STARBEAM:

REAL TIME LOCATION SHARING

<< INTERNAL ALPHA RELEASE >>

Collective Internal Transmission

Document author : <D:\IHaveNoLife>
Cowriters : <C:\BenCo> <F:\Zaff>

Date of creation : 2020-09-01
Version number : 1.0

>> INTRODUCTION 2
>> COMPUTER SECURITY 3
>> INGAME SECURITY 3
>> INSTALLATION 3
>> .EXE INSTALLATION 3
>> SHIP MODIFICATIONS 5
>> CREDITS 6

>> INTRODUCTION

Starbeam is a software service providing real time location sharing for Starbase.

How it works:

1. Player uses a spaceship equipped with **ISAN** to display coordinates.
2. Starbeam regularly takes a screenshot of Starbase, including the ISAN display.
3. Starbeam utilizes optical character recognition to read ISAN coordinates.
4. Coordinate data is sent to Starmap. (Screenshots stay on your computer).
5. Your location can be visibly shared to others, updating in real-time on Starmap.

This is a closed beta release.
There will be problems, bugs, etc. Please report them as issues on the github - https://github.com/Collective-SB/StarBeam-Client - or PM IHaveNoLife#6972 on Discord.

In the future we plan to switch to a web client so no installation is necessary, and use a machine learning model to allow for greater flexibility when placing ISAN panels. We need lots of screenshots for that machine learning model, so if you're in CA please check #ca_general in the Collective Discord server for a tutorial of how to help (Link).

Machine Learning Model	In Development

Web client	In Planning

>> STARBEAM VERSIONS

Starbeam 1.0 operates as a **desktop app**, written in Python. There's also a web version planned, but not yet available. See table below for a comparison:

Starbeam	Exe	Source	Web
Available now	✅	✅	❌
Visible code	❌	✅	✅
Open source	✅	✅	✅
Easy setup	✅	❌	✅
No install	❌	❌	✅
Loads quickly	❌	✅	✅

>> DATA SHARED

- While Starbeam is active, the program will take screenshots of Starbase once every 0.2 seconds. These remain on your own computer, never going anywhere.
- If the screenshot contains a readable ISAN screen, coordinate data is produced.
- Coordinate data is periodically sent to the Starmap web service. (numbers).
- By default, the coordinates make points visible on the layer "Player Positions".
- Player Positions are visible to all Collective members. For any planned mission, you may be given a more secure layer ID to use.
- Your position will remain visible for 30 minutes after last activity.

Your position is uploaded to a Collective-only Starmap layer called Player Positions, and your point disappears from the map after 30 minutes of inactivity. If you're doing a Swarm or Logistical operation, you may be given a layer ID for more confidential sharing. **Sharing to private layers is not possible due to technical limitations.**

>> INSTALLATION

There are two methods of installation: **.exe installation** and **installing from source.** Installing from .exe is much easier but you don't know what you're running, while installing from source allows you to inspect the code before you run it.

Currently only Windows is officially supported, but we have tried to keep it cross platform throughout development so Linux users should be able to install from source easily.

If you have any problems during installation, feel free to PM me (IHaveNoLife#6972)

>> .EXE INSTALLATION

1. Make sure Starbase is running in **borderless window** mode.
2. Install **Tesseract-OCR** (by Google, so it's safe) from https://github.com/UB-Mannheim/tesseract/wiki
3. Take a note of **where** it installs to. You'll need it later.
4. Download the Starbeam source zip file from https://github.com/Collective-SB/StarBeam-Client/releases
5. **Unzip** the file.
6. Open **`settings.yaml`** in Notepad and replace the **pytesseract** path with the **installation location** noted earlier. Replace all the backslashes in the path (\) with double backslashes (\\). If you don't know the path, it's probably at `C:\Users\YOUR_USERNAME_HERE\AppData\Local\Tesseract-OCR` or `C:\Program Files\Tesseract-OCR`, but you might need to do some trial and error.
7. Change your **username** in settings.yaml
8. Run the exe! If it crashes the tesseract path is probably not right, so experiment a bit to get it to work (like putting tesseract.exe on the end.) Note that it takes the exe a while to start because it needs to unpack Python first.
9. You should appear on Starmap!

>> INSTALLING FROM SOURCE

1. Make sure Starbase is running in **borderless window** mode.
2. Install Python with Pip. See https://wiki.python.org/moin/BeginnersGuide/Download

3. Go to the github (https://github.com/Collective-SB/StarBeam-Client/releases) and download the latest release's source code.
4. Unzip the source code.
5. Run the following command to install dependencies:

```
pip install WXPython yaml requests pyautogui pytesseract pillow
```

Linux users should replace "pip" with "pip3".

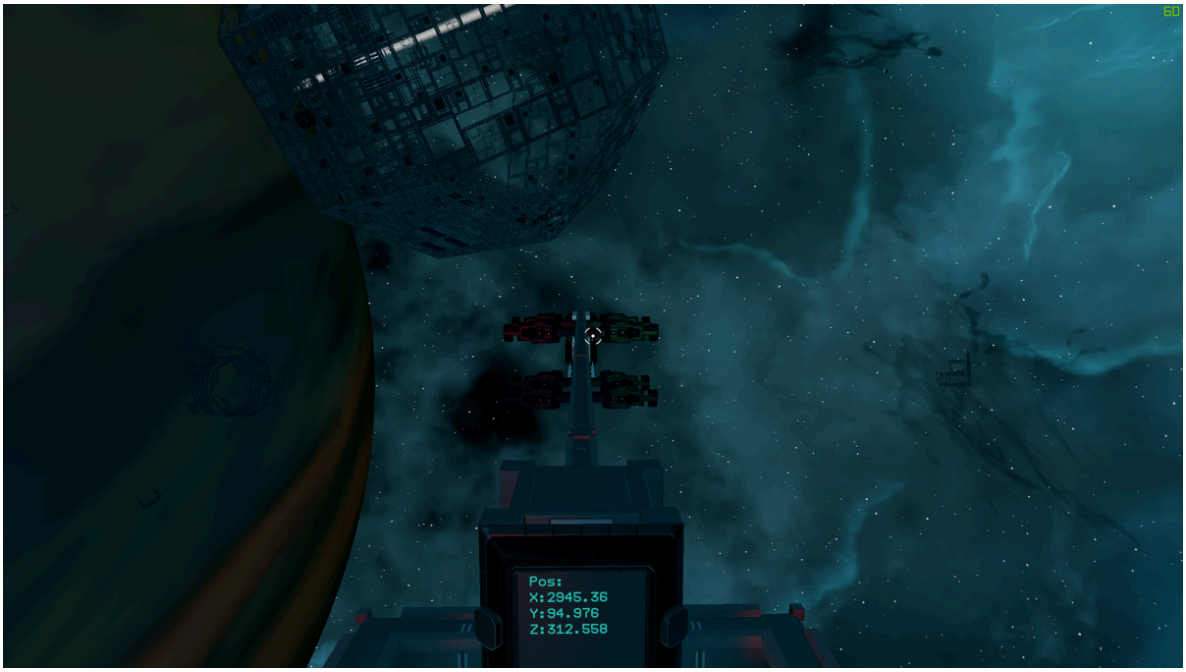
6. Install Tesseract-OCR (https://github.com/UB-Mannheim/tesseract/wiki for windows, https://github.com/tesseract-ocr/tesseract/wiki for Linux)
7. If it mentions where it installs in the install process, note it down. You'll need it later.
8. Go back to where you downloaded the source code and unzip it. If you're installing from source because you're suspicious of my code, now is when you should read through it. There isn't that much and it's Python, so it shouldn't take too long.
9. Navigate to the source code directory, then the client directory.
10. Open **`settings.yaml`** in Notepad and replace the **pytesseract** path with the **installation location** noted earlier. Replace all the backslashes in the path (\) with double backslashes (\\). If you don't know the path, it's probably at `C:\Users\YOUR_USERNAME_HERE\AppData\Local\Tesseract-OCR` or `C:\Program Files\Tesseract-OCR`, but you might need to do some trial and error.
11. Now replace the display name in that same file with your Starbase name.
12. Now run the program (main.py)! If it opens up in a code editor, right click and open with Python. You should be done!
13. You should appear on Starmap!

>> SHIP MODIFICATIONS

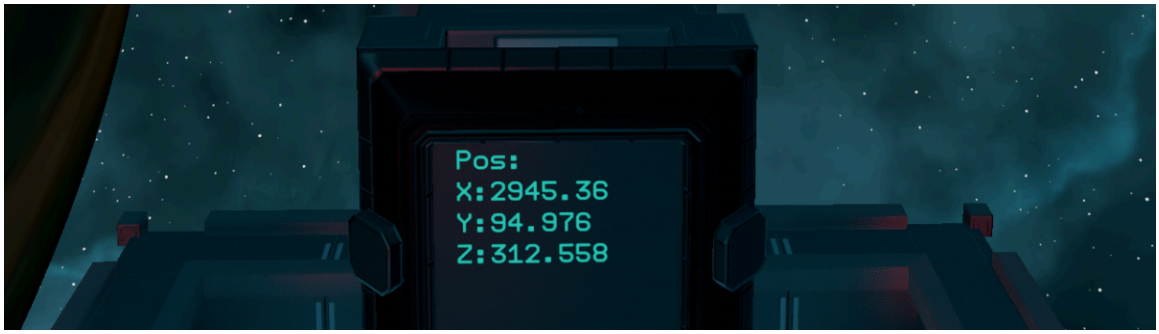
The requirements are quite particular right now, unfortunately. Here's how to guarantee your ISAN displays are read properly:

- Place the ISAN display directly facing you.
- Place it in the bottom-middle of the screen. If you set saveImages to "True" in the settings.yaml file you'll be able to see what the program sees in the form of a png saved to the same directory as the .exe. This requirement should go once we get enough data to train a machine learning model.
- Place it close to you. If you have a high resolution monitor, remember that a lot of people have lower resolution ones.

Here's an image of a Kolonio retrofit that works great on a 1080p screen:



In the image below, the program was seeing:



>> Questions / Concerns

Effect on	Answer
Does it eat **bandwidth**?	**No**. The data size is insignificant.
Does it eat **framerate**?	Test showed +2% CPU usage while active. So, **maybe** a minor effect if you're already running a low framerate.
Does it eat **memory**?	**No**. The .exe takes about 25mb.
Is it mandatory?	**No**. Only if you volunteer to join a Collective mission in which whomever in charge specifically requests usage of it.
Is this even allowed at all?	**Yes**, we have checked with JohannaFB and Frozenbyte is fine

	with it. (Starbase Developers)
How can I help improve it?	You can contribute to it on Github (it's open source) or contribute images for the machine learning model here . You can also ask IHaveNoLife#6972 about joining the project.
When xyz?	Add an issue on Github and we'll tell you :)

We hope you enjoy Starbeam!

>> CREDITS

- **IHaveNoLife** - Project lead. Created the server and client.
- **BenCo** - Created a screenshotting tool for getting images for the machine learning model, upgraded code to PEP8, main developer of the web client.
- **Strikeaglechase** - Creator of *Starmap* and its API, helped set up the API, helped design the architecture, and supplied bot tokens.
- **Lumi Virtual** - Project coordinator.
- **Polar Bear** - Created the logo.
- **Zaff** - Documentation co-writer
- **Volcano** - Provided the server and deployed to it many, many times.

Also thanks to:

+ **Recatek**, **Destravous**, **Craven**, and **Blargh** for helping with name suggestions
+ **Loumardes** and **MuNK** for pointing out issues
+ **VentusTraveler** and **Tatakai** for ship refitting.
+ **Blargh** for some useful feature suggestions
+ Developers of ISAN.