

Progetto di Ingegneria Informatica

Architetture e Sistemi di Elaborazione
GPS Data Logger and Parser

Luca Padalino – 10695959 – 935033

Tutor: proff. Mirko Reguzzoni, Lorenzo Rossi

Introduzione

Scopo del progetto è realizzare un data logger che registra, in tempo reale, uno stream binario proveniente da uno (o più ricevitori) u-blox EVK-M8T (connessi mediante porta COM), per poi analizzarlo successivamente e convertirlo nel formato standard seguendo le linee guida del protocollo UBX.

Dal file UBX viene prodotto il file RINEX nelle due componenti relative alle osservazioni e alla navigazione.

Scheda Progetto: [GPS data logger and parser | Progetto di Ingegneria Informatica \(polimi.it\)](#)

Analisi dei Requisiti

Il software consente di registrare stream di byte provenienti da uno o più ricevitori, contemporaneamente e in tempo reale.

Dagli stream acquisiti vengono campionate le **stringhe (sentences) NMEA** e i **messaggi UBX-RXM-RAWX** da cui si estraggono a loro volta i **time-stamp** necessari alla sincronizzazione dei tempi tra il GNSS e computer locale.

Al termine delle operazioni, è possibile **generare i files RINEX (Receiver INdependent EXchange format)** relativi alle osservazioni e alle navigazioni a partire dagli stream *.ubx* salvati in precedenza, mediante l'utilizzo della libreria *CONVBIN del toolkit RTKLib*.

Implementazione

Il tool è stato sviluppato in **Python 3.9**. La scelta del linguaggio scaturisce dalla volontà di realizzare un applicativo cross-platform, leggero, espandibile e facile da utilizzare.

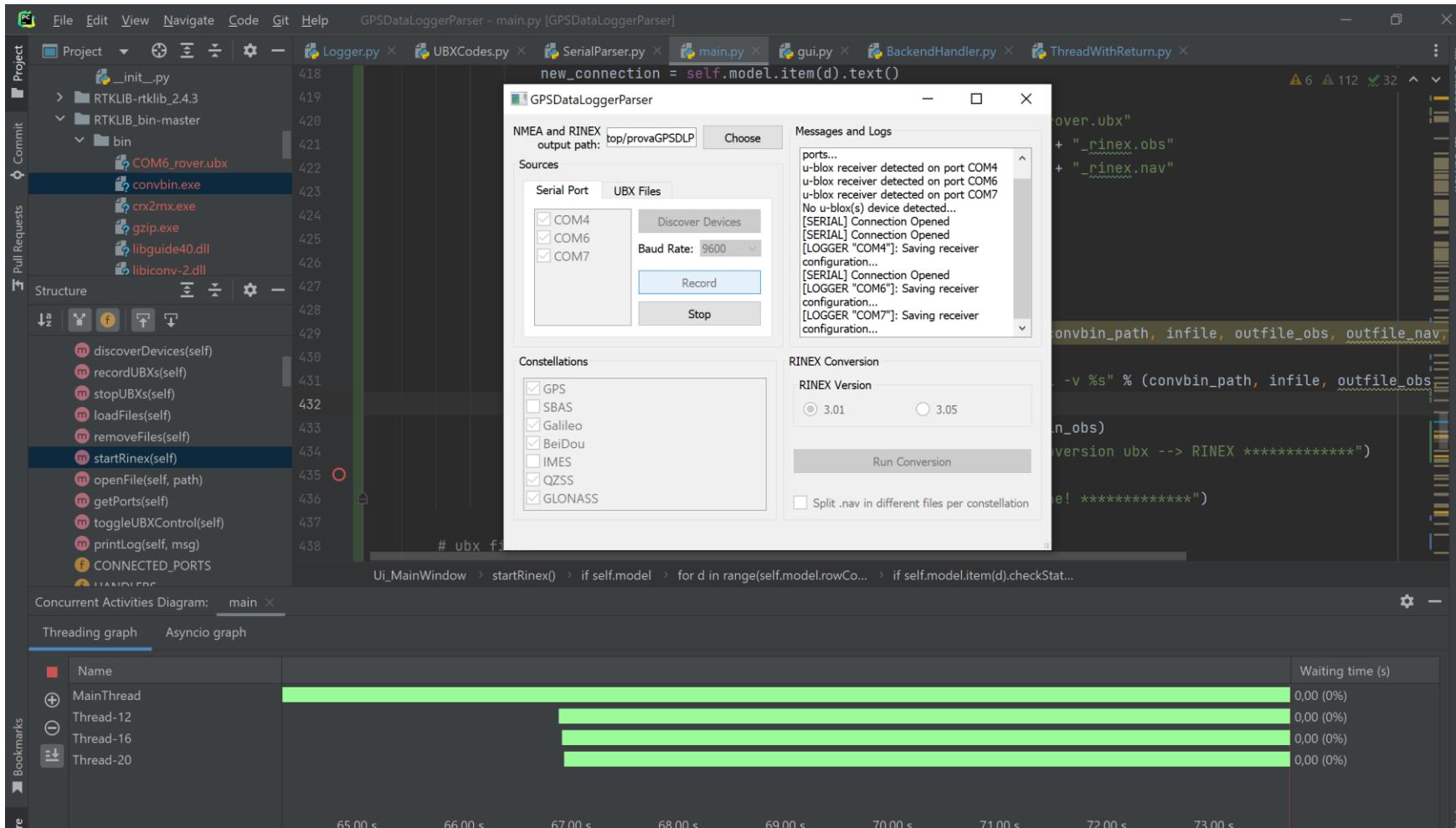
Per rendere l'esperienza utente più semplice e intuitiva possibile, è stata realizzata una **GUI** con il framework *PyQt5*.

Interfaccia Grafica

The screenshot displays the 'GPSDataLoggerParser' application window. It features a top title bar with standard window controls. The main interface is divided into several functional areas:

- NMEA and RINEX output path:** A text field containing 'ap/Desktop/prova' and a 'Choose' button.
- Sources:** A sub-panel with two tabs: 'Serial Port' (active) and 'UBX Files'. Under 'Serial Port', there are three checked checkboxes for 'COM8', 'COM9', and 'COM10'. To the right are buttons for 'Discover Devices', a 'Baud Rate' dropdown menu set to '9600', and 'Record' and 'Stop' buttons.
- Constellations:** A list of satellite systems with checkboxes: 'GPS' (checked), 'SBAS' (unchecked), 'Galileo' (checked), 'BeiDou' (unchecked), 'IMES' (unchecked), 'QZSS' (checked), and 'GLONASS' (checked).
- Messages and Logs:** A scrollable text area displaying log messages such as '[LOGGER "COM8"]: disabling ZDA', '[LOGGER "COM8"]: disabling ZDA', '[LOGGER "COM9"]: disabling DTM', '[LOGGER "COM9"]: disabling DTM', '[LOGGER "COM9"]: disabling DTM', '[LOGGER "COM10"]: disabling DTM', '[LOGGER "COM10"]: disabling DTM', '[LOGGER "COM9"]: disabling GBQ', '[LOGGER "COM8"]: disabling GBQ', '[LOGGER "COM8"]: disabling GBQ', '[LOGGER "COM9"]: disabling GLQ', '[LOGGER "COM8"]: disabling GLQ', '[LOGGER "COM8"]: disabling GLQ', and '[LOGGER "COM8"]: disabling GLQ'.
- RINEX Conversion:** A section with a 'RINEX Version' dropdown menu showing '3.01' selected and '3.05' as an option. Below it is a 'Run Conversion' button and a checkbox labeled 'Split .nav in different files per constellation'.

Ambiente di Lavoro



The screenshot shows the PyCharm IDE environment for the `GPSDataLoggerParser` project. The main window displays the code editor with a file explorer on the left and a console on the right. A dialog box titled `GPSDataLoggerParser` is open, showing configuration options for NMEA and RINEX output paths, serial ports, baud rate, and constellations. The console shows the execution of the program, including the detection of u-blox receivers on COM4, COM6, and COM7, and the saving of receiver configuration files.

Dialog Box Configuration:

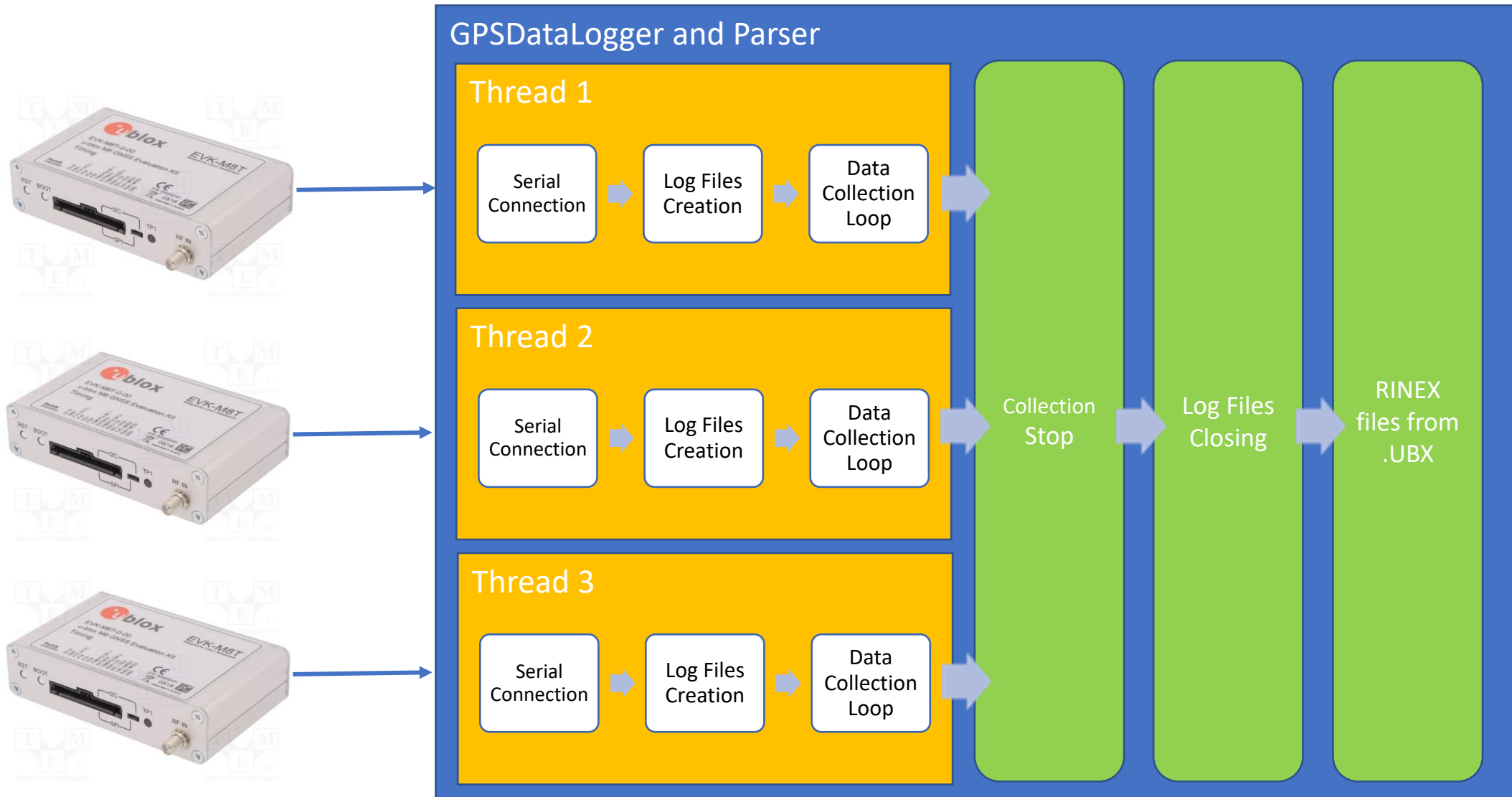
- NMEA and RINEX output path: `top/provaGPSDLP`
- Sources: ☒ COM4, ☒ COM6, ☒ COM7
- Baud Rate: `9600`
- Constellations: ☒ GPS, ☒ SBAS, ☒ Galileo, ☒ BeiDou, ☒ IMES, ☒ QZSS, ☒ GLONASS
- RINEX Conversion: ☒ 3.01, ☐ 3.05
- ☐ Split .nav in different files per constellation

Console Output:

```
ports...
u-blox receiver detected on port COM4
u-blox receiver detected on port COM6
u-blox receiver detected on port COM7
No u-blox(s) device detected...
[SERIAL] Connection Opened
[LOGGER "COM4"]: Saving receiver configuration...
[SERIAL] Connection Opened
[LOGGER "COM6"]: Saving receiver configuration...
[SERIAL] Connection Opened
[LOGGER "COM7"]: Saving receiver configuration...
```

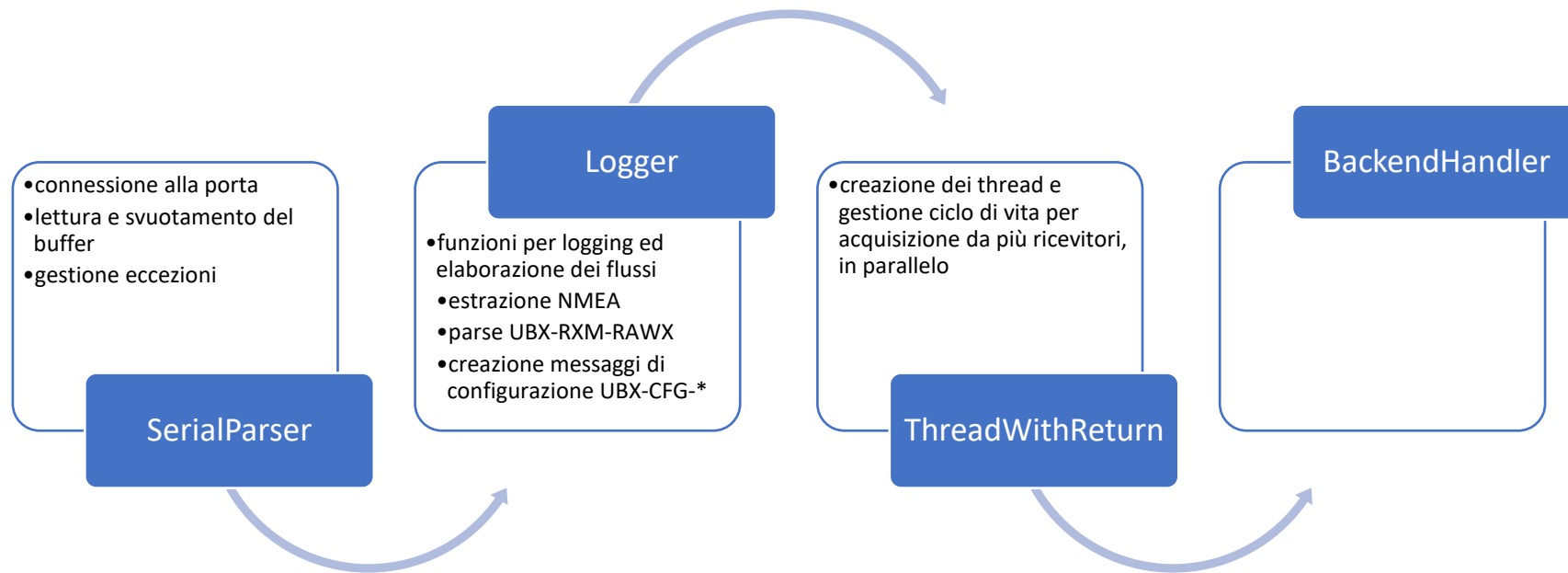


Funzionamento



Architettura BackEnd

Il software è costituito da moduli (classi di codice), ed è stato sviluppato seguendo un *approccio bottom-up*.



main.py

GUI
(PyQt5)

+

Tool conversione
RINEX

Backend
Handler

per ogni ricevitore selezionato, crea un istanza di Logger, SerialParser e
ThreadWithReturn

Logger

Contiene tutte le funzioni per la
codifica/decodifica dei messaggi
da/per il ricevitore.
logData è la funzione eseguita
dal Thread per l'ascolto dei dati
dal ricevitore

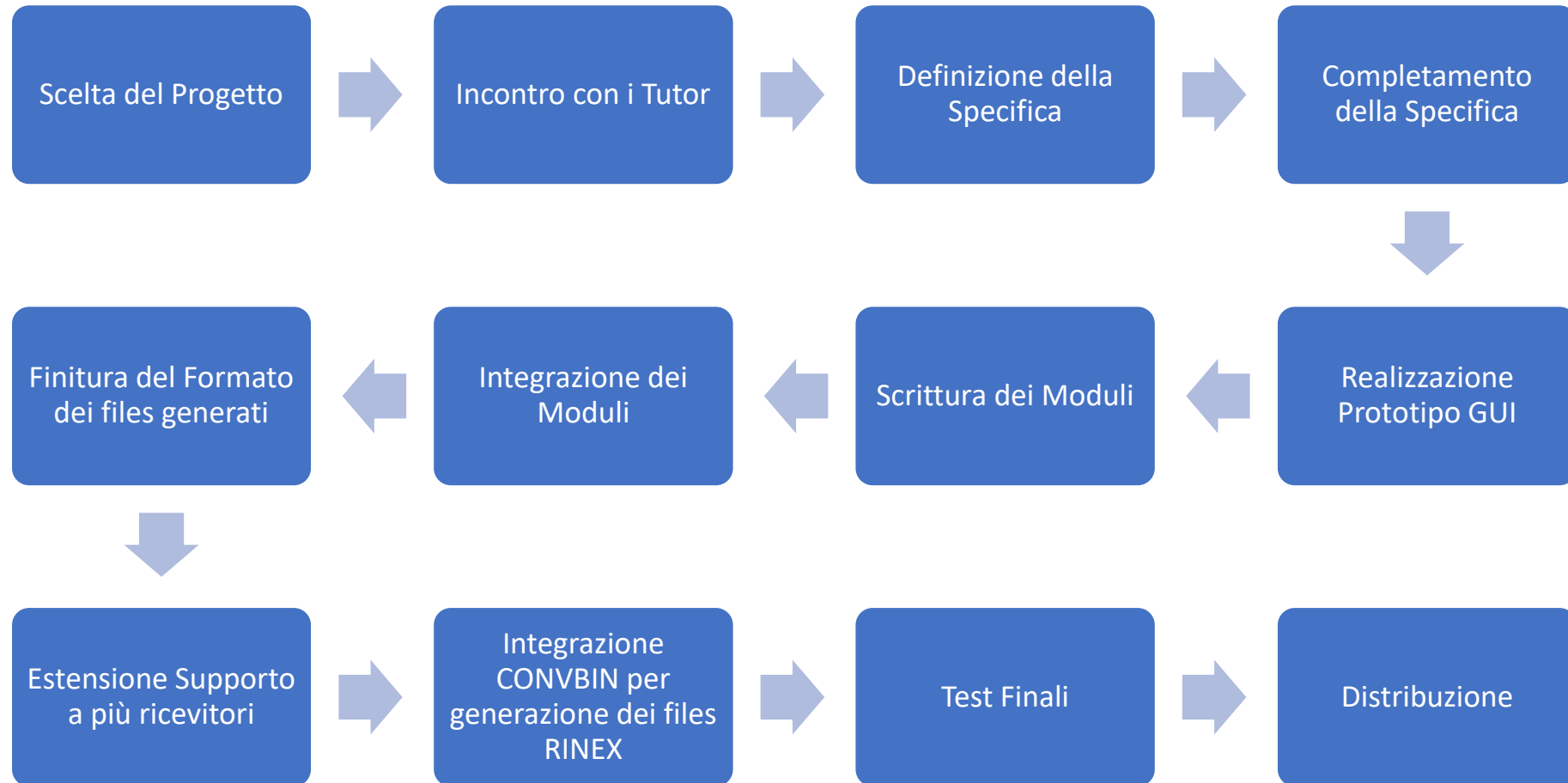
Serial Parser

gestisce la porta
seriale ed eventuali
eccezioni

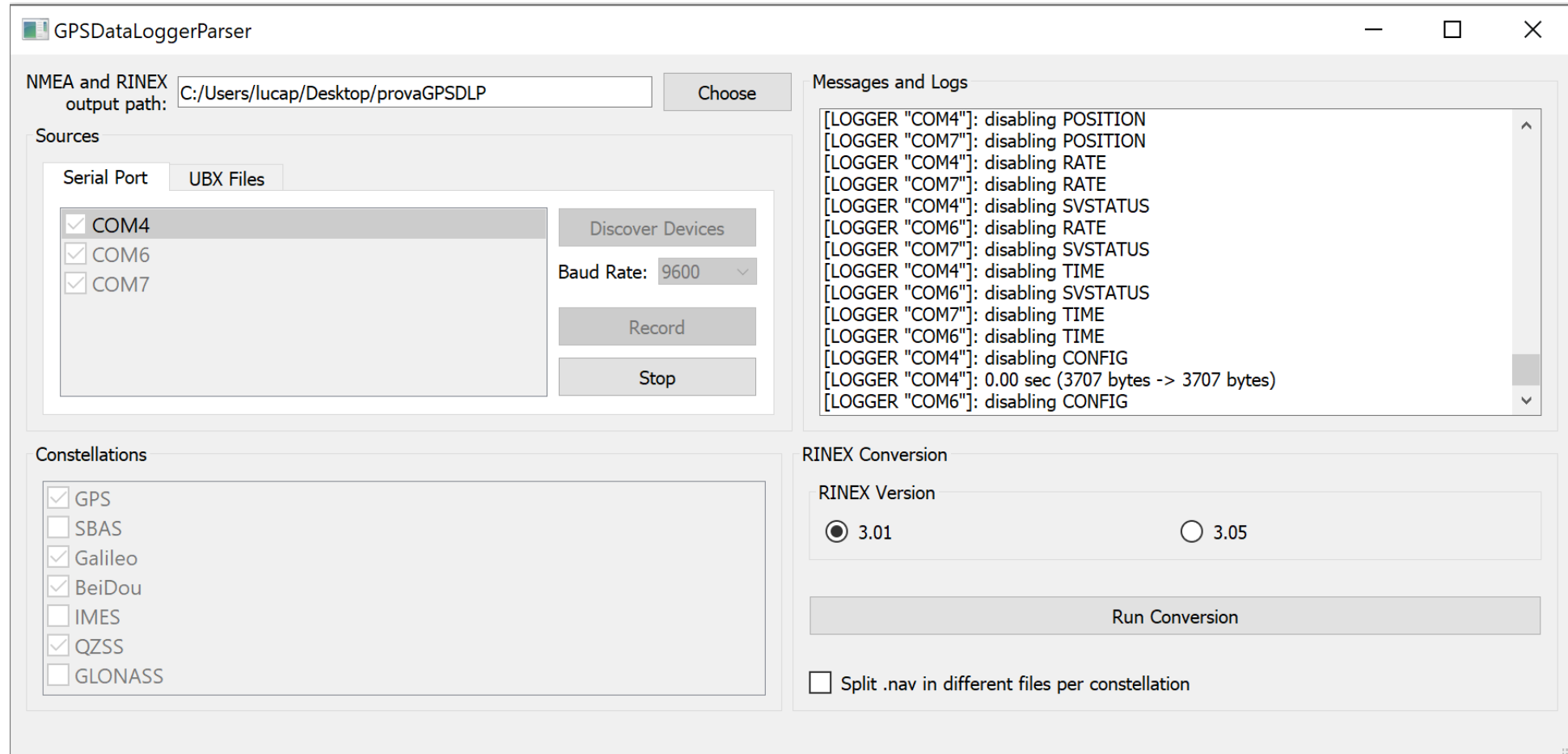
ThreadWithReturn

avvia un thread
eseguendo la
funzione logData di
Logger

Timeline



Prodotto Finale



Link Repository GitHub: <https://github.com/lucapada/GPSTDataLoggerParser>