Simulation Framework for Enhanced Train Localization

Gianluca D'Amico, Federico Nesti, Mauro Marinoni,
Giorgio Buttazzo, Gianluigi Lauro*, and Salvatore Sabina*
Scuola Superiore Sant'Anna, Pisa, Italy
**Hitachi Rail STS*

Sant'Anna
Scuola Universitaria Superiore Pisa

HITACHI
Inspire the Next

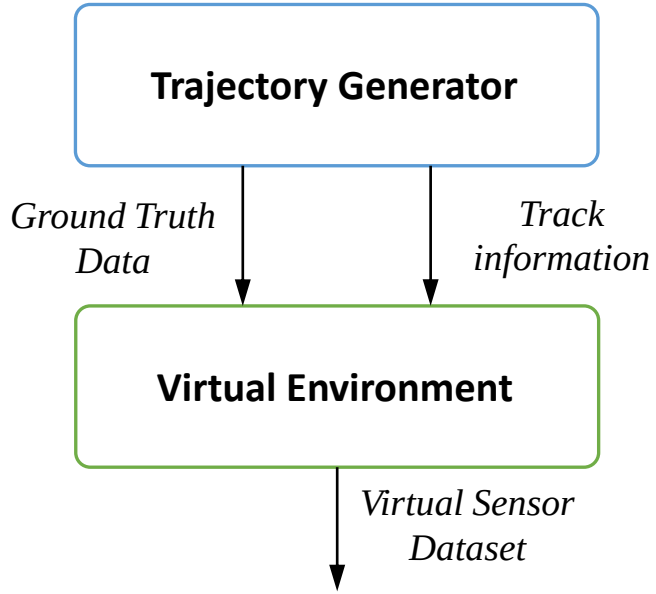
Objectives

Develop a **railway simulation framework** to test

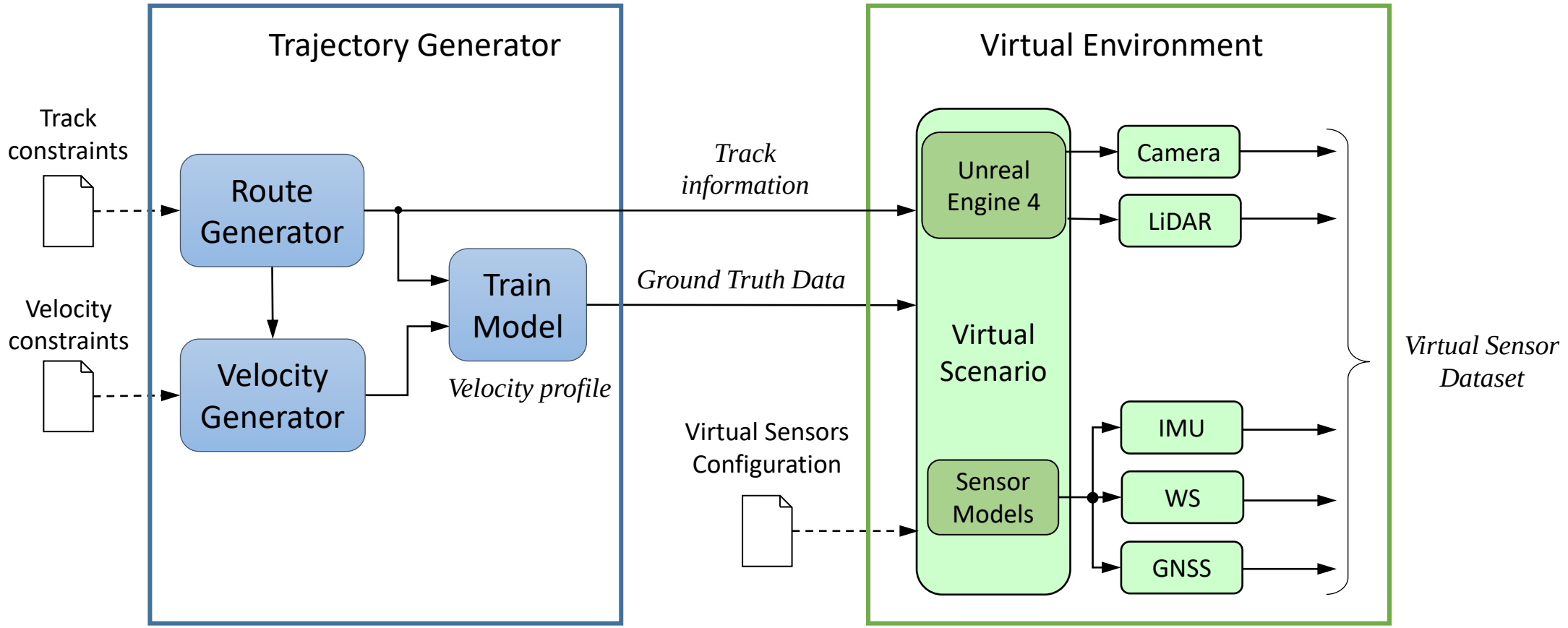
- novel localization algorithms and
- sensor-fusion methods in different scenarios.

This is done by:

1. Producing a sequence of train trajectory waypoints
2. Generating a realistic virtual scenario, following the trajectory waypoints



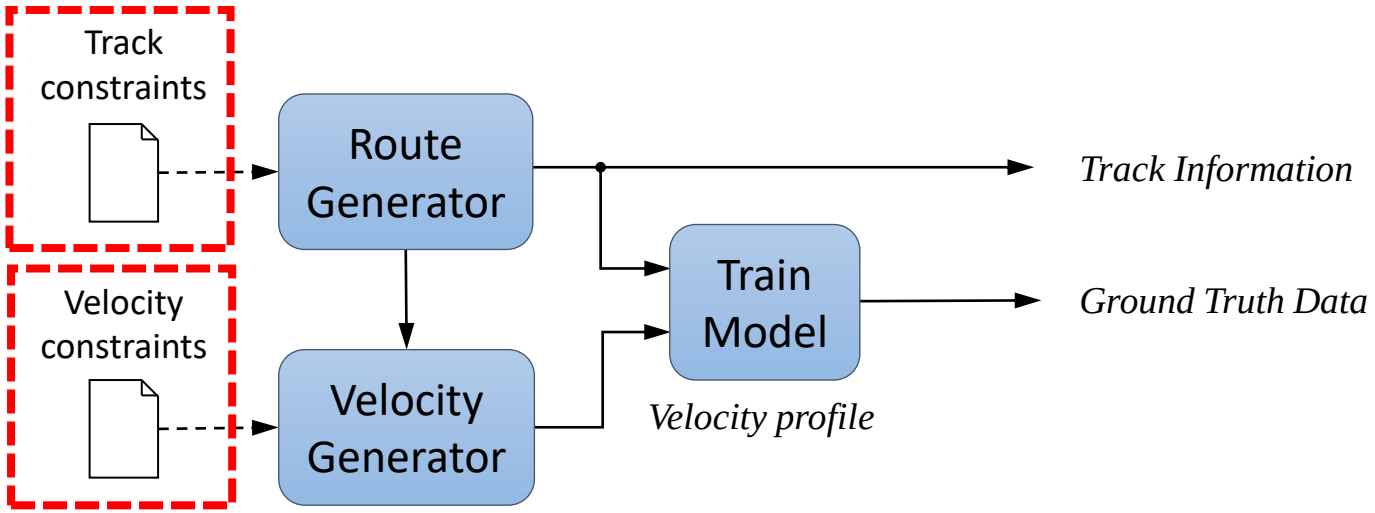
Simulator architecture



Trajectory generator

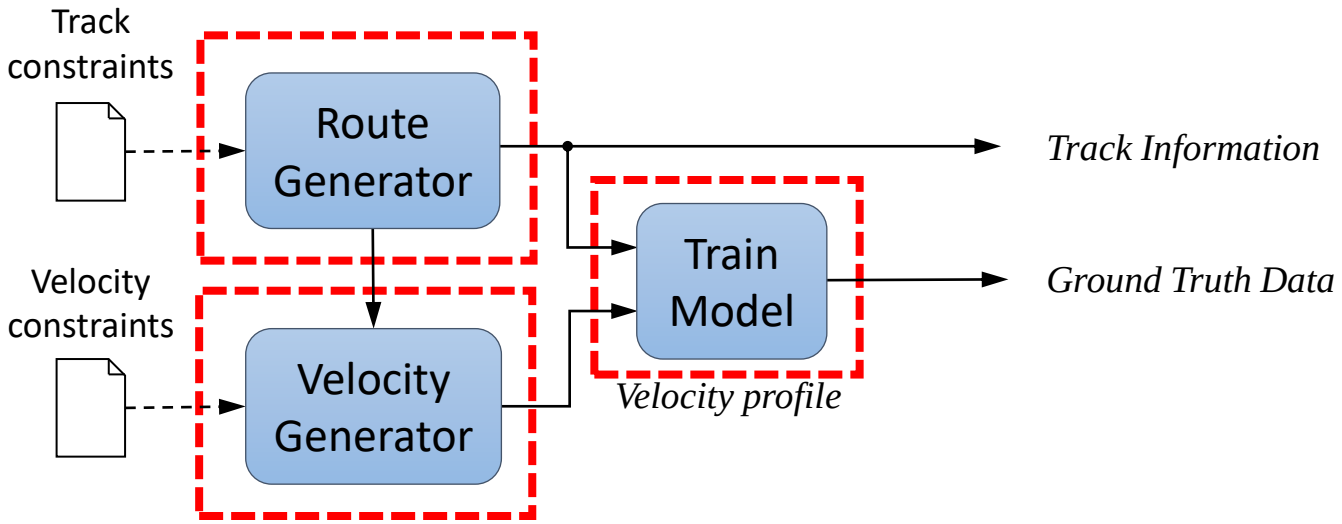
Inputs

- **Track constraints**: architectural constraints (e.g., maximum curvature, minimum strainght track between two consecutive curves).
- **Velocity constraints**: safety speed constraints of the train travelling along the railway.



Trajectory generator

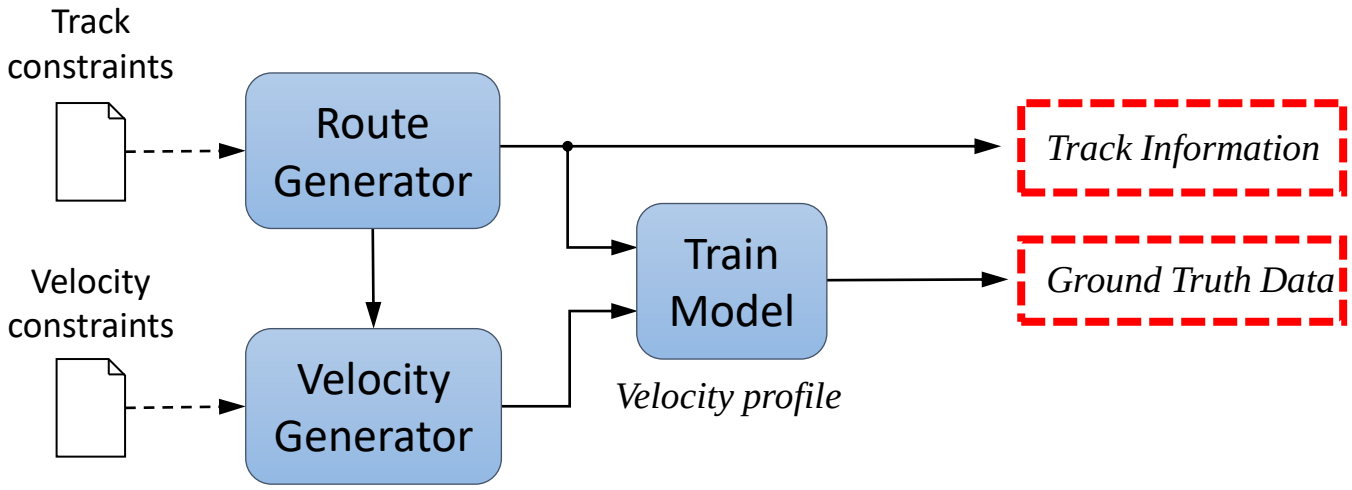
- **Route Generator:** produces a sequence of railway waypoints in 3D space, and defines the railway structure organization.
- **Velocity Generator:** produces a sequence of speed values accordingly to the railway waypoints.
- **Train Model:** produces the train time arrival at each waypoint, generating the ground truth data, based on a given train model, the railway waypoints, and the velocity profile.



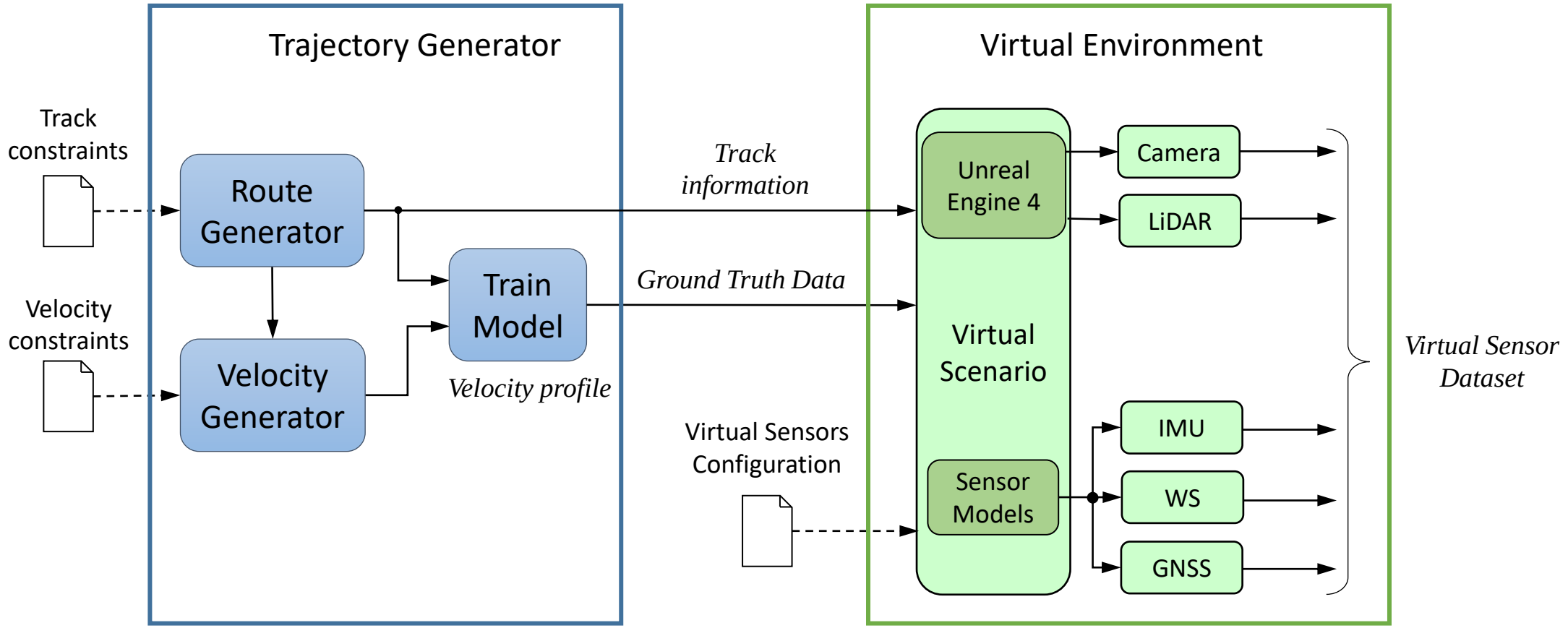
Trajectory generator

Outputs

- **Ground Truth Data** $s_{GT} = (a_i,\ v_i,\ p_i,\ \theta_i,\ t_i)$: includes the acceleration, velocity, position, and orientation of the train along with their timestamp
- **Track Information**: defines the organization of the railway track, the track is divided in chunks of different types, such as straight, curve, or station



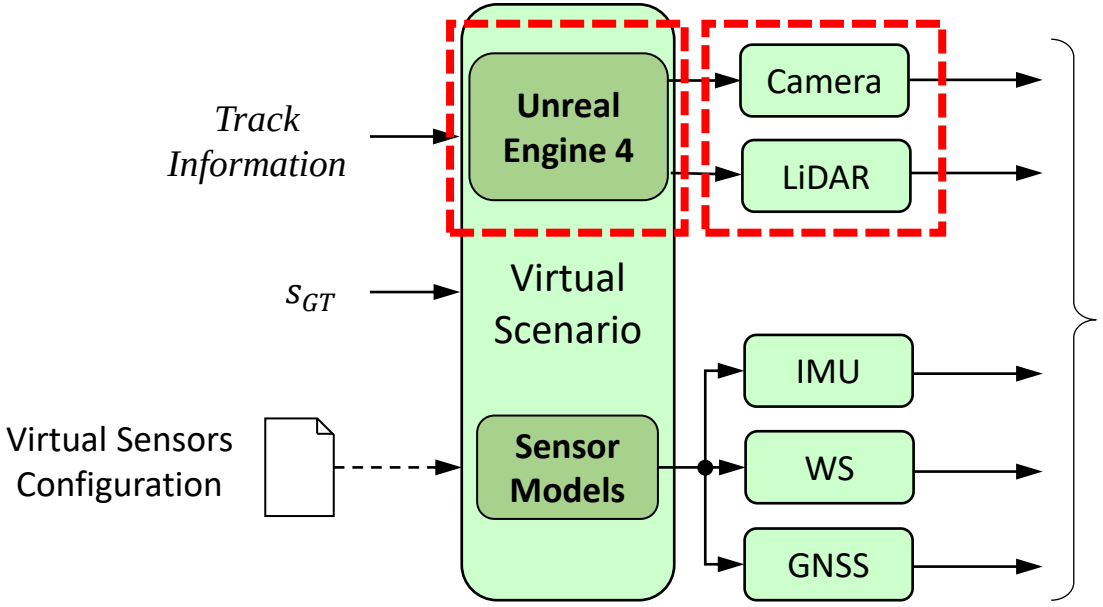
Simulator architecture



Virtual environment

Graphic Engine: using the s_{GT} and the track information,

- Generates the virtual graphic environment, including the railway infrastructures (trains, tracks, tunnels, etc.) and random environmental objects (mountains, trees, etc.).
- Moves the train on the waypoints, producing a new graphic frame at each timestamp of the s_{GT}.
- Emulates the visual sensors working principles.



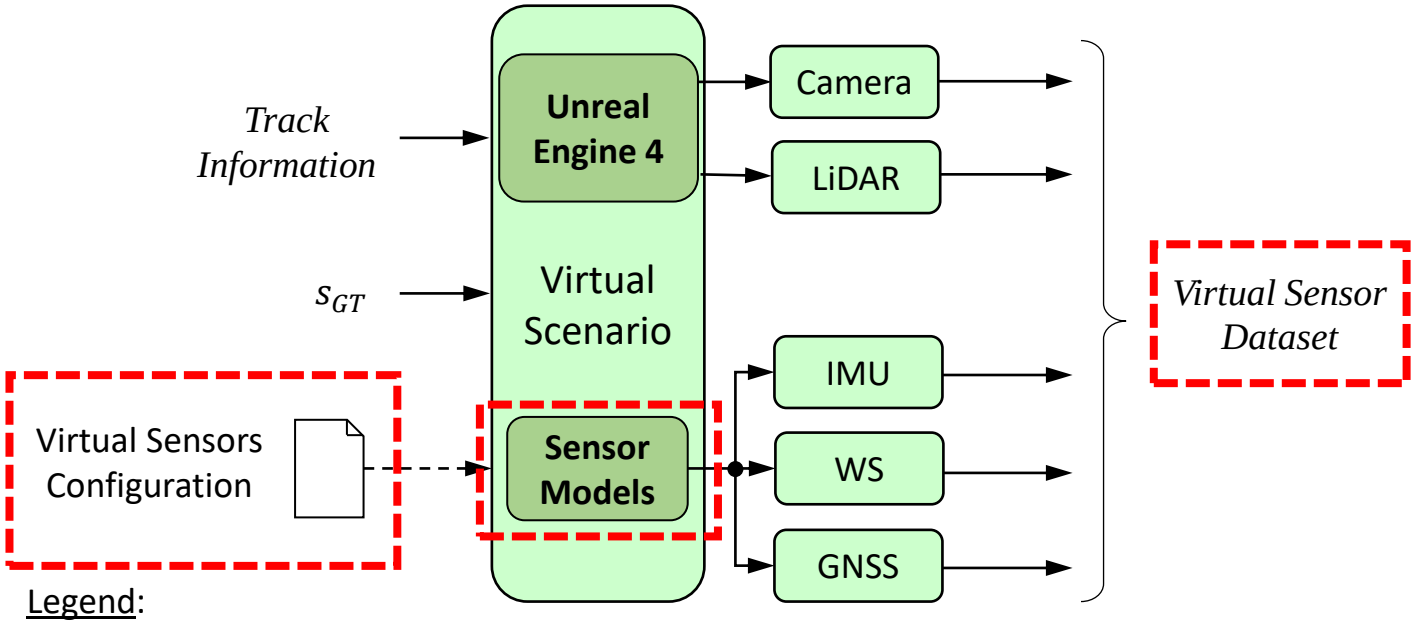
Legend:
IMU: Inertial Measumerement System; **WS**: Wheel Sensor; **GNSS**: Global Navigation Satellite System

Virtual environment

Sensor Models:

- Generates the virtual sensor measurements given the s_{GT}.
- Injects noise values in the sensor measumerements based on the virtual sensor configuration.

Output: Virtual Sensor Dataset, including all sensor measurements with correleted timestamps.

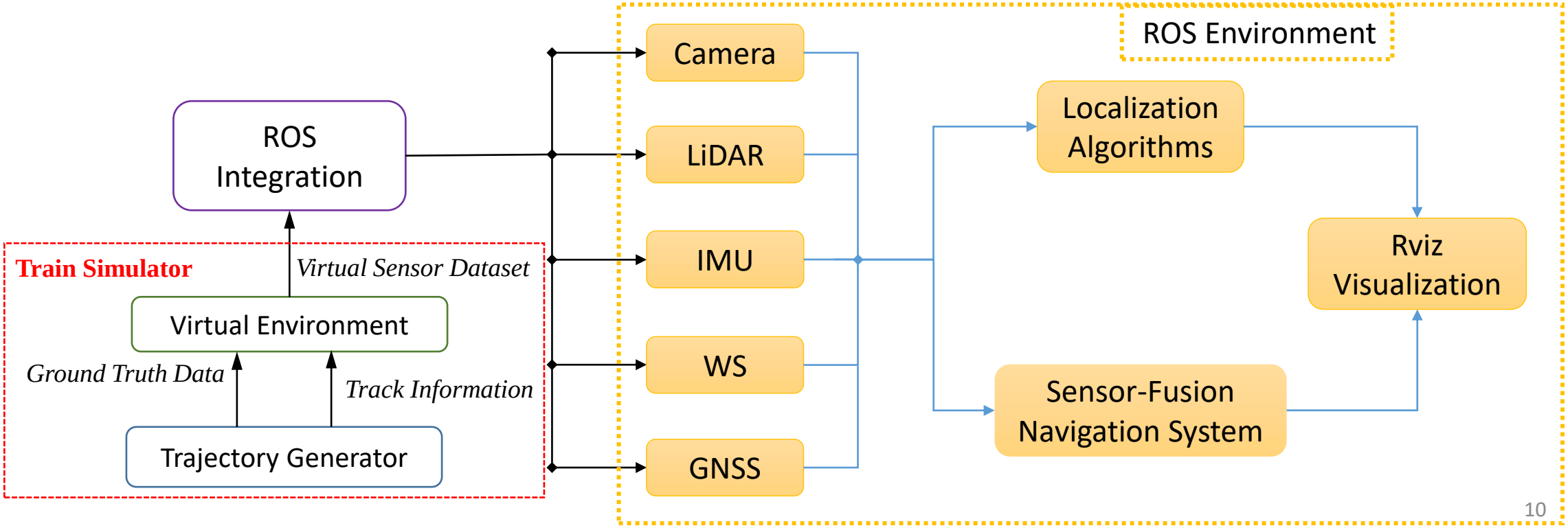


Legend:
IMU: Inertial Measumerement System; **WS**: Wheel Sensor; **GNSS**: Global Navigation Satellite System

Integration with ROS1

ROS1: Robot Operating System, it is a set of software libraries and tools to build robot applications (www.wiki.ros.org/kinetic). It allows to interconnect resources and operating nodes, through a plublisher/subscriber system.

The simulator is directly connected to ROS1 (Kinetic on Ubuntu 16) to test and visualize the results of the localization and navigation algorithms.



Conclusions

- Create a train trajectory generator that produces the ground truth data of a realistic train travel.
- Create a realistic virtual environment to produce visual sensor data, and to emulate sensor measurements dataset.

Future work

- Test novel localization algorithms on several realistic virtual scenarios.
- Use the generated sensory data for developing and testing novel sensor-fusion algorithms.
- Apply the algorithms for track discrimination and enhanced odometry.
- Compare the results obtained by simulation against those achieved on real data.

Thank you!

Gianluca D'Amico
gianluca.damico@santannapisa.it