirely independent of each other or parented in a structure typical for animating models in traditional CG software like blender.

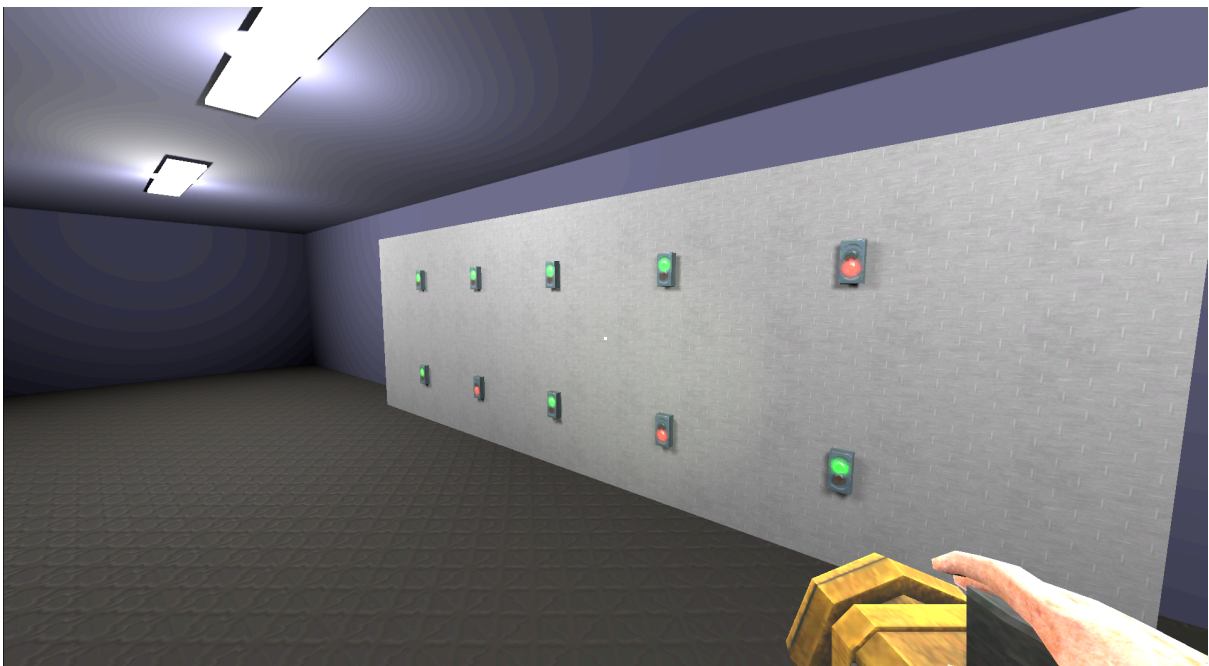
Each tendril is then fitted to a 3D curve, which can be adjusted by any animation system to bend the individual tendrils into a desired shape.

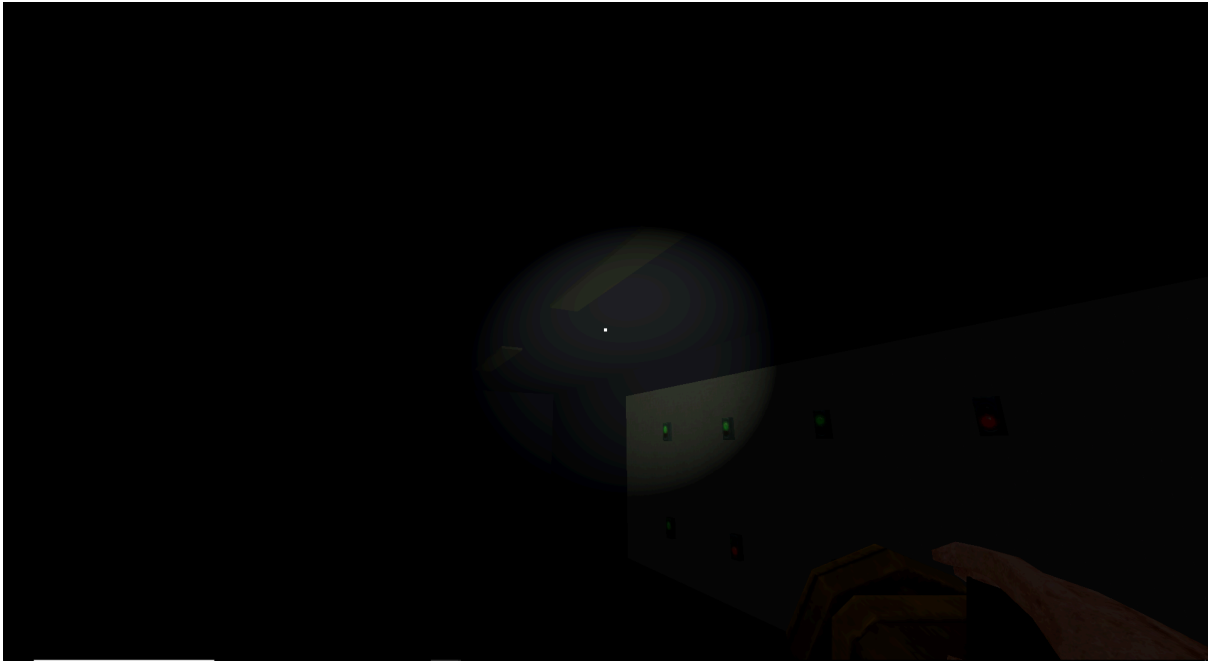
For the leg animations the creature has an array of emitters, which can hold and control a configurable amount of tendrils. The legs are animated by bending the individual appendages' curves using a self made inverse kinematic system.





We have expanded our light system significantly, adding multiple lights in every created room. Each light has multiple different states, affecting coloration and brightness, that by default change when the submarine sinks or rises (more on this later in this document). Additionally, all lights can be controlled manually by the player using a console in the electrical room of the ship, although this manual control currently only supports the two states of full brightness and no light at all. Still to be done before playtesting is the integration of the light controls with the reactor mechanics to enable the power management aspect of our game for the player.





We have slightly changed our original plans for an oxygen system towards a general “sanity” system. This should reflect the player character’s deteriorating mental state while trapped on the submarine deep under the water, affected not only by potential oxygen shortage but also the general pressure and stress of the situation. Our current implementation of this system is closely tied to the submarine’s depth level, with sanity falling while the submarine is deep under the water and slowly regenerating while the submarine is close to the surface. This is supposed to balance out with the creature’s higher activity levels at higher depths, forcing the player into a tradeoff. Effects of low sanity currently include ominous noises (e. g. fake footsteps, water-like whooshing) being played, and also loss of the game if sanity reaches its lowest level.

We have also added many previously generated as well as new sound effects to the game, so we are now only missing some creature noises and background music for a complete sound experience.

### Desired Target

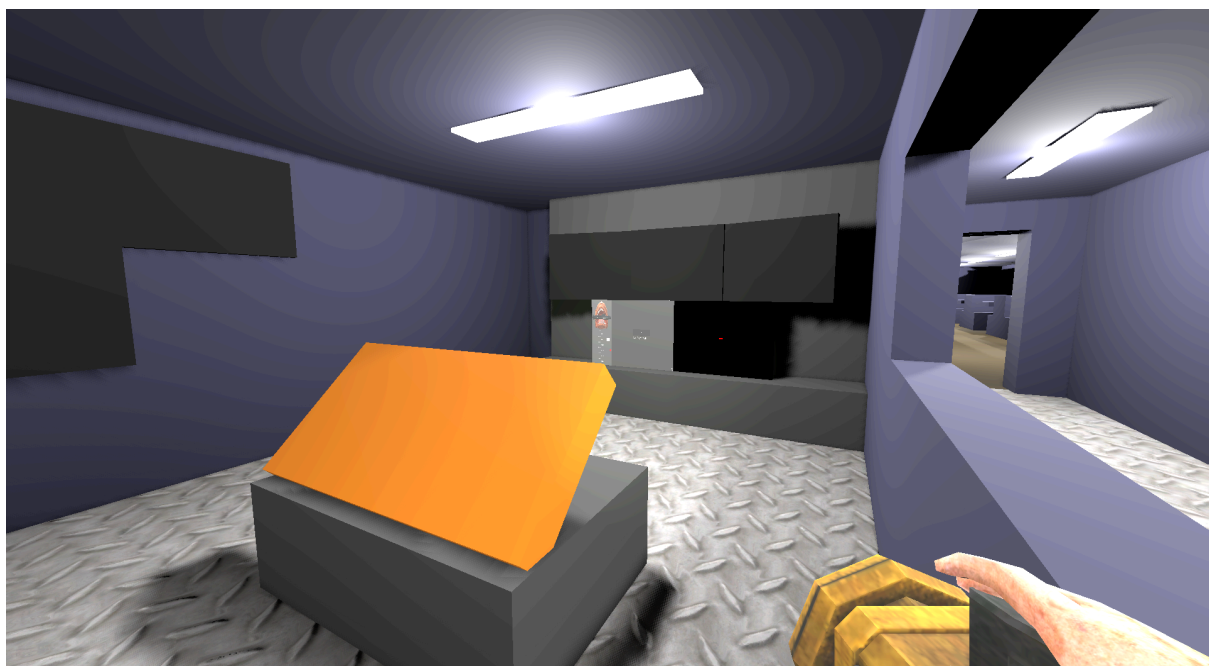
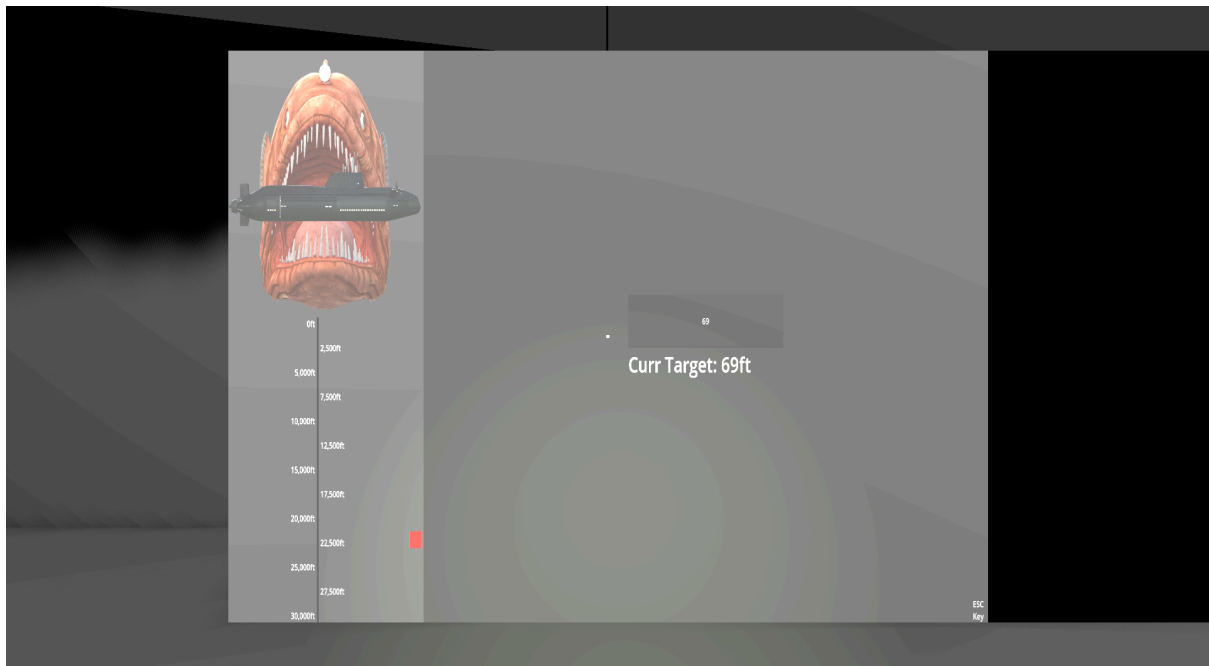
- Puzzles
- Procedural Map Generation (incl. more rooms)
- Equipment
- Improved Creature Mechanics + Hiding Spots
- Reactor Mechanics Rework

### High Target

- Story, NPCs
- Alternate Endings

- Submarine Up & Down Mechanics
- Atmospheric Soundtrack
- More Details on the Submarine

Due to our redesign of the sanity system we decided to prioritize the rising and sinking mechanics of the submarine more than we initially anticipated, as the two systems are closely related now. The submarine's depth control can be accessed by the player in the bridge room and set to a target altitude between 0 and ~30.000 ft. As previously stated this affects the sanity level and before playtesting must still be integrated into the creature's behaviour as well.





## Extras

- Submarine Dive Up and Short Landtrips
- Creature AI
- Cutscenes

Not implemented.

## General Notes

Since we were significantly behind our planned schedule for the interim report milestone, we had a lot of work to catch up on. While we made good progress on most of our planned features for the low and desired target, a couple of things are still missing and need to be implemented before playtesting makes sense. The main areas of focus here include the creature and its related features and integrating the reactor into the general power management of the submarine. We are confident that we will achieve a fully playable state of the game with these features by the end of the current week and can commence playtesting afterwards.



