# Development and Deployment of ML Systems

The paper talks about a new way to build and use machine learning systems called MLTRL. It's like a step-by-step guide inspired by how they make spacecraft systems, focusing on making sure ML systems are strong, reliable, and ethical. MLTRL helps move ML projects from research to real-world use, with examples showing how it works in different areas. It's all about making machine learning better and more responsible.

To explain the research paper in detail, We need to go through it section by section, breaking down the concepts and findings in simpler terms.

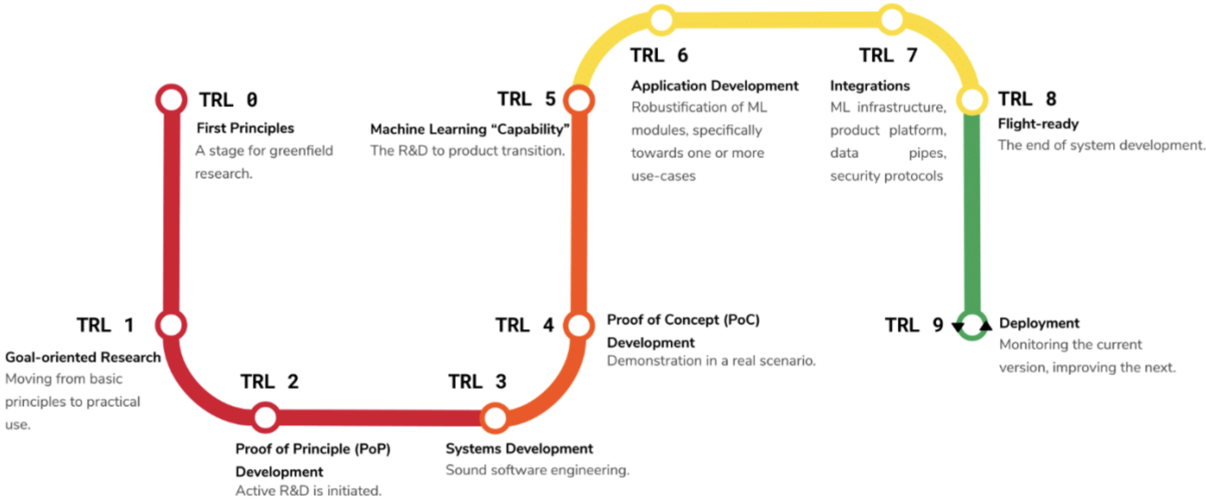
1 **Introduction:** The paper starts by discussing the importance of machine learning (ML) and the challenges in developing ML systems that are robust, reliable, and ethical. It introduces the concept of Machine Learning Technology Readiness Levels (MLTRL) as a framework designed to address these challenges, drawing inspiration from the Technology Readiness Levels used in other engineering fields.

2 **Background:** This section provides context on traditional software engineering practices and how they differ from the development of ML systems. It highlights the unique challenges of ML, such as data dependency and model uncertainty, and discusses the need for a new framework to better guide the development of ML technologies.

3 **The MLTRL Framework:** Here, the paper details the MLTRL framework, explaining each level of readiness from initial concept through to deployment and continuous improvement. It emphasizes the importance of rigorous testing, validation, and ethical considerations at each stage.

4 **Case Studies:** The document presents several case studies to illustrate how the MLTRL framework can be applied in real-world scenarios. These examples demonstrate the framework's versatility and effectiveness in guiding the development of ML systems across different domains.

The infographic above illustrates the Machine Learning Technology Readiness Levels (MLTRL) framework in a step-by-step process. It visually represents the journey from the initial conceptual stage, through development and testing phases, to the deployment and continuous improvement of an ML system. Each stage is marked with specific symbols or icons to denote key activities such as idea generation, data collection, model training, testing and validation, deployment, and monitoring for feedback. This infographic is designed to provide a clear and educational overview of the MLTRL framework, with a color scheme that distinguishes each level for easy understanding.




**Level 0: First Principles**

**Description:** Identification of fundamental principles and theoretical research relevant to the problem.

A research team identifies that principles of reinforcement learning could be applied to develop an autonomous driving system. They explore fundamental theories and algorithms that enable a vehicle to learn from interactions with its environment.

**Level 1: Goal-Oriented Research**

**Description:** Focused research aiming to address specific goals identified in Level O.

The team conducts research on how to use reinforcement learning for specific goals in autonomous driving, such as obstacle avoidance and route optimization, focusing on algorithms that can learn from simulated driving scenarios.

**Level 2: Proof of Principle (POP) Development**

**Description:** Development of a theoretical model or prototype that demonstrates the feasibility of the concept.

A simple simulation is created where a basic reinforcement learning model successfully navigates a virtual car around obstacles in a controlled environment, demonstrating the principle's viability.

**Level 3: System Development**

**Description:** Developing a more complete system that incorporates the machine learning model into a broader framework.

Development of a more sophisticated system that integrates the reinforcement learning model with sensors and actuators in a prototype vehicle, allowing for basic autonomous navigation in a lab

**Level 4: Proof of Concept (POC) Development**

**Description:** Development and testing of a prototype in a lab environment to demonstrate its functionality and potential for real-world application.

The prototype vehicle is tested in a closed environment (e.g., a test track), where it demonstrates the ability to navigate complex scenarios like intersections, stop signs, and pedestrian crossings.

**Level 5: Machine Learning "Capability"**

**Description:** The machine learning system achieves a level of capability where it can perform designated tasks effectively in a controlled environment.

The system is now capable of handling real-world driving scenarios in a controlled environment, showing reliable performance in navigating, parking, and traffic understanding.

**Level 6: Application Development**

**Description:** Development of a full-scale application that utilizes the machine learning system within a specific context or domain.

The autonomous driving system is developed into a full application, designed for integration with consumer vehicles, including user interfaces, connectivity features, and safety

**Level 7: Integration**

**Description:** Integration of the machine learning system into its target environment, ensuring compatibility with existing systems and workflows.

The autonomous driving application is integrated into a commercial vehicle model, with extensive testing to ensure compatibility with the vehicle's hardware, software, and user expectations.

**Level 8: Flight Ready**

**Description:** The system is fully tested, validated, and ready to be deployed in its operational environment.

After final testing and validation, including on public roads under various conditions, the system is certified as safe for public use and ready for mass production.

**Level 9: Deployment**

**Description:** The machine learning system is deployed and operational, being used in its intended real-world environment.

The autonomous vehicles are now deployed and in use by consumers, with ongoing monitoring to collect data on performance, user feedback for improvements, and updates deployed to address emerging challenges or enhance functionality.