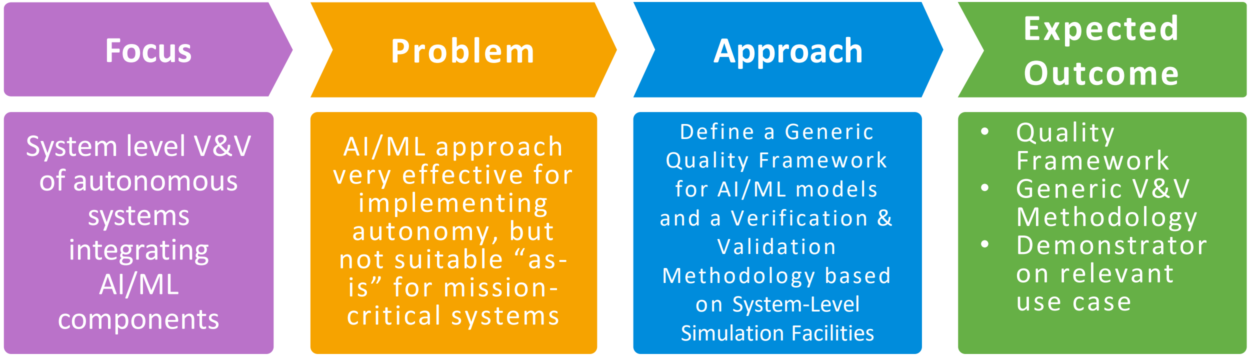
Verification and Validation of Autonomous Systems

S. Fratini, P. Fleith, N. Policella (SOLENIX)
A. Griggio, S. Tonetta, S. Goyal, T. Hoa, J. Kimblad, C. Tian (FBK)
K. Kapellos, C. Tranoris (TRA)

ASTRA 2023 – ESA/ESTEC

Study Overview

"to propose and demonstrate a **generic Verification and Validation methodology** based on the usage of the **System-level Simulation** Facilities, specifically targeted at **autonomous systems** using **AI-models**"



Focus	Problem	Approach	Expected Outcome
System level V&V of autonomous systems integrating AI/ML components	AI/ML approach very effective for implementing autonomy, but not suitable "as-is" for mission-critical systems	Define a Generic Quality Framework for AI/ML models and a Verification & Validation Methodology based on System-Level Simulation Facilities	• Quality Framework • Generic V&V Methodology • Demonstrator on relevant use case

Consortium

Consortium

SOLENIX

Solenix Engineering GmbH (DE)

Prime Contractor

FONDAZIONE
BRUNO KESSLER

Fondazione Bruno Kessler

Subcontractor

TRASYS
INTERNATIONAL
BUSINESS UNIT OF NRB
DARING TO COMMIT

Trasys International

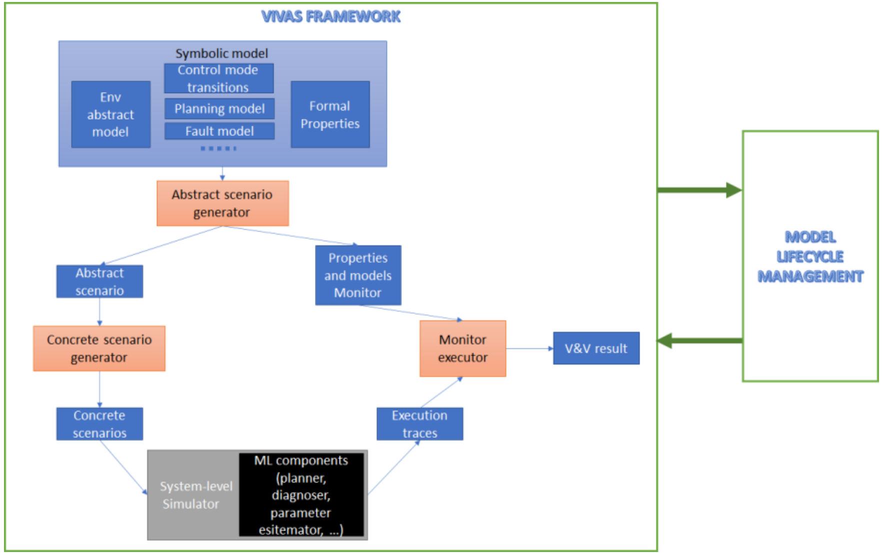
Subcontractor

Plan

- VIVAS Framework
- Proof of concept use case
- VIVAS Architecture components / instantiation for the POC
- VIVAS Framework and MLOps
- Deployment
- Conclusions and recommendations

The VIVAS Framework

VIVAS Framework



- Formal symbolic model of the autonomous system and of the properties to be tested

I. Abstract scenario & monitors generation

II. Concrete scenarios generation

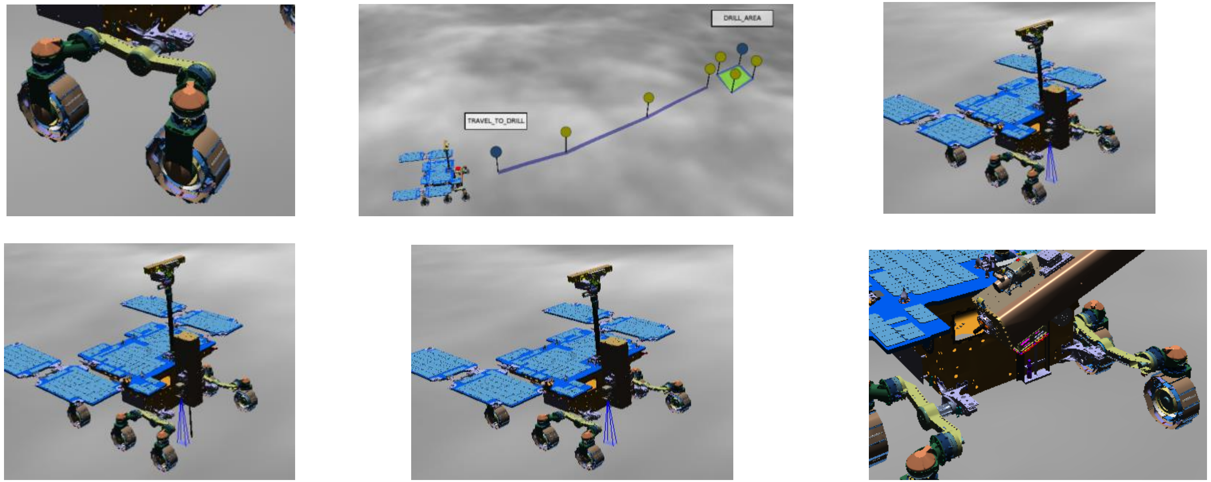
III. System level simulation of the concrete scenarios and execution traces generation

IV. Monitoring of the execution traces and generation of V&V results

- The analysis of the results confirms the validity of the AI/ML models or improvement is required

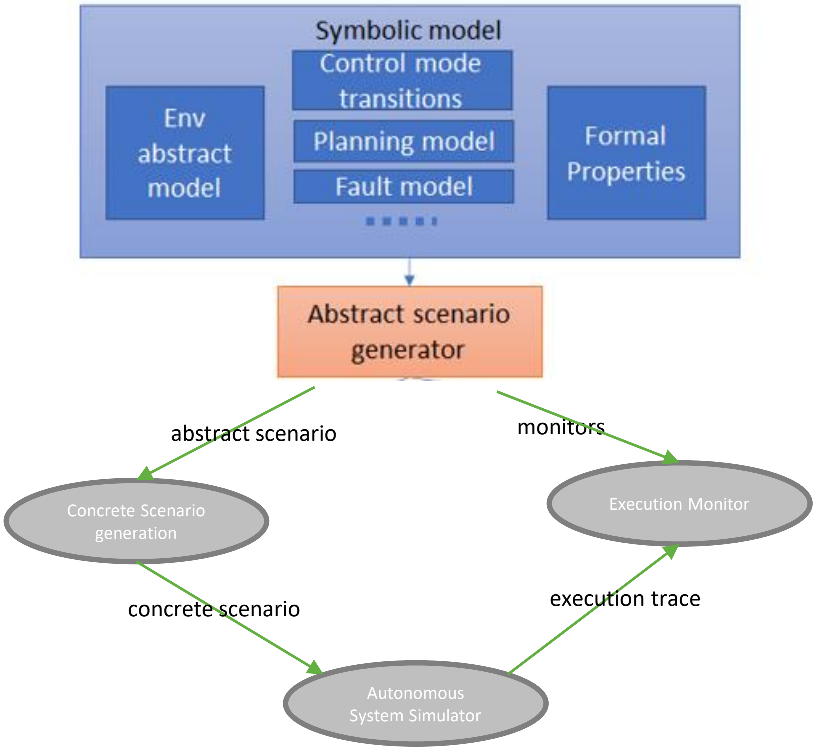
Proof Of Concept

- ExoMars mission: subsurface sampling



- Extended to introduce AI components and ML models
  - **On-board activity planner** (AIPlan4EU / ROBDT)
  - **ML model estimating the ‘warm-up’ duration** of external mechanisms (ROBDT)
  - **ML model for novelty detection** from images (opportunistic science – developed in the VIVAS activity)

VIVAS Framework – Abstract Scenario Generation

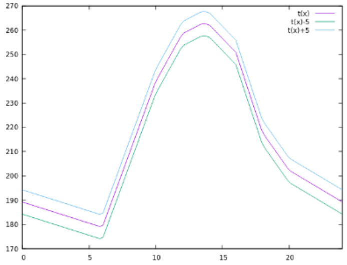


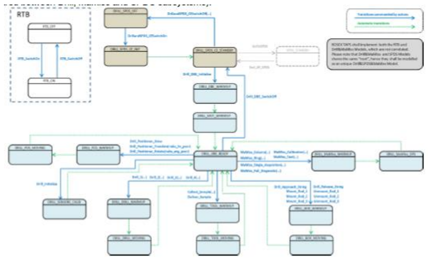
- Formal symbolic model of the system
  - Environment model
  - System under test (including the AI/ML parts)
- Written as a symbolic transition system in the SMV language of the nuXmv model checker

- The Formal Properties represent the formalization of the system-level requirements of interest, specified as linear temporal logic (LTL) formulas
- The nuXmv model checker computes the execution trace witnessing the realization of the abstract scenario of interest

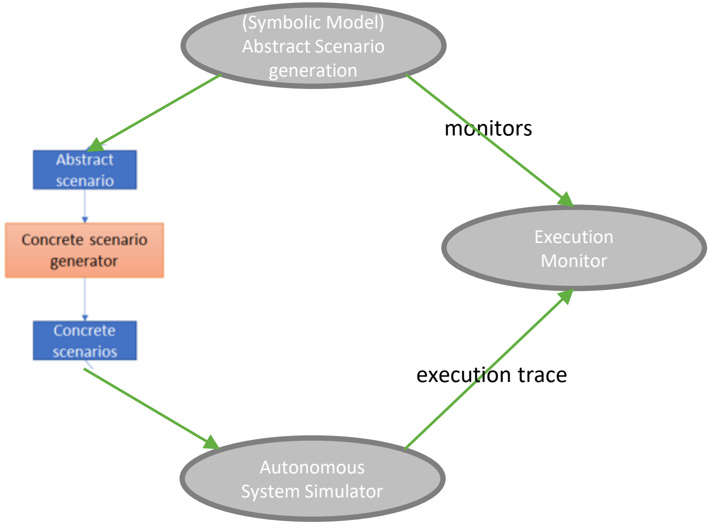
VIVAS Framework – Abstract Scenario Generation

- POC Use Case
  - **Environment abstract model**: type of terrain, position of the rover in a grid, season and time in the day, areas of interest, temperatures, fluxes (*environment.smv*)
  - **Autonomous System model**
    - **Planning model**: models the autonomous system at the logical level of mission activities (*planner.smv*)
    - **Model for estimating resources** consumption (*estimator.smv*)
  - **Formal properties** (*scenario.smv*)
    - Goals achieved in the available resources
    - The objects of interest (novelty) are detected
    - Decisions to analyze the objects of interest do not preclude the achievement of the mission goals





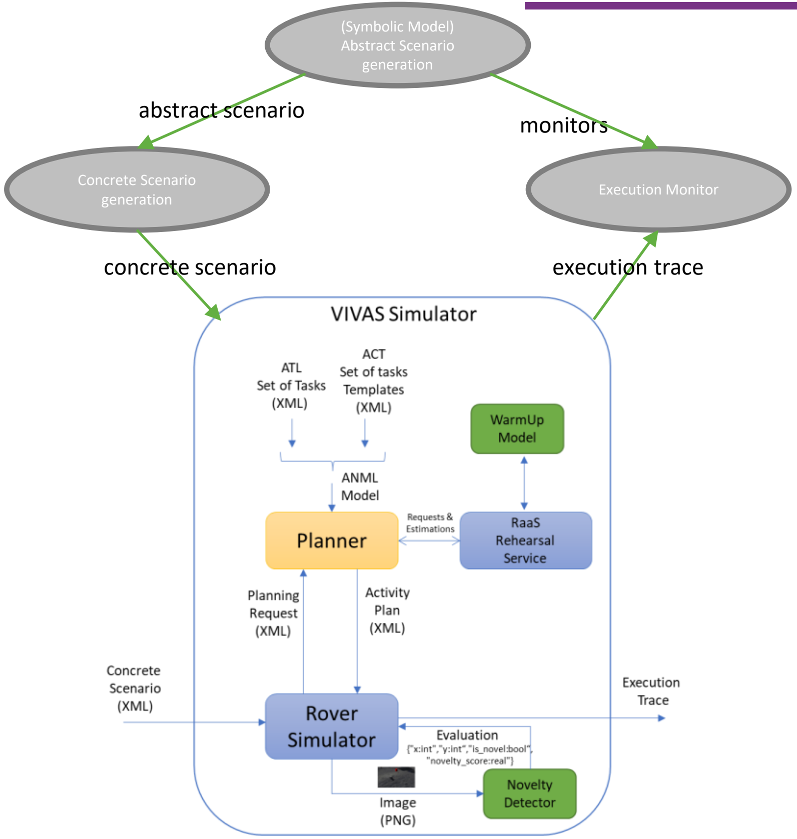
VIVAS Architectural Design – Concrete Scenario Generation


(Symbolic Model) Abstract Scenario generation
monitors
Abstract scenario
Concrete scenario generator
Concrete scenarios
Execution Monitor
execution trace
Autonomous System Simulator

- Each trace produced by the model checker (abstract scenario) is refined into a set of concrete scenarios
- Mapping from abstract variables to parameter ranges and probability distributions
- Ensures adequate level of coverage at abstract level (e.g., coverage of the abstract scenarios wrt. the set of properties, coverage of the properties wrt. the model)

- POC Use Case:
  - Goals request
  - Initial state of the environment (e.g., positions of the objects of interest) and the system (state of the different subsystems)
  - Evolution of temperatures and fluxes

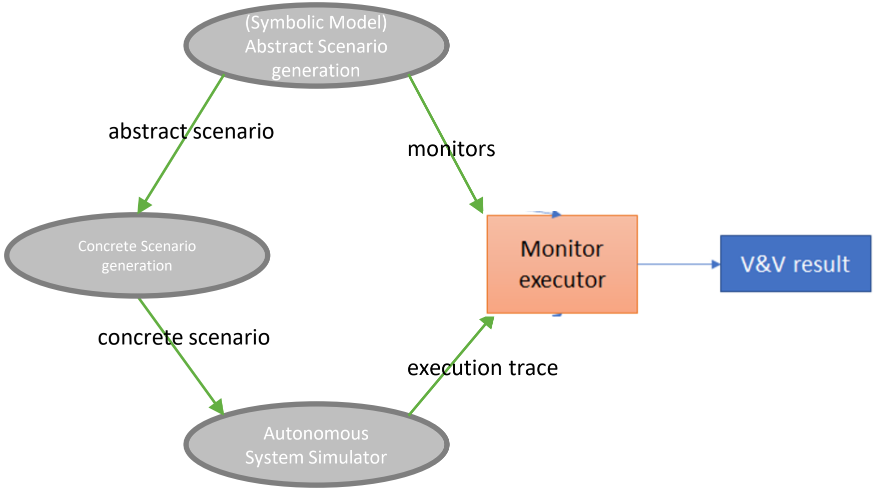
VIVAS Architectural Design – Simulator



- The Autonomous System simulator includes the AI/ML components under test
- Simulates the concrete scenario under the requested conditions
- Generates the execution trace

- POC Use Case: Extension of the 3DROV simulator including the AI/ML models and o/b planner
  - Concrete scenario:
    - Goals
    - Initial System state
    - Environmental information: location of the objects of interest, temperatures, fluxes, …
  - Execution trace
    - State of all subsystems
    - Executed Activities
    - Reports

VIVAS Architectural Design – Execution monitor



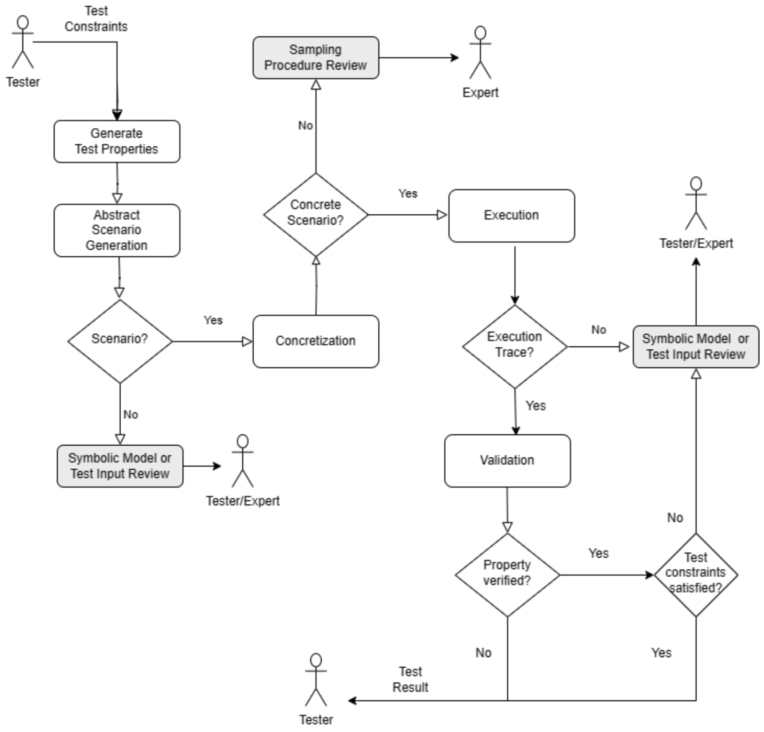
(Symbolic Model) Abstract Scenario generation
abstract scenario
monitors
Concrete Scenario generation
Monitor executor
V&V result
concrete scenario
execution trace
Autonomous System Simulator

- Monitor generator
  - Synthesis of the monitor from the system-level property of interest and the abstract system model
- Monitor executor
  - Analysis of the traces produced by the simulator to check the properties and determine the overall coverage
  - Results saved in a database
- POC use case:
  - The requested Goals are achieved within the available resources
  - The novel objects are detected
  - The simulated trace complies with the abstract scenario

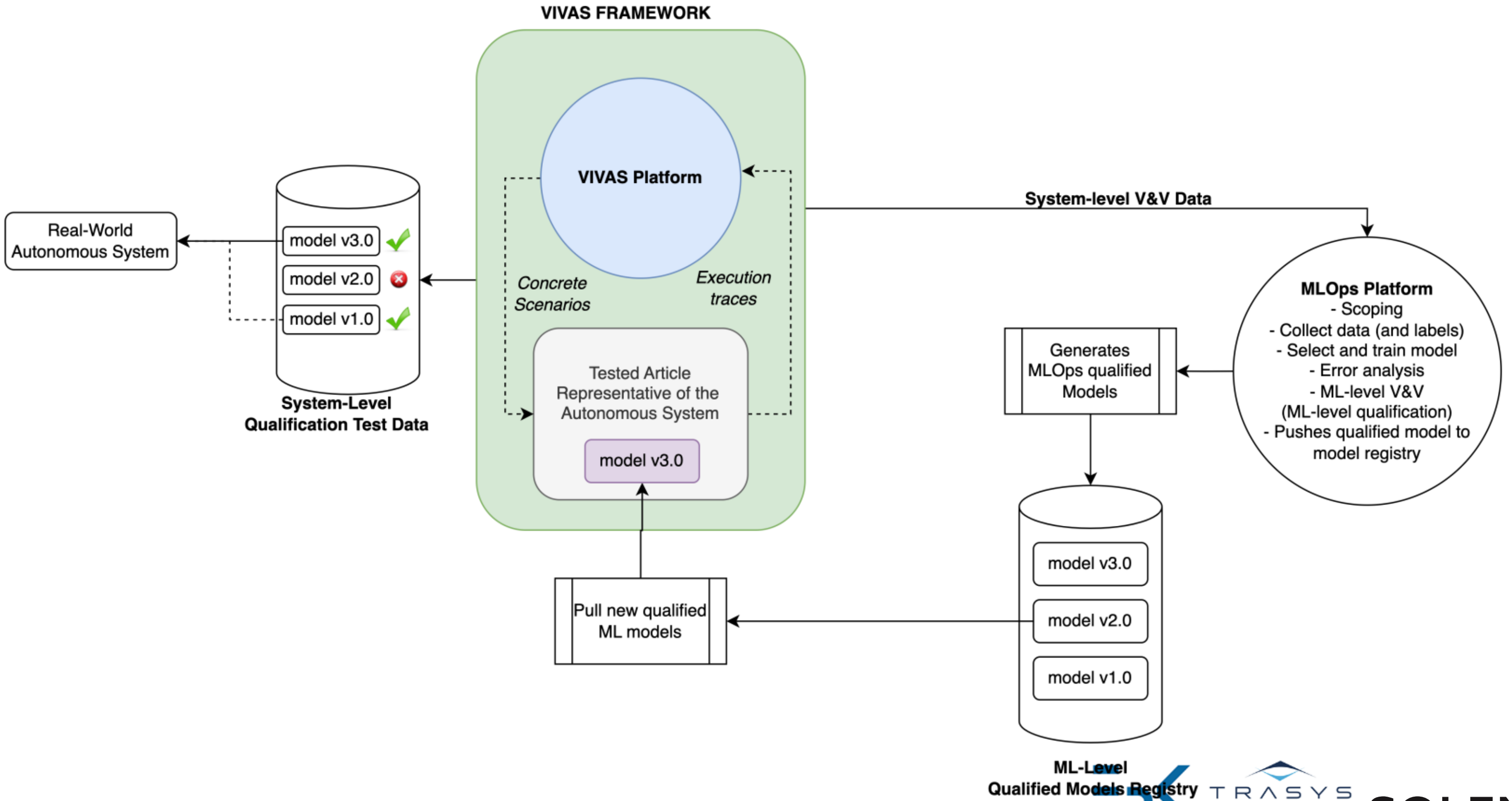
Test Results

- For each test case, 4 possible outcomes:
  - **Property satisfied:**
    - Scenario constraints satisfied: **OK**, test works as expected
    - Scenario constraints violated: **WARNING**, problem with coverage
  - **Property violated:**
    - Scenario constraints satisfied: **KO**, found problematic scenario
    - Scenario constraints violated: **WARNING**, found unexpected problematic scenario

VIVAS Framework – User Interaction Summary

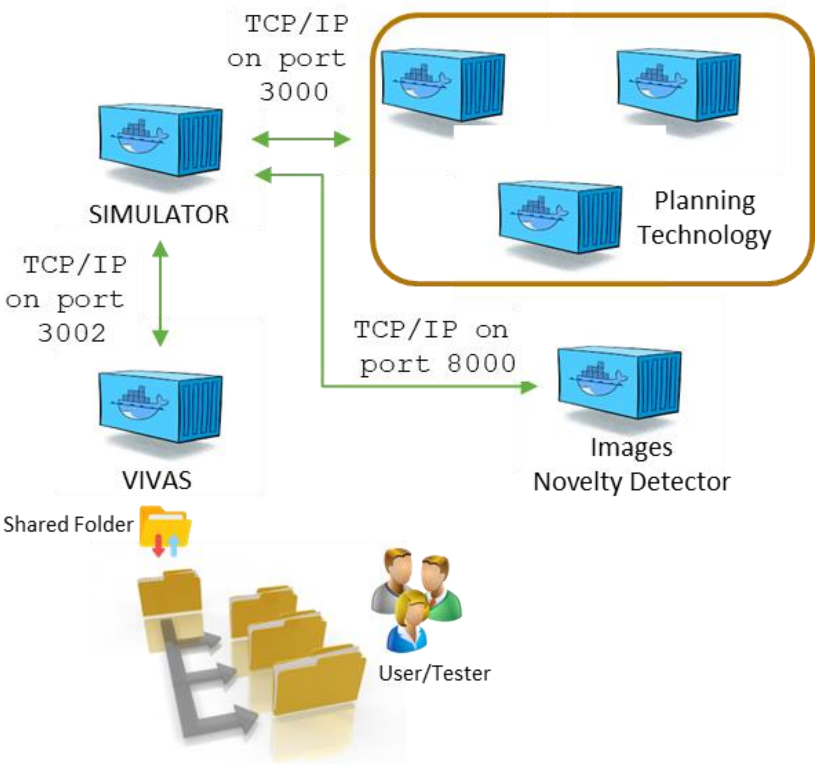


VIVAS Framework Interface with Model Development Life-Cycle



VIVAS FRAMEWORK
VIVAS Platform
Concrete Scenarios
Execution traces
Tested Article Representative of the Autonomous System
model v3.0
Real-World Autonomous System
model v3.0 ✓
model v2.0 ✗
model v1.0 ✓
System-Level Qualification Test Data
System-level V&V Data
MLOps Platform
- Scoping
- Collect data (and labels)
- Select and train model
- Error analysis
- ML-level V&V
(ML-level qualification)
- Pushes qualified model to model registry
Generates MLOps qualified Models
model v3.0
model v2.0
model v1.0
ML-Level Qualified Models Registry
Pull new qualified ML models

Deployment



- VIVAS is deployed in 6 docker containers:
  - VIVAS-Framework
  - Planning Technology:
    - WIA Planner
    - RaaS Service
    - DHS Model
  - 3DROV Simulator
  - Novelty Detection Model

- Shared folder for I/O
- Python script to launch tests

Conclusions

Achievements

- Demonstrated the **feasibility of a model-based approach to system-level validation and verification** of autonomous systems integrating AI/ML components

- Implemented the **VIVAS Framework**, a general, domain independent **support for qualification of AI/ML in operation**, leveraging the rigorousness provided by the model checking approach

- The test cases have been deployed using 3DROV and ROBDT, ESA technologies for autonomous planetary robotic assets

- For information, contact **Quirien.Wijnands@esa.int**

Recommendations

- Consider a follow-up:
  - To analyze the results of VIVAS runs providing more detailed insights on tests results
  - To integrate VIVAS in an MLOps loops for ML models qualification

- Investigate the customization of VIVAS in various scenarios:
  - Robotic assets
  - Autonomous Platforms for EO

- Consider the possible integration of VIVAS as a building block for future deployments like:
  - The DT (digital twin) infrastructure
  - ExoMars ground rover control infrastructure

Thank You

Solenix Engineering GmbH
Spreestrasse 3
64295 Darmstadt
Germany

info@solenix.de

www.solenix.de