



中国科学技术大学  
University of Science and Technology of China

# AIOS - 架构分析

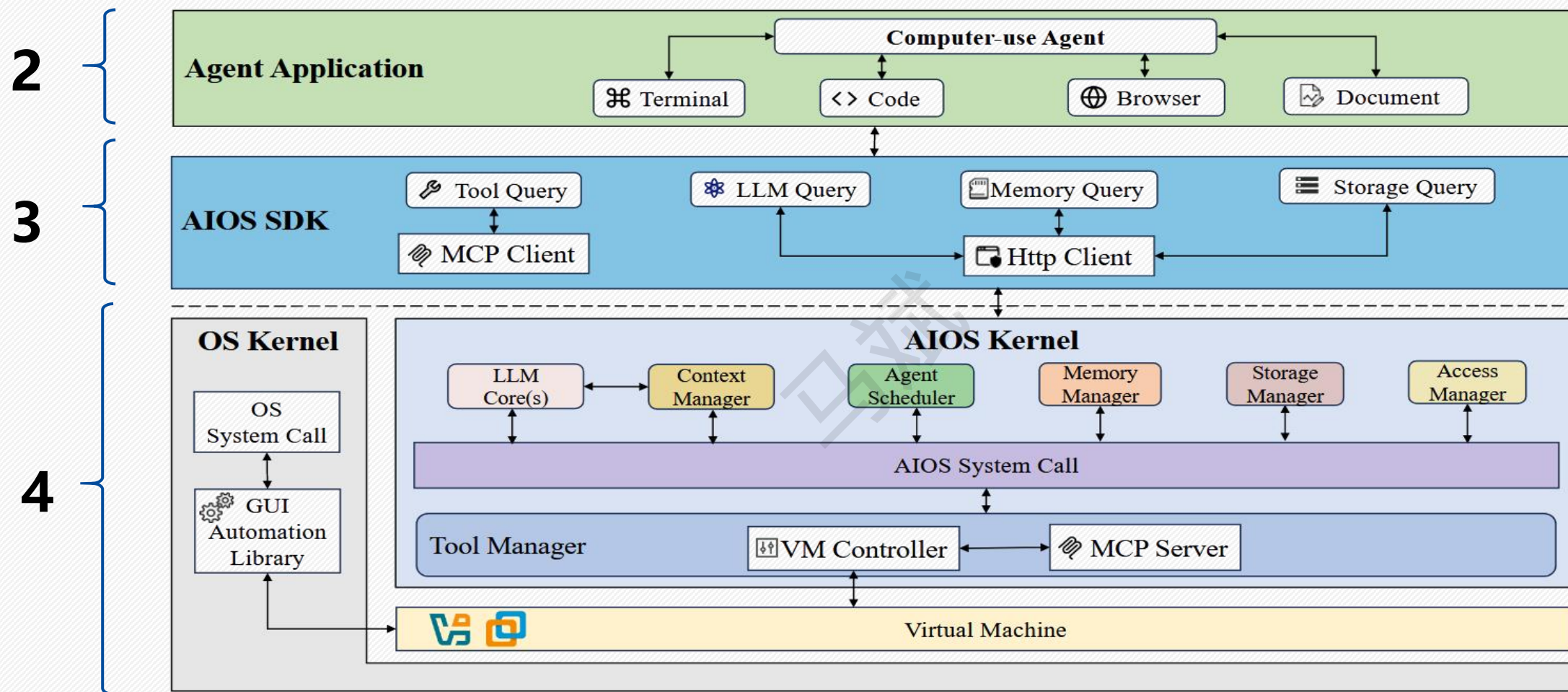
---

AIOS Architectural Analysis Report

马斌

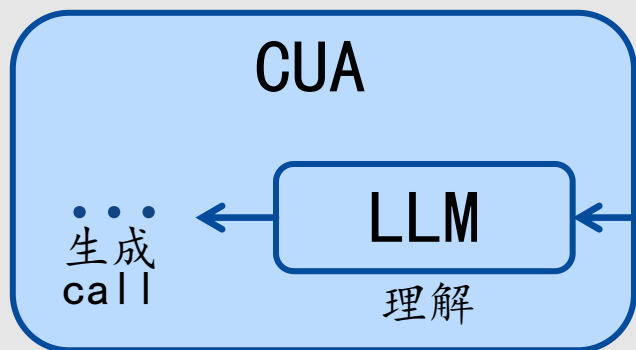
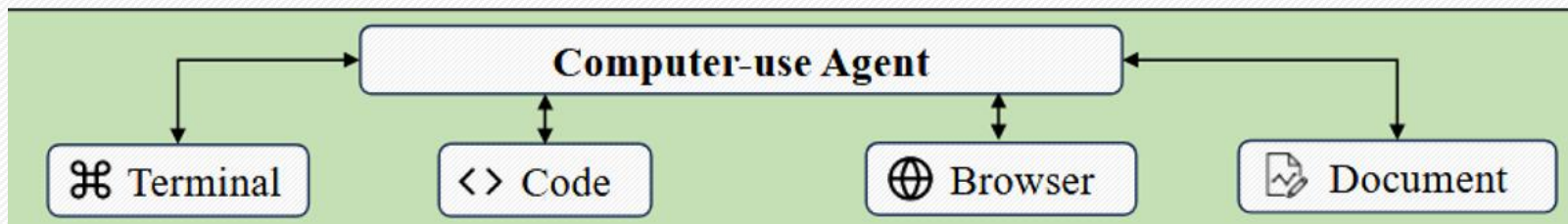
2025-07-18

# 1 总体架构



AIOS Kernel、AIOS SDK (Cerebrum) 及所有直接与GUI交互的部分均在用户本地。

## 2 Agent App



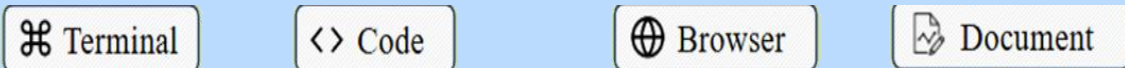
结构化的“上下文环境”

【LLM可理解】

如可语义化编码为JSON

AIOS Kernel

4个能交互的关键组件

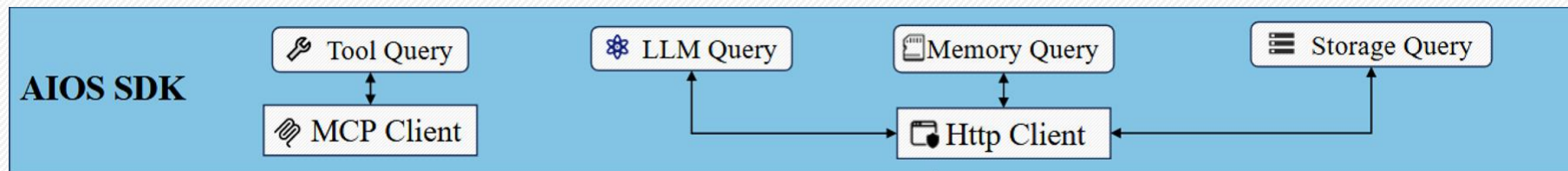


用户 Input

非格式化抽象信息

(如计算机的状态、可执行的操作、界面元素等)

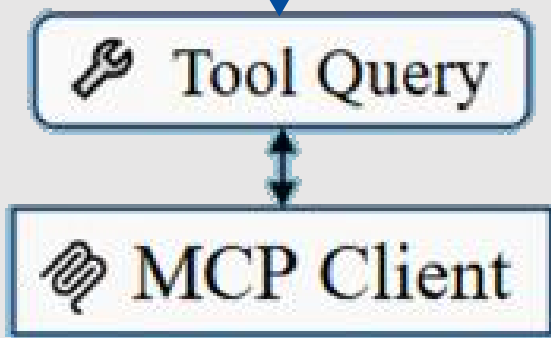
# 3 AIOS SDK



上图仅体现了4种Query分别交由哪个Client完成。

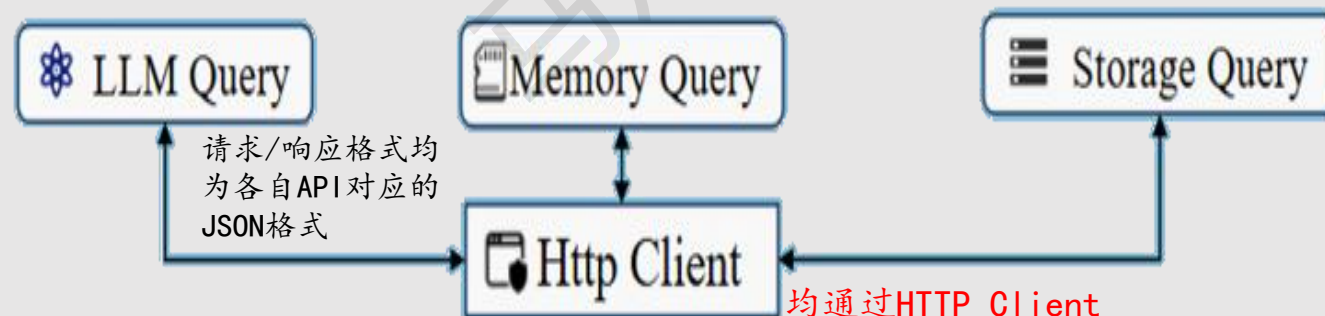
MCP客户端

↓ 工具调用请求



封装有各种Tool, Hub在云端, Agent只需知道Tool的名称、参数即可调用

核心组件(LLM、内存、存储)可能是部署在本地、私有服务器或公共云服务上的(并不一定是LLM在云端, Memory, Storage在本地)



均通过HTTP Client进行通信, 可以灵活地支持这些不同的部署模式。

尽管底层实现可能不同(本地文件系统、远程数据库、云API), 但AIOS SDK通过一个统一的API层(即均使用HTTP Client通信)提供对这些服务的访问, 并在内部处理具体的HTTP请求。

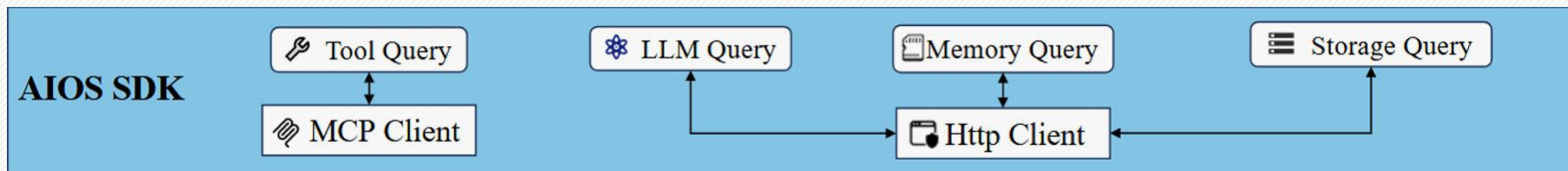


小结

【接口统一、部署灵活】这可能是作者作为一个架构设计者应兼顾到的, 实际应用中可能并不会体现出其灵活性, 毕竟大多是LLM在云端, Memory, Storage在本地。

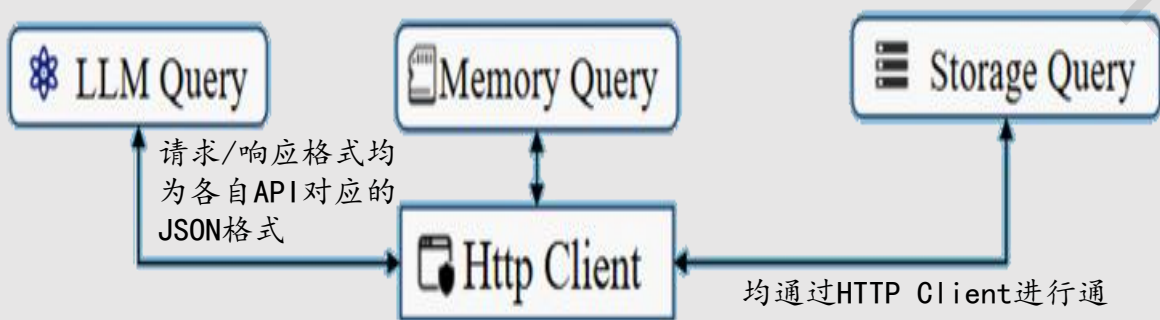


# 3 AIOS SDK



具体文章(Cerebrum)中体现作者同时考虑到云端/本地兼容性的一些证据:

核心组件 (LLM、内存、存储) 可能是部署在本地、私有服务器或公共云服务上的 (并不一定是 LLM 在云端, Memory, Storage 在本地)



尽管底层实现可能不同 (本地文件系统、远程数据库、云API), 但 AIOS SDK 通过一个统一的 API 层 (即均使用 HTTP Client 通信) 提供对这些服务的访问, 并在内部处理具体的 HTTP 请求。



小结

【接口统一、部署灵活】这可能是作者作为一个架构设计者应兼顾到的, 实际应用中可能并不会体现出其灵活性, 毕竟大多是 LLM 在云端, Memory, Storage 在本地。

## Abstract

Autonomous LLM-based agents have emerged as a powerful paradigm for complex task execution, yet the field lacks standardized tools for development, deployment, distribution and discovery of agents. We present Cerebrum, an Agent SDK for AIOS that addresses this gap through three key components: (1) a comprehensive SDK featuring a modular four-layer architecture for agent development, encompassing LLM, memory, storage, and tool management; (2) a community-driven Agent Hub for sharing and discovering agents, complete with version control and dependency management; (3) an interactive web interface for testing and evaluating agents. The platform's effectiveness is demonstrated through implementations of various agent architectures, including

这里理解是蓝色为本地, 绿色为云端

agents. Cerebrum advances the field by providing a unified framework that standardizes agent development while maintaining flexibility for researchers and developers to innovate

lacks a unified platform for researchers and developers to develop, deploy, and distribute their agents, tribute agents and agent components. At the core of the library is a unified framework for constructing diverse agents, containing optimized implementations of popular agent methodologies such as Chain

其接口统一化、部署灵活化根本目的是为了实现代理规范化 (“agent specifications-declarative definitions”)

The framework introduces the concept of agent specifications - declarative definitions that describe an agent's capabilities, resource requirements, and behavioral patterns. These specifications can be composed, extended, and modified dynamically, enabling flexible agent architectures that can adapt to different deployment scenarios. Its implemented resource management strategies include: ① Automatic tool resolution and dependency management. ② Dynamic resource allocation based on agent specifications. evaluation and testing. Our implementations of various agent architectures, including CoT, ReAct, and tool-augmented agents, demonstrate the platform's versatility and effectiveness.

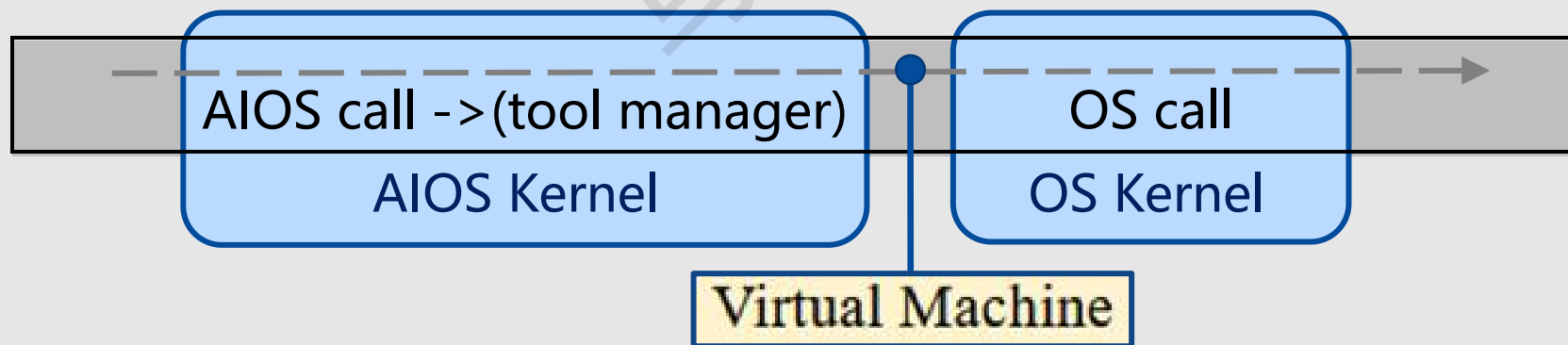
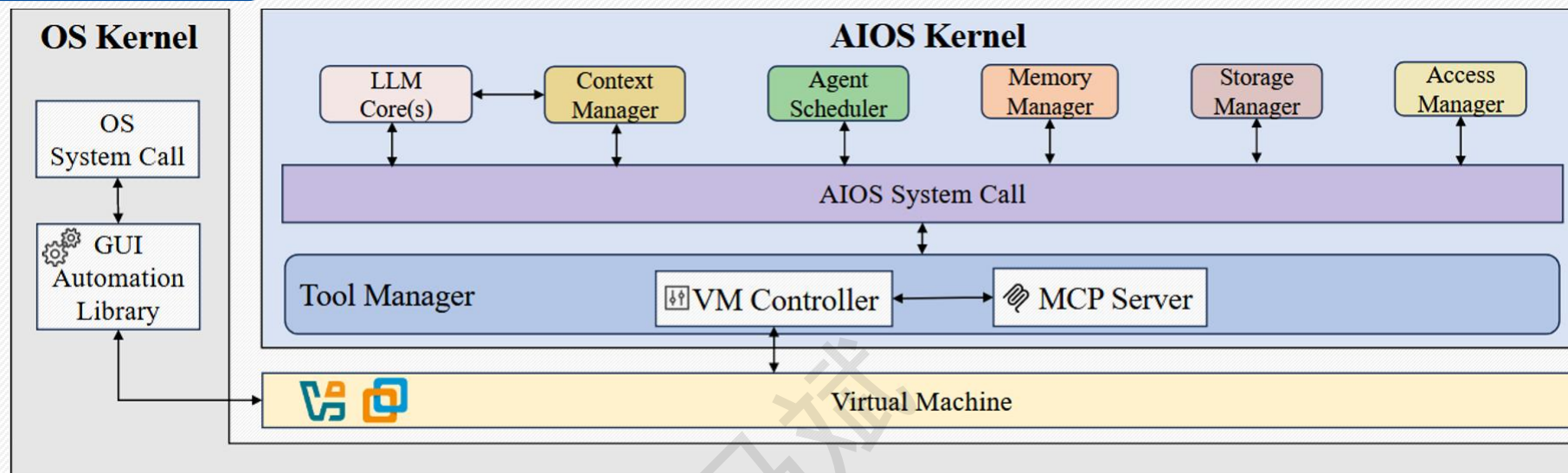
当开发者发布他们的利用 AIOS SDK 开发的 Agent 时, Agent 会被打包成标准化格式 (JSON), 通过 HTTP 协议进行传输到 Agent Hub (云端)。

且文中多处提到该架构是统一化的 (unified)

Cerebrum is designed to provide a standardized framework for developing LLM-based agents, addressing the growing need for systematic agent architectures in the artificial intelligence community. The library implements a modular approach

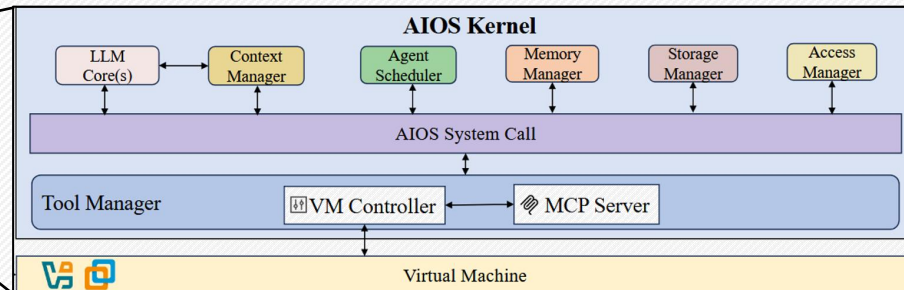
这也与文章的原始动机相符

# 4 Kernel

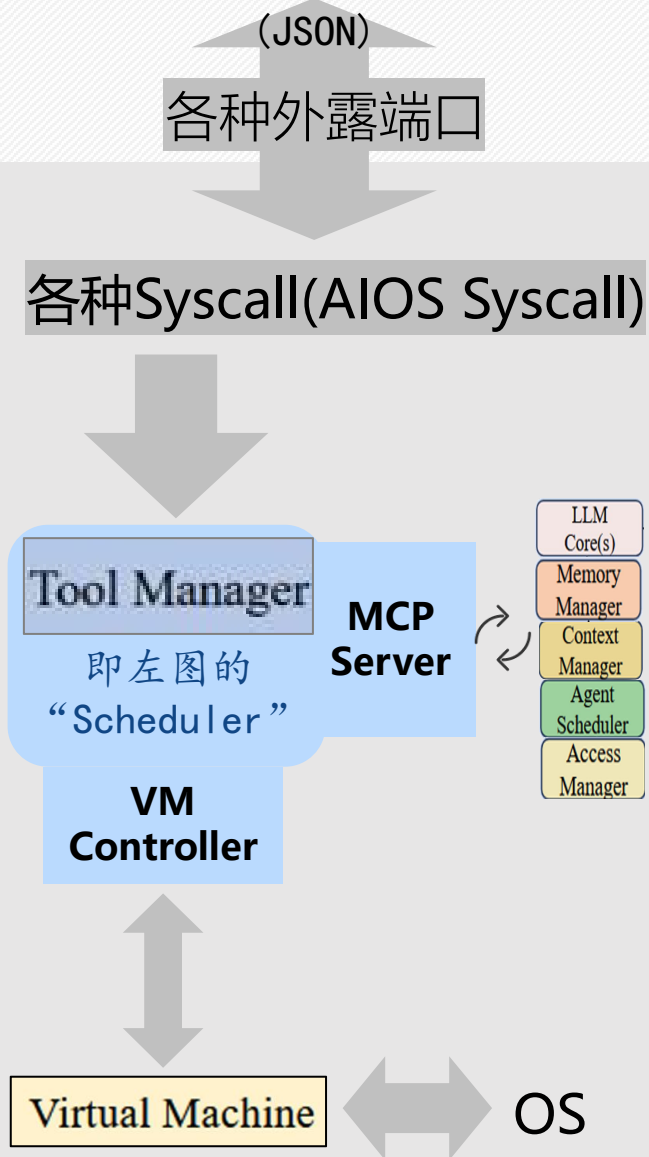
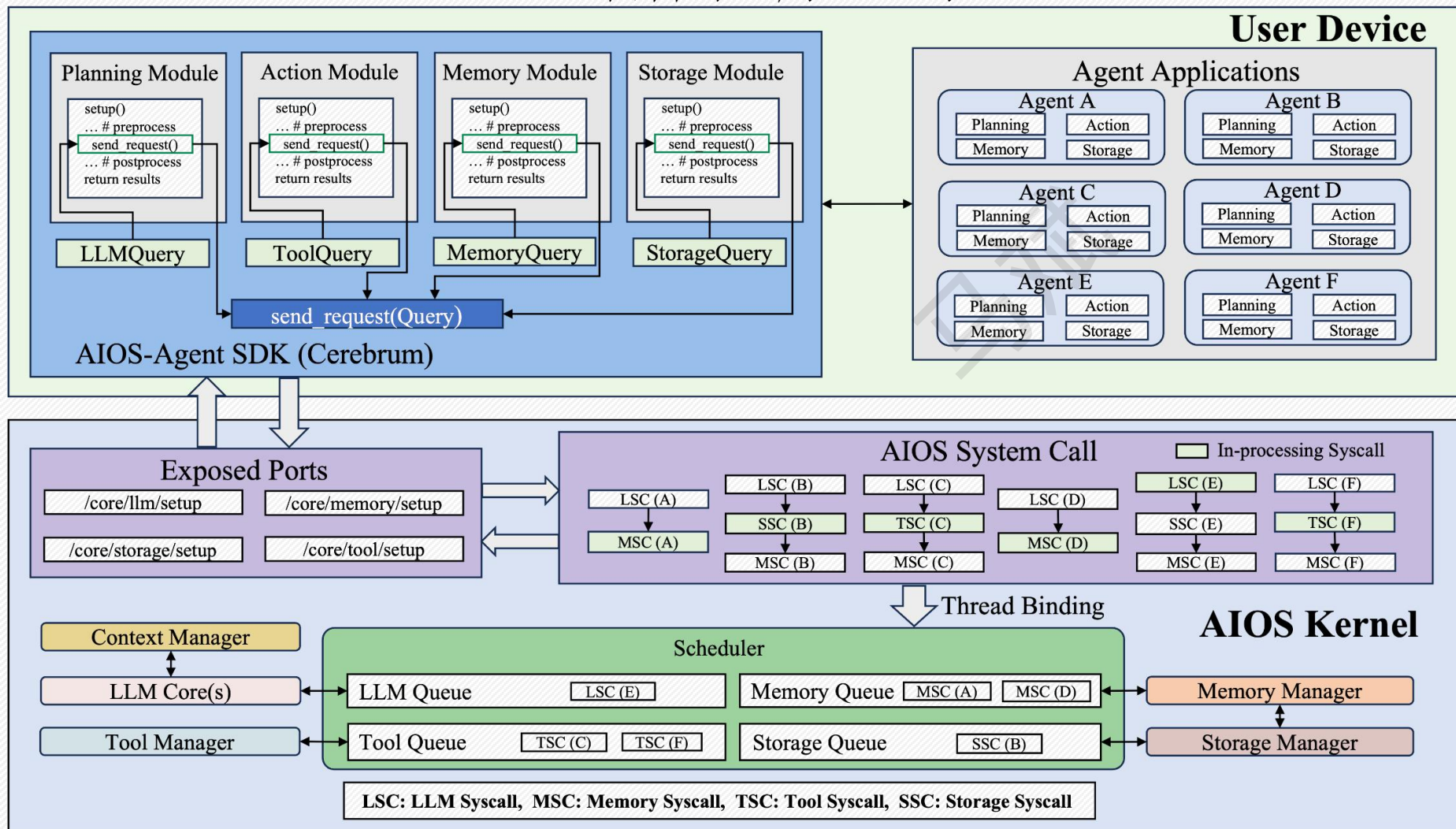


下面主要分析 AIOS Kernel

# 4.1 AIOS Kernel

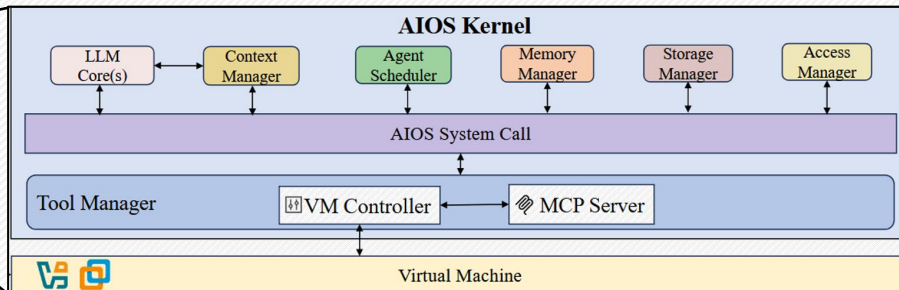


下图来自文章 (Cerebrum)





# 4.1 AIOS Kernel

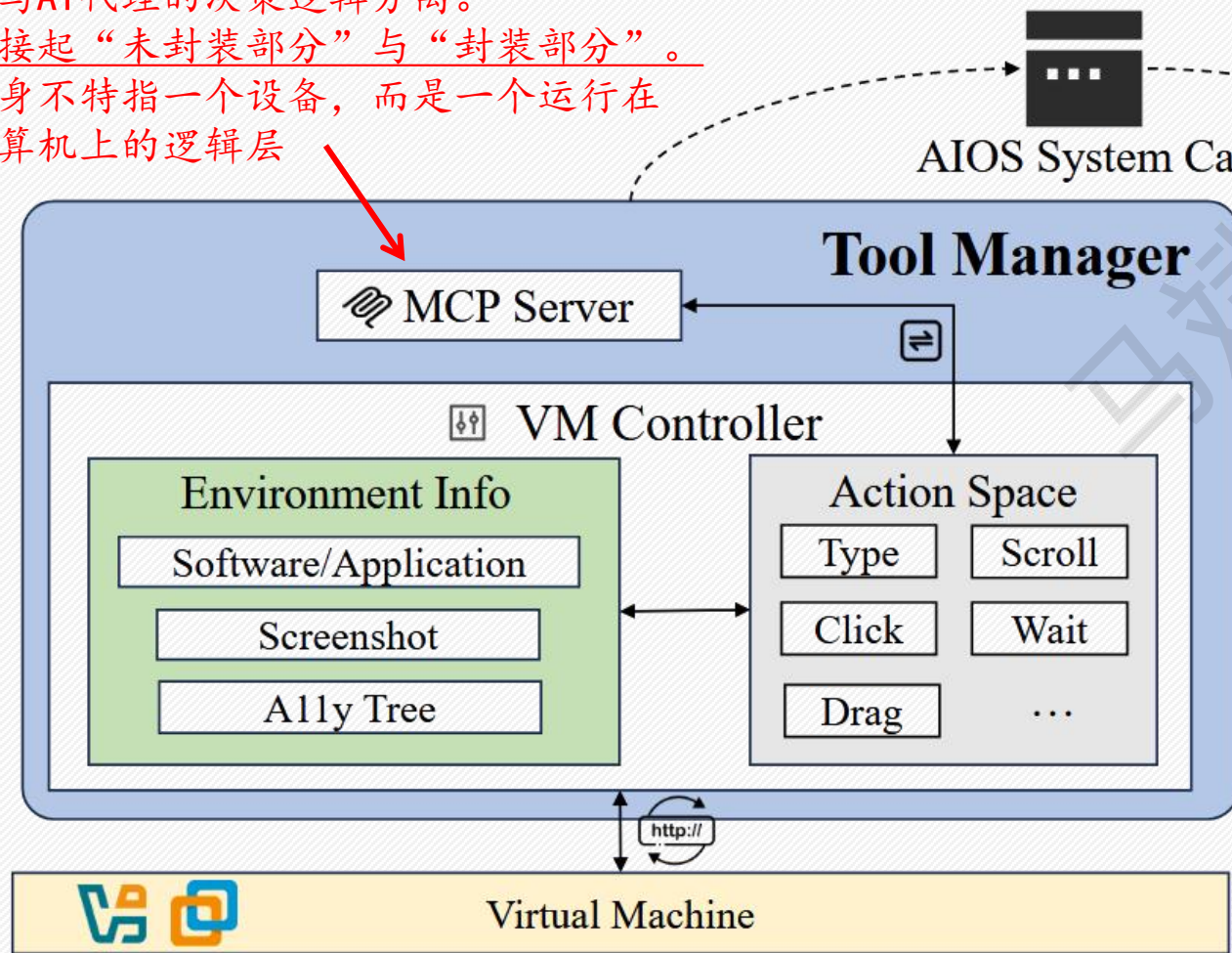


下图来自文章(LiteGUA)

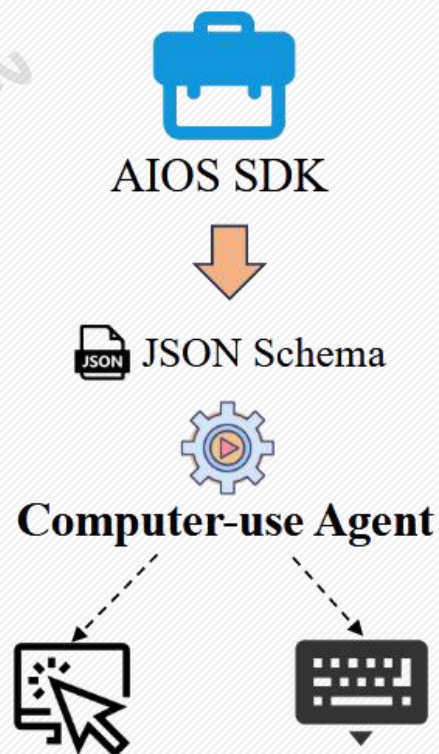
MCP Server将复杂的底层计算机界面细节与AI代理的决策逻辑分离。

连接起“未封装部分”与“封装部分”。

本身不特指一个设备，而是一个运行在计算机上的逻辑层

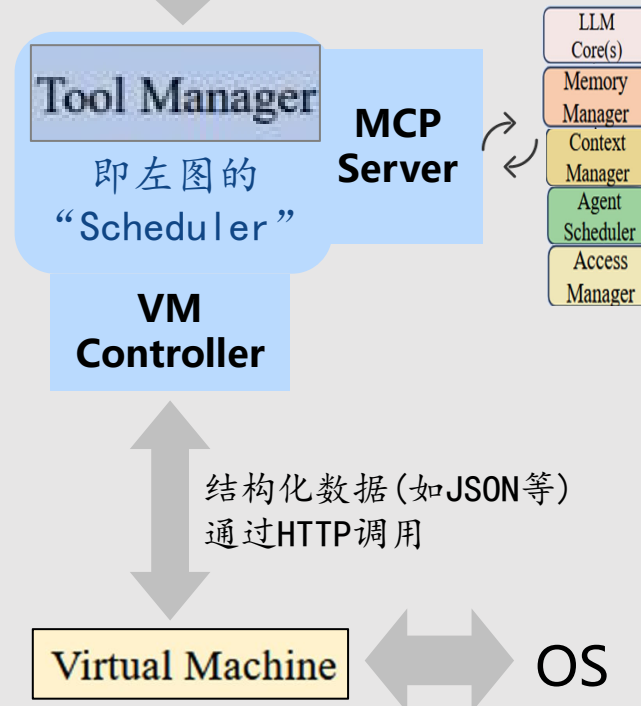


AIOS System Call



(JSON)  
各种外露端口

各种Syscall(AIOS Syscall)







中国科学技术大学  
University of Science and Technology of China

# Thanks !!!

---

A I O S Architectural Analysis Report

马斌

2025-07-18