Bridging AI with Multi-Domain Capabilities

Technical Whitepaper: Uli SDK

Executive Overview

The **Unified Link Interface (Uli) SDK** is a state-of-the-art software framework designed to develop adaptable functional modules for secure, safe, and reliable robotic systems. By providing a robust, pre-validated solution for the complex aspects of robotics development, the Uli SDK empowers developers to focus on unique, value-adding logic, significantly reducing project risk and accelerating time-to-market.

---

1. Core Architectural Principles

Dynamic and Self-Configuring Infrastructure

At the heart of the Uli SDK is a dynamic, self-configuring infrastructure. This architecture allows functional modules to be discovered, integrated, and utilized at runtime through unified interfaces. By eliminating static, pre-configured connections, the system achieves true agility, allowing components to be seamlessly added or removed to meet evolving mission requirements.

Multi-Domain Interoperability

The framework extends beyond individual modules to connect disparate systems, including:

- **Aerial and Ground Assets**: Drones and robot arms.

- **Operational Systems**: Weapon systems, process tools, and controllers.

- **Intelligence Layers**: AI agents and data viewers/loggers.

Leveraging a **DoD MOSA-compliant design** (Modular Open Systems Approach), the Uli SDK ensures seamless collaboration and rapid reconfiguration across multiple operational domains.

---

2. Advanced AI Integration and Implementation

MCP-Compliant Middleware

The Uli SDK serves as a first-in-class implementation of the **Model Context Protocol (MCP)** for robotics. It acts as a bridge between an AI's cognitive processes and a robotic system's physical capabilities.

Implementation Detail:

1. **Capability Export**: The SDK allows connected assets to export their data topics and agents.

2. **Discovery Services**: Through dedicated data topic and agent discovery services, clients can identify asset capabilities at runtime.

3. **Markdown Context Strings**: Discovered data and agents provide context strings in markdown format. These strings describe data structures, semantics of enclosed fields, and specific usage cases.

4. **Autonomous Reasoning**: AI agents query the SDK for discovered assets. Using the provided context strings, the AI decides whether to subscribe to data topics or execute specific agents to act upon its reasoning.

A2UI (Agent-to-UI) Framework

The SDK features native **Dart-FFI integration**, creating a powerful A2UI framework. This enables AI agents to drive real-time, high-fidelity user interfaces directly from data topic streams.

Implementation Detail:

- **UI Integration**: Via Dart-FFI, the SDK serves as the backend for Flutter-UI, capable of displaying media contents and 3D drawings.

- **Contextual UI**: Context strings provide the necessary information for the AI to select appropriate Flutter UI Widgets for displaying data or providing user input fields for agent configurations and control parameters.

---

3. Operational Integrity: Security, Safety, and Reliability

- **Security**: The framework utilizes a **certificate-based authentication** model. Clients must obtain explicit, separate permissions for data access (categorized into classified,

controlled, and unclassified tiers) and control access (categorized by Operator, Maintainer, and Administrator roles).

- **Safety**: An integrated safety framework combines proactive detection with a redundant triggering system. It automatically responds to application errors or manual e-stop commands by transitioning the entire system into a coordinated emergency state.

- **Reliability**: Reliability is achieved through a two-tiered state machine system that orchestrates overall application and module behavior. This is supported by distinct operating modes, such as Standard, Reduced, Training, and Maintenance.

4. Development and Deployment Ecosystem

The Uli SDK provides a comprehensive suite of tools to streamline the entire development lifecycle:

- **Code Generation**: Utilities produce concise C++ code for messages, services, and applications, while automatically generating **Python bindings** and **Dart-FFI**.

- **High-Efficiency Building**: The SDK utilizes Google's **Bazel** build tool, supporting cross-building for x86_64 and **NVIDIA Jetson** architectures (Nano, Xavier, Orin).

- **IDE Support**: Developers can utilize the **Google Antigravity IDE**, which possesses a comprehensive understanding of Uli SDK codebases to accelerate development.

- **Deployment**: Included shell scripts simplify the staging and deployment of applications across a network of devices.

Conclusion

The Uli SDK is engineered to deliver a decisive edge in adaptability and operational integrity. By bridging the gap between AI and robotics through MCP compliance and a modular,

MOSA-compliant architecture, it provides a high-quality, cost-effective solution that eliminates vendor lock-in and accelerates the deployment of mission-ready autonomous systems.

For more information, visit www.ulisdk.com.