AIPLAN 4EU

FONDAZIONE BRUNO KESSLER
FUTURE BUILT ON KNOWLEDGE

PSO
PLANNING SCHEDULING AND OPTIMIZATION

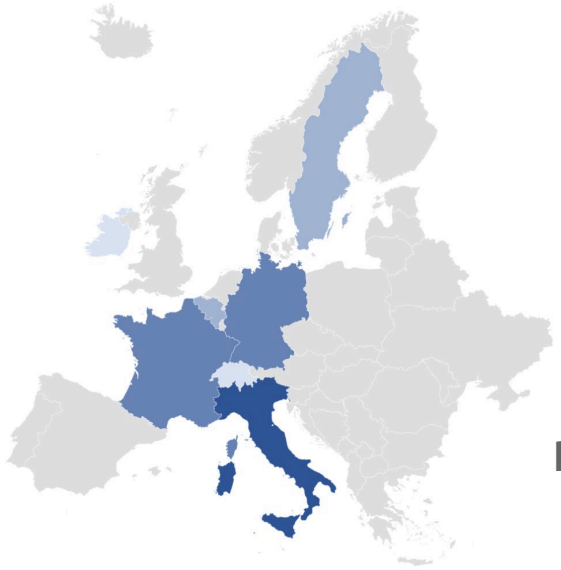
# AIPlan4EU

## Project description and results

Andrea Micheli
Head of Planning Scheduling and Optimization Research Unit
at Fondazione Bruno Kessler (amicheli@fbk.eu)

The AIPlan4EU project is funded by the European Commission - H2020 research and innovation programme under grant agreement No 101016442

# Partners



**Companies / SMEs:**

- Airbus SAS (France)
- Agrotech Valley Forum (Germany)
- F6S (Ireland)
- Magazino (Germany)
- Meritor (Sweden)
- Procter & Gamble Services (Belgium)
- Saipem (Italy)
- Trasys International (Belgium)

**Research Institutions:**

- **FBK (Italy)**
- LAAS-CNRS (France)
- DFKI (Germany)
- Örebro University (Sweden)
- Basel University (Switzerland)
- University of Brescia (Italy)
- "La Sapienza" University (Italy)

AI on Demand

AIPLAN 4EU

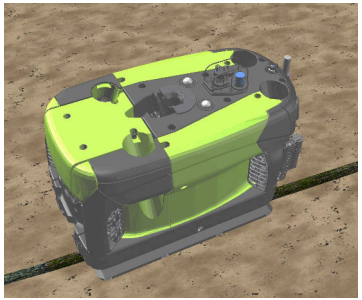
AIoD ([https://aiod.eu](https://aiod.eu)) is the European Platform for AI On-Demand

- Brings together AI stakeholders and AI resources overcoming fragmentation
- Accelerate AI-based innovations (research, products, solutions)
- One-stop-shop for anyone looking for AI knowledge, technology, services, software, and experts
- European AI market driver

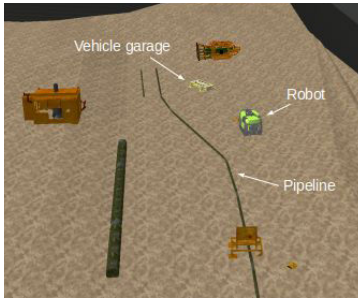
The **ICT-49 projects** had the task of enriching the platform bringing in new services

# AI Planning in a nutshell

Given a **model of a system** and a **goal to be reached under constraints**, find a **course of actions/schedule** to drive the system to the goal.



System Specification




Initial Configuration and Objectives

Planner


(Optimal) Plan/Schedule

# Applications

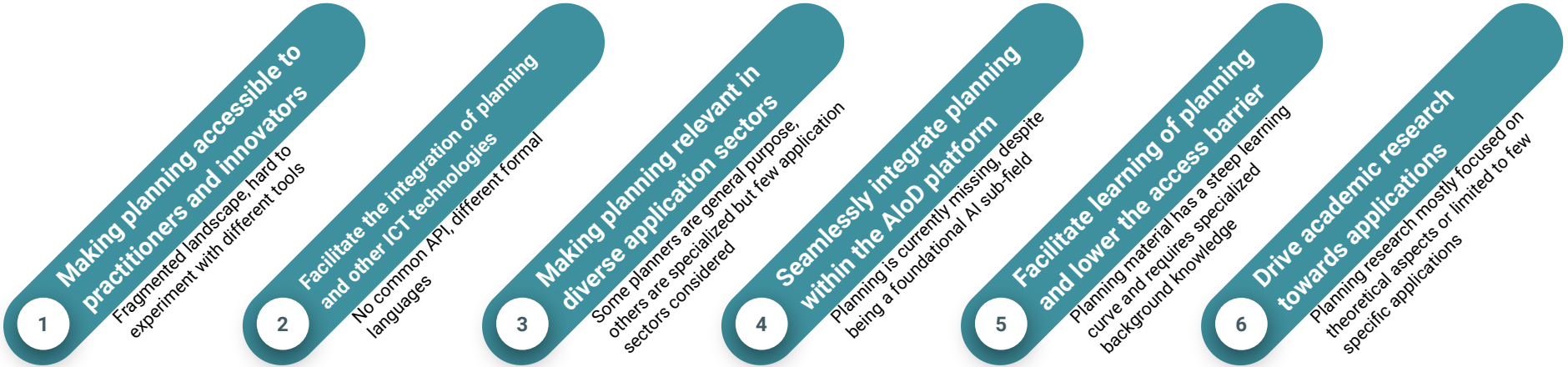
**Example applications**:

- Plan and coordinate the activities robots to achieve a desired high-level goal
  - Flexible, robust and cost-effective automation
- Plan the activities in a factory
  - Maximize production and minimize costs
- Organize the agricultural practices
  - Minimize environmental impact

Strong planning resources and capabilities are **vital needs** for the AIoD platform.

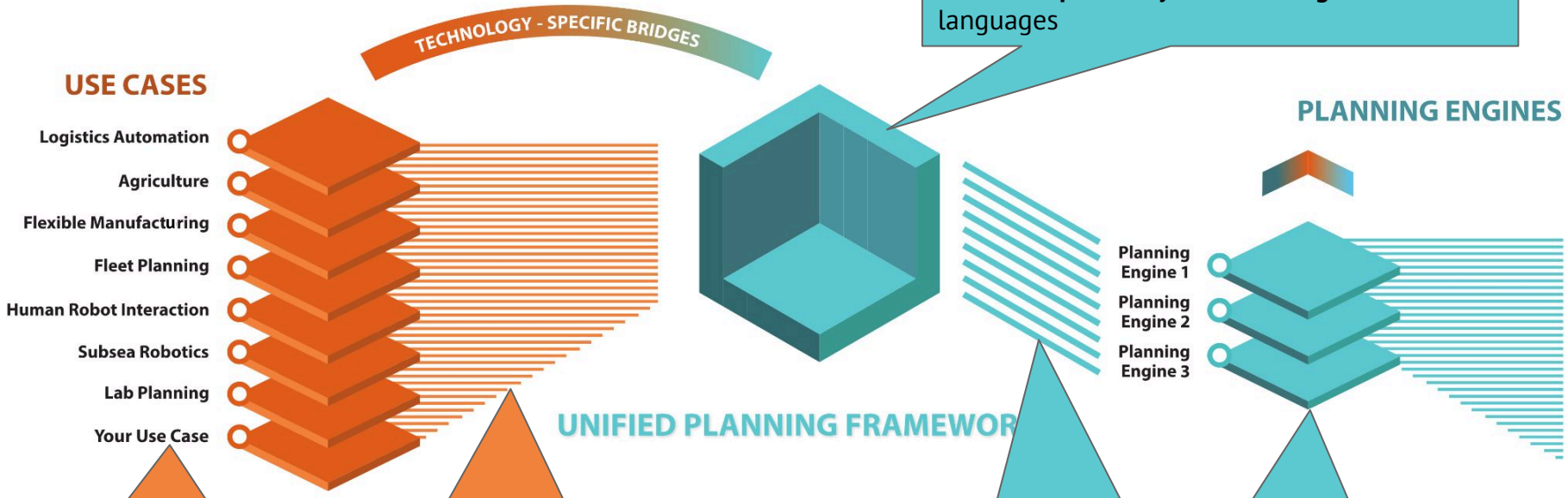
# Innovation Goals

In this Innovation Action we recognized **6 major goals**



# Vision

The UP framework is a **reusable, planner-agnostic** Python library, offering an **abstraction layer** for diverse planning engines and **interoperability with existing tools** and languages




The API offered by the UP is integrated by a **collection of reusable components** that are specific to a certain technology (not use-case specific). E.g. Integration in ROS, Integration in a WMS, ...

"Planning Engines" is a general word: plan **generators**, plan **validators**, **visualizations**...

# Results

# The Unified Planning (UP) framework

We developed a **library** to seamlessly invoke a portfolio of planning techniques

- Not only plan generation!
- https://github.com/aiplan4eu/unified-planning
- Written in python with a focus on usability and expressiveness
- Permissively open source (Apache 2.0)
- Open development, entrusted to a "**board of maintainers**"
- Planning "cookbook": https://unified-planning.readthedocs.io

```python
from unified_planning.shortcuts import *

x = Fluent("x")

a = InstantaneousAction("a")
a.add_precondition(Not(x))
a.add_effect(x, True)

problem = Problem("basic")
problem.add_fluent(x)
problem.add_action(a)
problem.set_initial_value(x, False)
problem.add_goal(x)

with OneshotPlanner(problem_kind=problem.kind) as planner:
    result = planner.solve(problem)
    if result.plan:
        print(f"{planner.name} found a plan: {result.plan}")
    else:
        print("No plan found.")
```

# Key UP Features

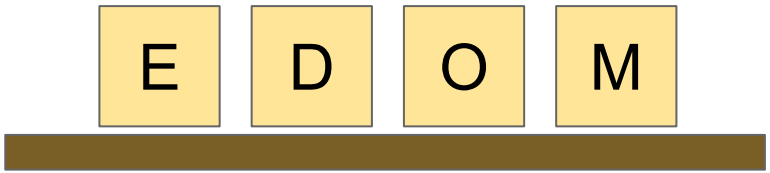
- Diverse planning problem classes supported
  - Action-based (Classical, Numeric, Temporal); (Temporal) Hierarchical Task Networks; Multi-agent; Resource scheduling; Contingent planning
- Non only plan generation: Operation Modes
  - OneshotPlanner, PlanValidator, SequentialSimulator, Compiler, AnytimePlanner, Replanner, PlanRepairer, PortfolioSelector
- Automatic “requirements”: ProblemKind
- Different types of plans supported
  - Sequential, Partial-order, Time-triggered, Simple Temporal Network, Hierarchical
- Interoperability with formal languages and other libs
  - PDDL, ANML, Tarski, GRPC
- Advanced features
  - Meta-engines, simulated effects, custom heuristics

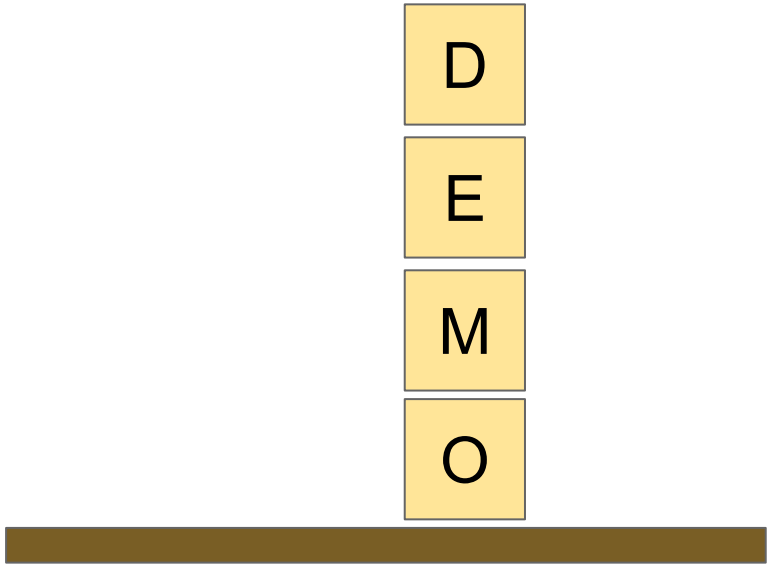

# Library Scope

- Prototype planning applications
  - Construct planning problems from data
  - Easily try multiple planners on the same problem
  - Explore multiple formulations
- Algorithms using planning as oracle
  - “Meta-planners” (more on this later)
- Combine multiple planners in a single solution
  - Ground with engine1 and solve with engine2
- Procedural modeling and solving (alpha)
  - Simulated effects
  - Custom heuristics

# See it in action!

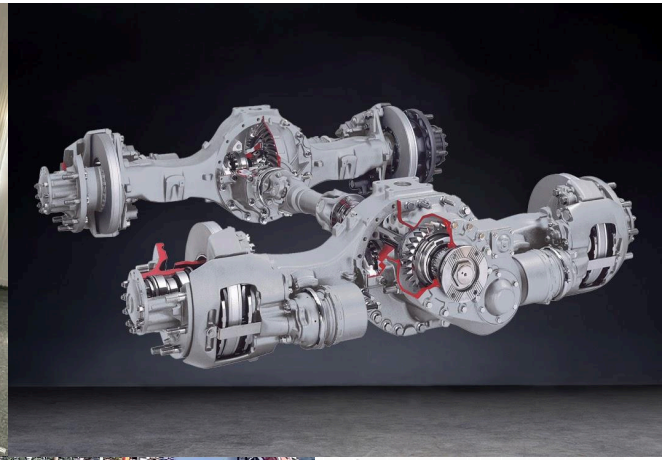
https://bit.ly/ai-cor-up

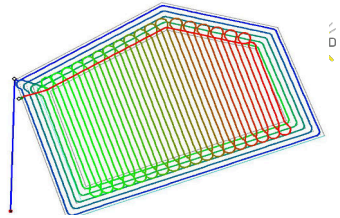


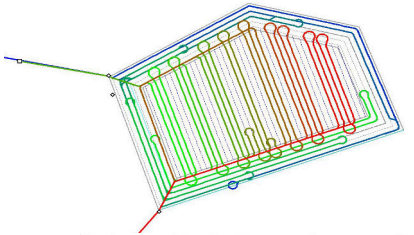



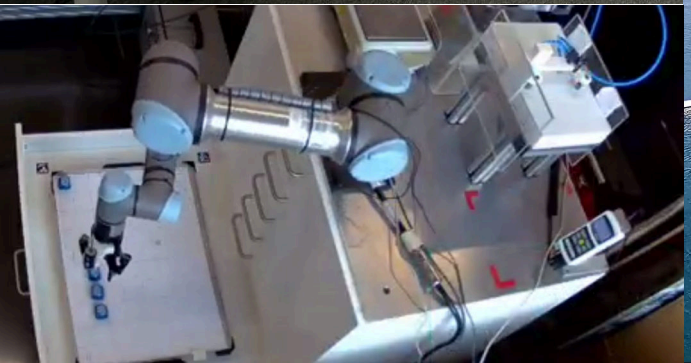
# Use-cases!

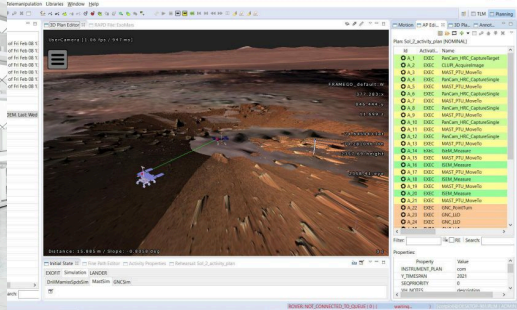
See them all on https://www.ai4europe.eu/ai-community/projects/aiplan4eu









# Reusable Software Ecosystem

- Embedded Systems Bridge (https://github.com/aiplan4eu/embedded-systems-bridge)
  - Python interface independent of middlewares/frameworks
    - Create domain based on existing Python implementation of actions and environment
    - Executable actions are prepared implicitly based on the plan provided by UP
  - Plan dispatcher based on dependency graph
  - Plan Monitoring to evaluate the state of the actions / plans

- UP4ROS & UP4ROS2 (https://github.com/aiplan4eu/UP4ROS)
  - ROS and ROS2 wrappers for UP
  - Provide single ROS node that expose the UP API via ROS messages
  - Converters for UP Python objects to and from ROS messages

- UP-Graphene (https://github.com/aiplan4eu/up-graphene-engine)
  - Easily connect to the unified-planner component in the AI4Experiments graphene platform

- And more!

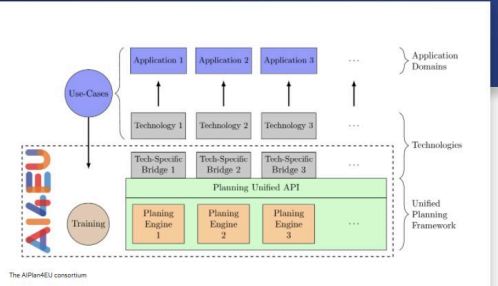
# AIPlan4EU in the AIoD

Home > Research > AI Assets > Unified Planning Framework

## Unified Planning Framework

The Unified Planning Framework (UPF) library makes it easy to formulate planning problems and to invoke automated planners.

Library | GitHub page




**Contact Details**

Andrea Micheli, Fondazione Bruno Kessler (amicheli@fbk.eu)

**License**

Apache License 2.0 (Apache-2.0)

**Main Characteristic**

Define problems in a simple, intuitive, and planner independent way

- Solve your planning problems using one of the native solvers, or by using any PDDL planner
- Dump your problems in PDDL (or ANML) format
- Parse PDDL problem formulations
- Simplification, grounding, removal of conditional effects and many other transformations are available
- and more...

**Detailed Description**

The purpose of the library is to provide an abstraction layer for planning technology allowing a user to specify planning problems in a planner independent way and then use one of the available planning engines installed on the system. The library is implemented as a Python package offering high level API to specify planning problems and to invoke planning engineers. Moreover, the library offers functionalities for transforming and simplifying planning problems and to parse problems from existing formal languages.
The library is being developed publicly under a

The UPF library is being developed by the

Home > > Planning for Logistics Automation

## Planning for Logistics Automation

Magazino is an innovative company producing robots for warehouse intra-logistics

**Developed by**

Magazino GmbH

**Business Category**

Manufacturing | Transportation

**Technical Category**

Planning and scheduling | Robotics and automation

tenorth@magazino.eu
gemignani@magazino.eu

https://www.aiplan4eu-project.eu/

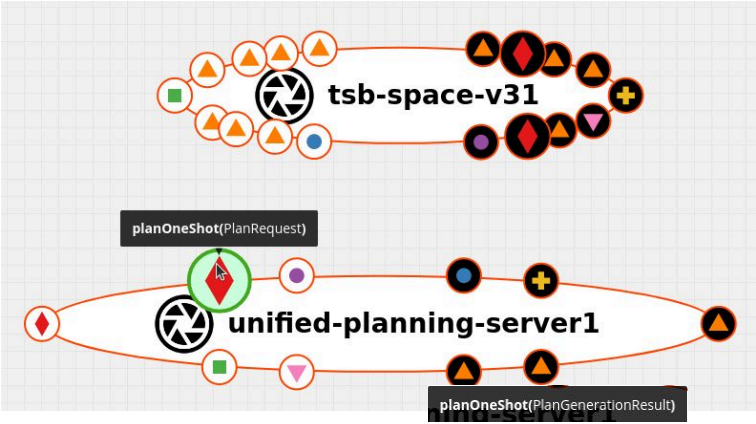
In Magazino, each robot is given a prioritized list of jobs to perform, and each of them is associated with a hand-written plan, encoded using a behavior tree formalism.

This solution has a few drawbacks: handwriting of the behavior trees requires expert knowledge and quite some brain power to be carried out without mistakes. This makes the writing tedious and the maintenance and update complex and error prone. For example, in addition to take into account the possible failures and unexpected events, such plans need to be crafted considering that the executor might get restarted at any point in time, losing and thus needing to reconstruct the world and internal states. Finally, the behavior trees are typically rather complex and they are not suitable to be analyzed or manipulated by some high-level reasoning mechanism.

In this context, planning techniques can be used to guide the user to design the complex plans required to accomplish the tasks the robot is given, and to follow their execution taking into account the automatic recovery from unforeseen circumstances, e.g., after a failure has occurred.

# AIPlan4EU in AI4Experiments

- UP-Server
  - Reusable component
  - Pre-installed planners and libraries
  - Standard GRPC interface
  - Readily usable for pipelines requiring planning!
- Demos (look for “aiplan4eu” solutions!)
  - Maze planner
  - TSB-space
  - 8 more to come!



# Open call programs

## 3 open calls for use-cases

To elicit a diversity of use-cases for automated planning technology

1500 euro to prepare the use-case and present it in a workshop with the AIPlan4EU consortium

15 use-case elicited and 9 selected to advance to call for innovators

## 2 open calls for innovators

Enrich the portfolio of planning engines (track A)
Develop new demonstrators (track B)
Address use-cases selected in the call for use-cases (track C)

Work organized in sprints, 7-months elapsed

26 projects funded
(12 track A, 9 track B and 5 track C)

# 1st call for innovators highlights

- **Engines**
  - Contingent Planning under Partial Observability
  - Planning as LTL satisfiability checking
  - SIADEX HTN Planner integration
  - Multi-Agent Coordination via Social Laws
  - Production Planning and Scheduling

- **Demonstrators**
  - Planning of cash flows
  - Integration in ROS2 for robotics
  - Intelligent Sales, Inventory and Operations Planning
  - AGV Fleet planning for intra-logistics
  - Planning for “Cloud Kitchens”
  - Planning in Field Service of Wind Turbines
  - Optimization in Nursing Homes internal activities

ROBOTICS
IN THE
WAREHOUSES

Dolomiti Robotics

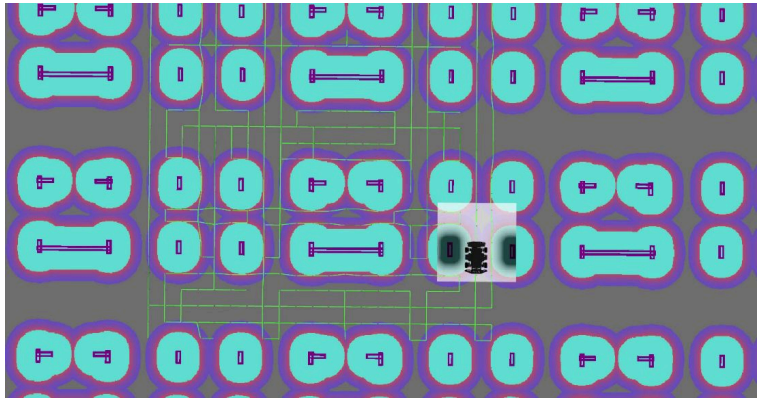
GRUPPO LOGISTICO

AIPLAN 4EU

See more details on
https://www.ai4europe.eu/ai-community/projects/aiplan4eu

# 2nd call for innovators highlights

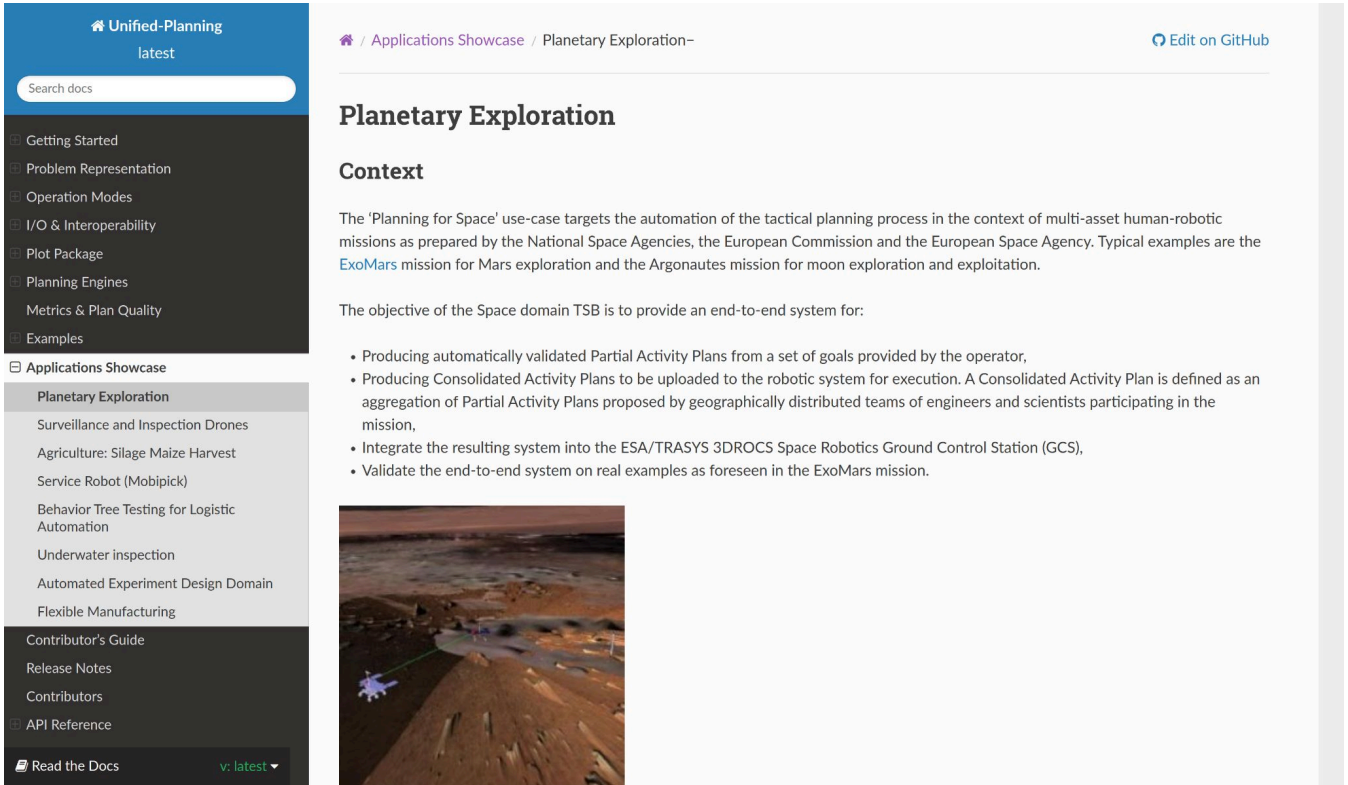
- **Engines**
  - Symbolic Search for Diverse Plans and Maximum Utility
  - PDSim: Planning Domain Simulation
  - Timelines Integration using ParaSpace
  - Integration of the NextFLAP planner
  - AIRoutePlan: vehicle routing support
  - A meta-planning engine for automatic parameter configuration
  - Integration of a Generalized Planner in the Unified Planning Framework
- **Demonstrators**
  - Hospitality employees' automated planning
  - Predictive Maintenance Driven Task Planning System
  - AI Planning for Integrated Urban Traffic Control
  - AIPlan4Grid: Plan and balance the electrical grid operations
  - Responding with AI-planning to Disasters
  - Just-In-Time Bunkering
  - Coverage Path Planning for Autonomous Ground Robots



See more details on
https://www.ai4europe.eu/ai-community/projects/aiplan4eu

# Planning Cookbook

https://unified-planning.readthedocs.io



# Conclusions

# Conclusions

- We developed a reusable and engine-agnostic python library for automated planning
  - Open source
  - Open governance
  - Ecosystem of reusable “TSB” components
- Project results integrated in the European AI on-Demand Platform
  - Both descriptions of the project activities and interactive demos
- Demonstrators for 7 internal use-cases + 14 open-calls
- Planning cookbook



# Thanks for your attention!

aiplan4eu-project.eu

**Andrea Micheli (amicheli@fbk.eu)**