Startaste

Release 1.0.0

Jan 28, 2022

Contents

Plan

1.1 What is "this" exactly?

The Startaste OS is just yet another operating system. Its goal is to make an operating system that is fast and efficient. To do this we use some newer methods that older operating systems don't have integrated into them from the start.

1.2 What technology is used?

Originally written in pure assembly, Startaste is now written in C. The OS is written on a ReactiveX IO structure, completely custom built. Sphinx Docs is used for this documentation. And GitHub is used for version control, issue handling, and improvements.

1.3 What are the features?

Startaste uses a custom ReactiveX structure so that all events are much faster. Creating more user interactive environments are encouraged and easier using this structure. Take a look at either the ReactiveX website, or the in-depth explanation of Startaste's ReactiveX structure. The basic kernel of Startaste is very minimal and will only start a few drivers depending on the compiled image. To overcome this, we've built a few drivers to start out. However, it is possible that you may need to make more drivers for specific devices or take some from the community.

Kernel

2.1 How did we get Here?

The kernel is the brains of an operating system. That is that, everything can be traced back to the kernel eventually. The first thing that a computer will do when booting up is run a 512 byte section of code called a bootloader. More information on the bootloader is available on its page. When the kernel starts up it will run a specific subroutine that initializes any possible variables and the ReactiveX implementations. Using ReactiveX concepts, the kernel will create places in the memory called Stars and Planets. These are equivalent to Observers and Observables, respectively. More on these two in the next section, More on the ReactiveX. Since Startaste is fully customizable, it is entirely up to the user what stars and planets are initialized. For a very basic operating system, a display and keyboard can be initialized such that only a terminal type window will show. Using the builtin REPL, users may do a variety of simple and complex operations.

2.2 More on the ReactiveX

ReactiveX has bindings for many different languages that can be found on their website. The main idea is that there are two types of objects or structures. The observables are things such as keyboards, mice, hard disks, or other IO devices. These will have an internal buffer, this doesn't truly exist, but rather data is distributed by calling observers. The second type of object is the observer, these are things such as displays, command lines, or programs. The observer will bind itself to one or more observable which when receiving data, will call the `next` command of the observer. Startaste does things a little different from ReactiveX. First, there is no current true binding for a C language ReactiveX, thus the structures are built from scratch. Second, the Observables and Observers are called Stars and Planets respectively.

CHAPTER 3

Bootloader

3.1 What even is a bootloader?

Text filler is here, so replace it later.

3.2 Important remarks

Text filler is here, so replace it later.