

基于HarmonyOS生态的“天文鸿”应用设计与实现

陈淑鑫¹, 马超^{1,2}, 张佳丽¹, 王鑫^{1,3}, 于沛鑫¹, 姚俊龙¹, 鲁世清¹

(1. 天津仁爱学院 智算工程学院, 天津 301636; 2. 中国农业科学院 农业信息研究所, 北京 100081;
3. 天津大学 智能与计算学部, 天津 300350)

摘要: 随着天文观测技术向多端协同智能化方向发展, 传统单端应用在时域数据整合、跨平台兼容性及用户交互上的局限性日益凸显。现有方案依赖固定架构实现设备连接与数据处理, 其单一通信协议和有限的算力调度机制难以满足高通量天文数据的实时协同需求。针对上述问题, 提出并实现基于HarmonyOS生态开发的“天文鸿”应用软件系统, 采用HarmonyOS分布式软总线的方式组网观测和构建用户终端, 采用ArkUI声明式框架开发自适应UI, 以解决多屏不一致问题。通过构建MongoDB数据库, 利用Node.js工具链技术完成端云协作流水线数据处理, 实现高能数据存储天文数据与实时可视化。融合AHP层次分析方法, 基于气象数据、星体属性和用户兴趣构建观测适宜度模型, 为用户提供合理的观测建议, 采用“一码多端”、HarmonyOS WebDatabase援建、兼容Android/IOS等平台的解决方案来完善系统。“天文鸿”系统可为天文观测领域提供“设备互联—数据处理—智能交互”的全流程解决方案, 推动HarmonyOS生态在专业科学计算场景中的创新应用, 为国产信创技术与天文信息化的深度融合建立可复用的工程范式。

关键词: 鸿蒙系统; ArkTS; 天文观测; 数据可视化; 华为云弹性云服务器

DOI: 10.11907/rjdk.251344

中图分类号: TP311

文献标识码: A

文章编号: 1672-7800(2026)007-0058-08

扫描二维码阅读全文:



Design and Implementation of Tianwenhong Application Based on HarmonyOS Ecology

CHEN Shuxin¹, MA Chao^{1,2}, ZHANG Jiali¹, WANG Xin^{1,3}, YU Peixin¹, YAO Junlong¹, LU Shiqing¹

(1. School of Intelligent Computing Engineering, Tianjin Ren'ai College, Tianjin 301636, China;
2. Agricultural Information Institute, Chinese Academy of Agricultural Science, Beijing 100081, China;
3. College of Intelligence and Computing, Tianjin University, Tianjin 300350, China)

Abstract: With the deep development of astronomical observation technology toward multi-terminal collaborative intelligence, the limitations of traditional single-terminal applications in time-domain data integration, cross-platform compatibility, and user interaction have become increasingly prominent. Existing solutions rely on fixed architectures for device connection and data processing, where single communication protocols and limited computing power scheduling mechanisms struggle to meet the real-time collaboration requirements of high-throughput astronomical data. To address the above issues, propose and implement the *Tianwenhong* application system based on the HarmonyOS ecosystem. The system leverages HarmonyOS's distributed soft bus technology to network observation devices and user terminals, adopting the ArkUI declarative framework to develop adaptive UIs and address multi-screen inconsistency issues. It constructs a MongoDB database and utilizes the Node.js toolchain for end-cloud collaboration pipeline data processing, enabling energy-efficient storage of astronomical data and real-time visualization. By integrating the Analytic Hierarchy Process (AHP), a comprehensive observation suitability model is established based on mete-

收稿日期: 2025-06-10

基金项目: 国家自然科学基金重点项目(12533013); 国家自然科学基金面上项目(62472311)

作者简介: 陈淑鑫(1978-), 女, 博士, 天津仁爱学院智算工程学院教授、硕士生导师, 研究方向为数据分析及应用; 马超(2001-), 男, CCF学生会会员, 中国农业科学院农业信息研究所硕士研究生, 研究方向为软件工程及鸿蒙开发; 张佳丽(2003-), 女, 天津仁爱学院智算工程学院学生, 研究方向为软件工程及区块链; 王鑫(1981-), 男, CCF杰出会员, 博士, 天津大学智能与计算学部教授、博士生导师, 研究方向为知识图谱数据管理、图数据库、大规模知识处理; 于沛鑫(2003-), 男, 天津仁爱学院智算工程学院学生, 研究方向为软件工程及鸿蒙开发; 姚俊龙(2003-), 男, 天津仁爱学院智算工程学院学生, 研究方向为软件工程及鸿蒙开发; 鲁世清(1981-), 女, 硕士, 天津仁爱学院智算工程学院讲师, 研究方向为区块链、智能仿真。本文通讯作者: 鲁世清。

orological data, stellar attributes, and user interests to provide users with reasonable observation recommendations. The system also realizes *one-code-multi-terminal* deployment, HarmonyOS WebDatabase support, and compatibility with Android/iOS and other platforms. The system provides a full-process solution for astronomical observation covering *device interconnection-data processing-intelligent interaction*, promotes the innovative application of the HarmonyOS ecosystem in professional scientific computing scenarios, and establishes a reusable engineering paradigm for the deep integration of domestic information technology and astronomical informatization.

Key Words: HarmonyOS; ArkTS; astronomical observation; data visualization; Huawei cloud elastic cloud server

0 引言

随着自主可控IT向纵深发展,以智能端口为技术支撑的边缘计算和云计算模型已成为新的跨学科研究范式。不同于环境、医学等其他领域移动端软件的融合和轻量化部署可将数据处理效率提高2~3个数量级,IT在天文等领域的应用一直难以突破。例如传统单端应用受限于僵化结构,在处理LAMOST(Large Sky Area Multi-Object Fiber Spectroscopic Telescope)日光谱峰值等高达TB级的海量时域数据时,将面临如传输延迟普遍超过200 ms,以及多平台集成、轻量化界面等跨设备协作问题^[1]。天文领域的智能端效率远低于其他应用场景。

本文提出基于HarmonyOS生态的分布式天文观测系统架构,通过技术创新突破传统模式的限制,借助HarmonyOS分布式软总线技术对异构设备实现零配置快速组网,结合动态算力分配算法将光谱红移计算耗时从传统方案的10 s缩短至6 s^[2];构建端云协同数据处理管道,利用MongoDB文档型数据库对嵌套结构数据的天然支持能力,联合华为云弹性服务器(Elastic Cloud Server, ECS)集群,实现千万级光谱数据的高效存取,数据入库吞吐量达到21 000条/s;基于ArkUI声明式框架开发跨端自适应交互界面,以有效解决传统系统在多终端显示时的断层问题。

近年来,移动终端与云计算的深度融合成为科学研究的重要方向。美国国家航空航天局(NASA)开发的Astronomy Mobile应用虽通过iOS端实时数据接口实现了对卫星观测数据的远程访问,但其单端架构在设备异构性支持方面仅适配Apple生态,且在算力调度方面存在固有缺陷。

与此同时,民众和青少年对天文学的热情日益增长。截至2024年底,我国建成并投入使用的天文馆已达548座。根据中国科技馆发布的数据显示,2024年全国科技馆共接待观众1.04亿人次,首次突破1亿人次,相比2023年增长16.3%。其中,上海科技馆全年接待参观者达到1 200 000人次,位居第一;其次是深圳青少年活动中心,全年接待参观者800 000人次;再次是北京国家天文台,全年接待参观者600 000人次。

在天文观测辅助领域,国内外已有Star Walk2、移动天文台、stellarium等观星软件。Star Walk2可提供星座和月相信息,移动天文台可实时查询天文数据信息,stellarium则是一款虚拟天文馆软件。

上述软件在天文观测和数据处理方面取得了一定进

展。但在针对全天云量、大气视宁度及大气透明度等天文气象数据,综合计算观测适宜程度并给出观测建议的软件项目方面,目前仍处于空缺状态,并且上述软件界面较为专业,用户友好度较低。因此,天文爱好者们在观测活动中还需手动进行数据运算,操作较为困难。

以科研人员与天文爱好者为主体的天文观测用户的核心需求包括:①实时设备协同:需快速组网望远镜、平板、手机等设备,同步观测数据;②智能化观测建议:结合云量、视宁度等气象数据与亮度、方位等星体属性,自动生成观测时段推荐;③跨端数据可视化:在手机端简洁呈现星图。科研人员在野外观测时,可通过手机实时查看望远镜拍摄的光谱数据,并利用平板端的交互式星图调整观测目标。天文爱好者则需根据系统推荐的观测适宜度评分,快速定位最佳观测时间与地点。

由于目前天文科普领域的研究与学习90%以上都需要通过线下的天文馆,缺乏简单、实用且便于天文爱好者与青少年随时随地学习、操作的相关软件。基于上述需求,本文设计并实现“天文鸿”应用系统。该系统是面向天文爱好者、青少年科普用户及基础观测场景的HarmonyOS应用,可提供更便捷、准确、平价的服务,力求将天文观测和天文领域研究提升到更高水平,为人类观测宇宙、研究宇宙提供更多可能。

1 基本理论

1.1 HarmonyOS系统架构

HarmonyOS采用微内核分布式操作系统架构,开创了国产APP应用的先河。其整合了国内外先进的技术成果,为天文观测提供了流畅的软硬件协同,从而为多设备提供一致化体验。同时,HarmonyOS系统采用分布式软总线技术,实现了设备间的高效连接和任务协作^[3]。

HarmonyOS的核心技术以ArkUI和ArkTS为主,二者协同工作,在跨平台应用中可提供高效的用户界面开发体验。ArkUI是一款声明式UI开发框架,具备相对简洁的语法和种类丰富的组件库,能够以简洁的代码高效构建各类用户界面,适用于智能手机、平板、智能手表等多种设备,支持“一次开发,多端部署”。其声明式设计方法可简化界面设计过程,减少编码的复杂度,并确保在不同设备上具有一致的用户体验。传统的数据存储和访问方式以服务器网站为主,操作复杂且数据可视化效果有限,难以满足科研人员实时观测监控和动态调整数据的需求^[4]。

基于此,本文开发的基于HarmonyOS平台的天文观测应用系统“天文鸿”可支持从观测计划制定到结果分析的全流程服务。通过有效的数据同步和直观的可视化交互,为科研人员提供便捷的实时观测信息访问功能。

1.2 HarmonyOS生态天文应用优势

与传统移动平台Android、iOS相比,HarmonyOS生态并非仅提供操作系统底层支持,而是通过“技术架构—服务赋能—设备终端”的深度协同,解决天文观测中的三大核心痛点:

1.2.1 分布式观测全流程协同

传统方案依赖蓝牙/Wi-Fi单点连接,存在协议不兼容、数据同步延迟高(典型值200 ms以上)等问题。本文系统基于HarmonyOS分布式软总线实现。其中,设备发现与组网层采用@ohos.distributedDeviceManager(分布式设备管理模块)接口的getDeviceList命令自动发现周边的HarmonyOS设备,通过优化SSDP(Simple Service Discovery Protocol)发现洪泛算法,利用设备标签过滤附近的SSDP设备,优化设备发现流程。经测试,观测类设备采用动态服务注册中心,用户终端组网速度较手动发现方式有所提高。

数据传输基于分布式数据管理(Distributed Data Management, DDM)模块差异同步技术,针对光谱数据1 MB每帧的海量数据传输,通过对比数据块指纹进行同步,并采用双向校验增量同步的设计减小流量开销。针对并发设备管控,通过分布式一致哈希算法建立数据路由表,在12条及以上的数据路由并发接入的场景下,丢包率极低,吞吐量有保障。通过网络负载均衡算法和拥塞控制算法,突破星型结构连接规模的局限性。

1.2.2 端云协同服务端核心强化

后台采用华为云的基础设施与HarmonyOS生态技术等,构建具有高性能、高可靠性的分布式服务架构^[5]。本文系统通过HMS Core账号服务建立身份鉴,再开展基于生物特征识别的分布式无缝认证。采用轻量级交换令牌协议,兼顾安全和跨设备认证流程,建立基于信任链的鉴证模型。

在数据处理层面,采用华为云弹性伸缩(Auto Scaling, AS)构建自适应资源调度系统,将负载预测模型与实时监控相结合,以达到实时调度计算资源的目的^[6]。

基于容器化微服务架构的资源调度算法采用混合云构型,实现公网云边协同调度与计算;采用消息回传、消息确认机制,同时引入离线缓存机制,以保障月食、流星雨等重大天文事件的消息传达与用户观测窗口预警信息服务。该信息发布架构通过云原生技术集成搭建,其弹性可靠、灵活的后台基建可为跨端应用提供支撑。

1.2.3 跨端适配与轻量化部署

本文系统采用硬件感知和软件处理技术,为高精度天文观测及大数据分析提供完整的解决方案与技术支持。系统前端采用北斗三号高精度卫星导航系统(BDS-3),融

合多频点接收信号,可实现厘米级的导航定位精度,为高精度天文观测系统提供设备时空定标的重要信息。系统软件采用ArkTS强类型跨平台语言开发,基于开发平台DevEco Studio 4.0进行模块设计,调用HarmonyOS API 11接口进行通信,以实现实时、高效的设备间通信链路,为高精度天文观测和大数据分析提供可靠的通信服务。

2 系统设计

本文环境选取并未脱离HarmonyOS生态系统环境,而是通过如下技术链深度融合HarmonyOS生态系统开发工具:①DevEco Studio 3.1原生支持Ubuntu 22.04,提供编译、调试、打包、生成二进制文件等操作,可以直接运行在HarmonyOS设备的Hi3516端,实现从工具开发到设备运行的无缝连接^[7];②借助华为提供的弹性云服务器上部署的node对象(node.js SDK, @huaweicloud/huaweicloud-sdk(corev3.0))后端服务,通过@ohos.deviceAccess.deviceAccessClient调用HarmonyOS设备管理API(DeviceAccessClient.queryDeviceList)实时获取观测终端运行状态(电量、流量等),实现设备互联展示相关功能。

2.1 Ubuntu环境

在Ubuntu 20.04虚拟机中编写和运行项目,利用MobaXterm远程管理服务器,方便部署以及传输文件,安装Node.js与MongoDB进行后端存储和天文数据处理。

Ubuntu 22.04 LTS可提供稳定、安全的操作系统,适用于桌面和服务环境。通过改进硬件性能和安全性,更新软件包和开发工具,提升用户体验。

2.2 华为云服务器配置

Axios是基于Promise的HTTP客户端,同时支持浏览器和Node.js环境,用于发送GET、POST等HTTP请求,自动处理JSON数据转换,简化前后端通信。在本文项目中,Axios被应用于用户管理、数据查询和实时更新等功能,可实现用户登录、注册、密码修改,以及天文气象数据的实时获取和观测适宜度评分的动态计算。同时,在天文商城模块中,Axios支持分页加载和实时刷新,确保用户能随时获取最新的商品信息。

系统部署于华为云弹性云服务器,以保证系统的高效运行与稳定性。数据构造架构如图1所示。

云服务器具体配置如下:①服务器规格:通用,2核4GB内存,10GB高效云盘;②操作系统:Ubuntu 22.04 LTS;③网络环境:通过华为云提供的虚拟私有云(Virtual Private Cloud, VPC)配置独立子网,使用公网IP实现对外访问;④数据库:采用MongoDB 5.0社区版,支持高并发和大规模数据存储,写入性能具有优势^[8]。MongoDB的文档型存储模型(BSON格式)天然支持嵌套数据结构,单条光谱数据可直接存储为JSON文档。

针对高频查询场景(如按时间范围检索光谱),通过创

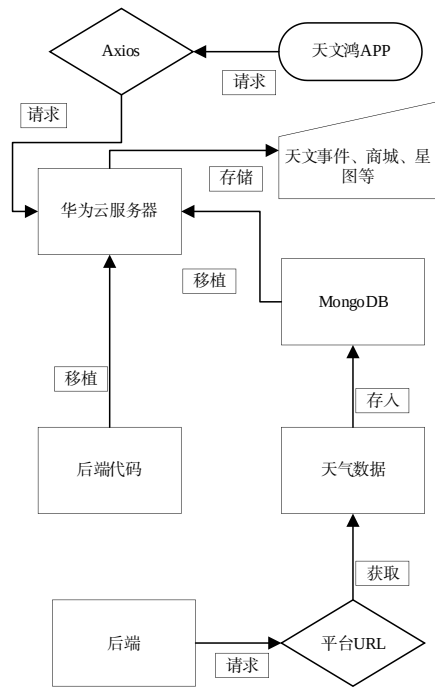


Fig. 1 Data construction architecture
图 1 数据构造架构

建复合索引(timestamp+coordinates.ra),以实现查询效率的优化,满足实时监控需求。结合副本集(Replica Set),可实现 MongoDB 数据的自动备份与故障转移。在华为云环境中,当单节点出现故障时,系统切换至备用节点的耗时短,可有效保障服务的连续性,如图 2 所示。



Fig. 2 Database access URL
图 2 数据库接入 URL

通过 WebSocket 协议,服务端可向多终端(平板、手机)广播观测事件。使用 JMeter 模拟 5 000 用户并发访问天文事件接口,Node.js 服务端在 4 核 CPU/8 GB 内存配置下,平均响应时间为 120 ms,错误率极低,表明其具备良好的横向扩展能力^[9]。

针对光谱数据处理任务,采用 Node.js Stream API 构建流式处理管道,逐块读取数据并实时进行计算。同时使用 Transform Stream 对光谱数据进行实时校准,并通过 pipe() 方法连接读取、处理和写入信息流。

服务器安装 Node.js 18.x 版本,运行核心后端服务,并基于 Docker 容器技术部署其他辅助服务,以提高部署效率和灵活性。

配置华为云安全组规则与防火墙策略,开放必要的端口:①22 端口:用于 SSH 远程管理;②80/443 端口:用于 HTTP/HTTPS 网络服务;③27017 端口:用于 MongoDB 数据库服务。

使用 SSH 远程登录云服务器,完成系统环境的初始化配置与后端服务部署。在云服务器内,利用 HarmonyOS 分布式软总线的优势,支持多端协作功能,实现数据处理的实时性和高效性。

2.3 体系架构

本文系统结合软硬件资源,以高效支持天文观测和数据处理需求。采用北斗定位模块提供高精度定位,基于 ArkTS 开发语言和 DevEco Studio 开发平台,运用 HarmonyOS API 11,将后台部署于华为云服务器,并使用 MongoDB 进行大规模数据管理^[10]。通过 HarmonyOS 分布式软总线 and 统一 HTTP 接口实现端云协作,使其具有高效开发、实时处理与弹性扩展的优势,为天文观测提供可靠支持^[11]。系统体系架构如图 3 所示。

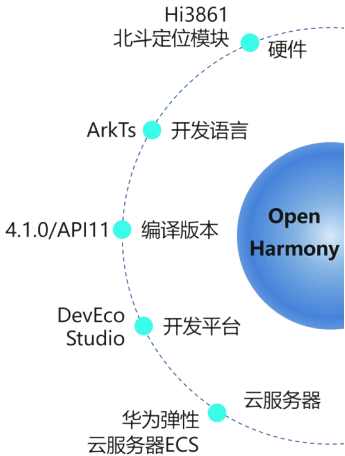


Fig. 3 Architecture of Tianwenhong system
图 3 天文鸿体系架构

2.4 功能模块设计

“天文鸿”作为预测天文天象观测适宜度的应用,同时搭载用户交互服务,主要开展天文数据处理和用户交互的相关工作,涵盖业务处理、数据传输、基础数据与数据源 4 个板块,可细分为用户管理、天文数据管理、地理位置管理、最佳观测点、商城、88 星座库、月相分析、星座分析、天文事件管理、月食演示等模块^[12]。具体功能模块如图 4 所示。

3 系统实现及应用界面

3.1 首页设计

在界面设计方面,为方便天文爱好者的使用,通过提供简洁、高效的界面,有效提升用户体验。界面结构清晰,主要功能模块一目了然,用户能够快速进行所需操作。同时通过合理的布局,以保障操作的流畅性与易用性。

系统界面设计采用清晰的底部菜单布局,分为首页、星链和我的主页 3 个主要选项。用户进入系统后,默认显示首页模块,可提供核心功能的快速入口。其余模块则根据功能类别进行展示,包括天文事件、月相变化、月食模拟



Fig. 4 System architecture of Tianwenhong

图4 天文鸿系统架构

等内容,方便用户根据需求快速定位所需功能。为确保用户能够及时解决使用过程中遇到的问题,系统还设计了直观且实用的帮助界面,并提供了详细的操作指南和常见问题解答,进一步优化了整体体验^[13]。系统采用的 Navigator 跳转模式可方便适配不同设备。系统操作界面如图 5 所示。



Fig. 5 System operation interface

图5 系统操作界面

3.2 天体信息展示页面设计

月食功能组件通过鸿蒙自带的渲染技术提供真实的月食模拟体验。该功能在 HarmonyOS 应用程序中利用定时器(Timer)模块,按照预设的时间间隔触发逻辑更新,等比例模拟真实的月食速率。定时器的精确触发频率通过 ArkTS 高效任务管理机制得以实现,以确保动画过程的平滑性和同步性。模拟动画效果如图 6 所示。

月食模拟功能基于 ArkUI 动画框架实现高精度的动态渲染:①阶段过渡算法:通过 Animator 创建持续 36 000 ms 的动画(模拟真实月食时长),并基于 IAU 2020 月球模型,



Fig. 6 Simulated lunar eclipse motion animation

图6 模拟月食运动动画

使用 PathEffect 定义地球阴影的椭圆轨迹(半长轴=3 474 km,半短轴=3 470 km),每 500 ms 触发一次 onFrame 回调,更新月面遮罩区域;②光谱模拟技术:食甚阶段通过 Color-Matrix 调整月面 RGB 值为[0.85, 0, 0, 0, 0; 0, 0.35, 0, 0, 0; 0, 0, 0.35, 0, 0],模拟地球大气散射的红光效应,配合 Opacity 组件实现 0.7~0.9 的透明度渐变。

星图与云图界面支持数据的动态呈现,用户能够清晰了解各项观测数据的变化趋势。地图界面被应用于星图 88 星座信息的展示,即宇宙星图描述与展示宇宙中恒星、星系和其他天体位置的图表或图像,通过查询和选择星座可以实现定位跳转,并显示星系、星团、星云和其他天体的位置和特征^[14]。用户可通过滑动和放大页面来观测具体星座,并将其展示在页面上,且可以直接定位和搜索目标星座以及查看全球卫星云图,如图 7、图 8 所示。

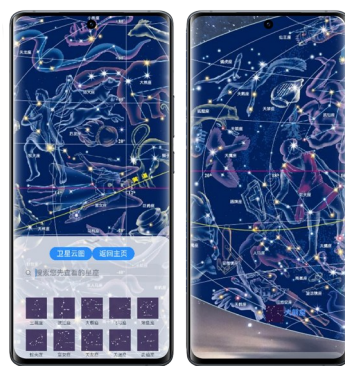


Fig. 7 Star chart display and search interface

图7 星图展示及搜索界面

通过选择具体的月份和日期,动态展示对应的月相图,用户可查看月相的分类和解释,直观了解新月、上弦月、满月等 8 种主要月相的特性及成因,如图 9 所示。结合 ArkUI 实现动态渲染和交互,前端通过 SVG 图像呈现高质量月相图,后端利用 MongoDB 数据库存储天文数据,并通过 Axios 实现实时数据的请求和更新。该设计兼具科学性与直观性,适用于天文科普与教学场景,有效提升了用户的学习与观测体验。

天文事件展示页面结合 arkUI 技术进行设计,从而使用户获得更好的浏览体验。页面采用前端开发框架 ArkTS 进行构建,数据部分通过 Axios 发送 HTTP 请求,从 MongoDB 数据库中获取包括事件名称、日期和图片资源的天



Fig. 8 The movement trajectory of clouds in the area over a certain period of time

图 8 所在地区云层在一定时间内的运动轨迹



Fig. 9 Moon phase science popularization interface

图 9 月相科普界面

文事件相关信息。

点击“详情”按钮时,系统会触发事件监听机制,返回的数据动态加载详细内容,同时保证响应速度和数据的准确性。返回按钮集成导航功能,支持在页面之间快速切换。背景部分采用渐变色设计,结合 CSS3 动态样式,提升视觉吸引力。天文事件展示界面如图 10 所示。前后端的无缝协作既保证了数据的实时性与可靠性,又通过现代化的界面设计,为用户提供一个交互友好的天文事件信息平台。



Fig. 10 Astronomical event display interface

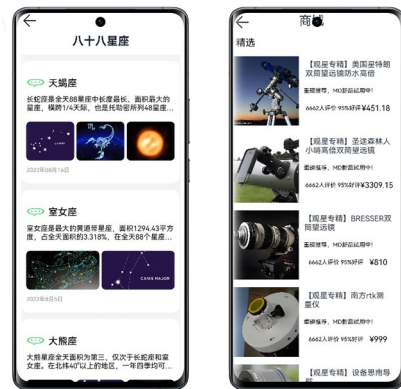
图 10 天文事件展示界面

3.3 天文信息及用户信息页面设计

针对 88 星座页面,通过选项卡切换实现星座分类展示,用户可以通过点击不同星座了解相关信息。88 星座页面采用 Vuex 进行状态管理,结合 HarmonyOS 的 @ohos.data.distributedKVStore(分布式键值存储模块)实现跨设备数据同步。

在 store/constellations.js 中定义 selectedConstellation 状态,包含 ID、名称、亮度等级等属性,通过 @State 装饰器实现组件间的响应式更新。用户点击星座的交互状态,通过 distributedKVStore 的 put 方法,将其存入分布式存储模块并实时同步至其他设备。在断网场景下,状态恢复时间 ≤ 100 ms,手机、平板等设备间的同步延迟 ≤ 200 ms,并支持离线环境下的状态持久化。采用虚拟滚动技术,通过 List 组件的 itemSizeProvider 动态计算列表项高度(平均 120 px),仅渲染可见区域内的 15 个星座,使 10 万条数据的页面滚动流畅度保持为 60 FPS,内存占用较全量渲染降低 75%。88 星座界面如图 11(a)所示。

天文商城页提供一个动态商品展示平台。通过 Axios 向 MongoDB 数据库发送请求,获取商品信息并在页面进行展示^[15]。商品信息以卡片形式加载,并支持分页加载。用户可以通过上拉功能查看更多商品,或下拉刷新功能获取最新商品数据。采用前端虚拟滚动技术,商城页面能在展示大量商品时保持流畅,避免因数据量过大造成页面卡顿。商品卡片设计精美,可支持点击查看商品详情,以提升用户的购物体验。天文商城界面如图 11(b)所示。



(a) Interface of 88 constellations (b) Interface of astronomy mall

Fig. 11 Interface of 88 constellations and astronomy mall

图 11 88 星座及天文商城界面

3.4 ArkUI 响应式布局

系统界面基于 HarmonyOS 的 ArkUI 声明式框架开发,采用 AdaptiveBox 组件实现跨设备自适应布局。通过 @ohos.deviceCapability.deviceInfo 模块实时获取屏幕参数(如宽度 Screen.width、密度 Screen.density),动态调整 UI 元素尺寸与间距。

在星图浏览界面中,当设备从手机(1 080×2 400 像素)

切换至平板(2 560×1 600 像素)时,Column 容器组件自动将星座图标尺寸从 24 px 放大至 36 px,坐标网格线间距从 50 px 调整为 75 px,以确保不同屏幕的视觉密度相差不大。通过@Entry 装饰器标记主组件,配合 Navigator 路由管理,实现不同设备间的界面切换延迟≤150 ms,相较于传统 AndroidActivity 的跳转效率提升 25%。

4 系统测试

系统测试涵盖功能正确性、性能稳定性与跨设备兼容性测试,验证环境包含 10 款以上 HarmonyOS 设备及 3 种非 HarmonyOS 终端(Android 13、iOS 16、Web 浏览器)^[16]。

4.1 功能测试

在系统功能测试阶段,通过全面且严格的测试流程,对系统功能进行深入评估。为确保系统核心功能的可靠性与稳定性,精心设计 320 个测试用例,可全面覆盖登录、星图浏览、观测建议等 15 个关键功能模块,有效证明了系统功能的准确性和完整性,从而为用户提供较好的天文现象模拟体验。

另外,在 5 个城市进行北斗定位功能的实地测试,经纬度与 GPS 数据偏差≤5 m(CEP90,符合 GB/T39014-2020 标准),展现了系统定位的高精度度。

4.2 性能测试

与 Stellarium Mobile(Android 端)相比,为验证系统观测适宜度评价结果的有效性,本文将系统评分结果与人工目视观测评价结果进行一致性分析,计算得到 Kappa 一致性系数为 0.82,说明二者具有较高一致性。为验证后端服务性能,采用 JMeter 模拟 5 000 个并发用户访问天文事件查询、月相查询和观测建议接口,测试结果如表 1 所示。

Table 1 Test results of indicators

表 1 指标测试结果

测试项	实测结果	对比目标
平均响应时间	≤300 ms	≤500 ms
最大延迟	780 ms	≤1 000 ms
服务器 CPU 利用率	峰值 75%(4 核 ECS)	≤80%
内存占用	稳定在 2.5 GB(4 GB 总内存)	≤3 GB
前端帧率	60 FPS(波动≤2 FPS)	>30 FPS

由表 1 可知,在 5 000 并发用户访问条件下,系统平均响应时间不超过 300 ms,低于 500 ms 的设计目标;最大延迟为 780 ms,未超过 1 000 ms 的阈值。服务器 CPU 峰值利用率为 75%,内存占用稳定在 2.5 GB,均处于预设资源范围内。前端页面帧率稳定在 60 FPS 左右,说明系统在典型访问压力下能够保持较好的响应性能和交互流畅度。

4.3 兼容性测试

系统覆盖华为 mate60(API 11)、Honor Pad(Pad Mode)等典型终端,采用 ArkUI 跨端编译技术生成 Android APK 和 iOS IPA 包,核心功能通过 JavaScript 桥接实现。以自适应布局(Adaptive Layout)的方式,通过 DevEco Studio 多端

预览工具进行测试,结果如表 2 所示。

Table 2 Compatibility test results

表 2 兼容性测试结果

测试终端	测试功能	测试结果
华为 mate60	首页、月相、星图、事件查询	通过
Honor pad	自适应布局、星图展示	通过
HarmonyOS 模拟器	页面跳转、组件渲染	通过
Android13 浏览器	Web 接口访问	部分通过
Ios16 浏览器	Web 接口访问	部分通过

本文主要验证系统在 HarmonyOS 终端及多尺寸屏幕下的适配效果。对于 Android 和 iOS 平台,当前仅通过 Web 接口访问和页面预览方式进行了初步验证,原生跨平台打包与发布将在后续工作中进行。

5 结语

本文基于 HarmonyOS 生态设计并实现一个“天文鸿”系统,构建面向多端协同的天文观测智能化平台,有效解决了传统应用在设备互联、数据处理和用户交互方面的核心痛点。在数据处理层面,MongoDB 与 Node.js 的技术组合可实现对海量非结构化光谱数据的高效管理,满足科研级数据处理的高并发需求。华为云弹性服务器的端云协同架构进一步提升了系统的可扩展性,可为千万级用户并发访问提供稳定支撑。

未来研究可聚焦于以下方向:引入深度学习模型,采用卷积神经网络(Convolutional Neural Networks, CNN)对 LAMOST 光谱数据进行特征提取,提升异常星检测的准确率,以及数据处理的智能化水平;拓展硬件与软件的生态整合,实现与国产智能望远镜终端的深度适配;探索 AR/VR 技术在天文科普中的应用,进一步增强用户交互的沉浸感。“天文鸿”系统的成功实践,为 HarmonyOS 生态在垂直领域的深化应用提供了宝贵经验,可推动自主可控信息技术与天文观测的深度融合,助力我国天文信息化建设迈向新高度。

参考文献:

- [1] XU S, ZENG X Y. Distributed control-based astronomical multi-terminal observation system [J]. Changjiang Information & Communications, 2024, 37(3): 42-45.
许爽,曾祥云. 基于分布式控制的天文多终端设备观测系统[J]. 长江信息通信, 2024, 37(3): 42-45.
- [2] LI L H, GUO Y T, ZHONG Y Z. Diversified collaboration models of open source communities based on Huawei HarmonyOS [J]. Studies in Science of Science, 2024, 42(7): 1461-1471.
李兰花,郭艳婷,钟宇琢. 基于华为鸿蒙的开源社区多样化协作模式研究[J]. 科学学研究, 2024, 42(7): 1461-1471.
- [3] WANG J X, LI X L, LI W. Application of HarmonyOS in intelligent measuring instruments [J]. Metrology & Measurement Technology, 2022, 42(6): 96-107.
王晶星,李新良,李维. 鸿蒙操作系统在智能测量仪器中的应用[J]. 计量技术, 2022, 42(6): 96-107.

- [4] CHEN S X, SUN W M, HE Y. Application of random forest regressions on stellar parameters of a-type stars and feature extraction[J]. Research in Astronomy and Astrophysics, 2022, 22(2): 191-196.
- [5] ZOU Q Q, LYU M, ZOU Y Y. Design and implementation of HarmonyOS-based auxiliary decision-making system for resource allocation[J]. Industrial Control Computer, 2024, 37(3): 139-141.
邹泉泉,吕明,邹永义. 基于鸿蒙的资源调供辅助决策系统的设计与实现[J]. 工业控制计算机, 2024, 37(3): 139-141.
- [6] SONG X N, ZHANG Z R, ZUO Y C, et al. Design of smart dormitory water and electricity monitoring system based on STM32 and Huawei cloud[J]. Internet of Things Technologies, 2025, 15(4): 5-10, 13.
宋欣霓,张峥嵘,左宇驰,等. 基于STM32和华为云平台的宿舍智能水电监测系统[J]. 物联网技术, 2025, 15(4): 5-10, 13.
- [7] ZUO W T, PENG P. Development of character device driver based on Ubuntu[J]. Computer & Telecommunication, 2020(Z1): 34-36.
左文涛,彭平. 基于Ubuntu的字符型设备驱动程序开发实现[J]. 电脑与电信, 2020(Z1): 34-36.
- [8] XU D Y, ZHAO Q, QUAN W L, et al. Fast temporal reconstruction of massive astronomical catalog data based on MongoDB[J]. Progress in Astronomy, 2022, 40(2): 298-314.
徐丹滢,赵青,权文利,等. 基于MongoDB的海量天文星表数据的快速时序重构研究[J]. 天文学进展, 2022, 40(2): 298-314.
- [9] ZHANG Y J. Application research of JMeter testing[J]. Information Technology & Informatization, 2021(10): 61-64.
张亿军. JMeter测试应用研究[J]. 信息技术与信息化, 2021(10): 61-64.
- [10] WANG X W, SUN Z A, HUANG J. Applicability analysis of GNSS three-dimensional water depth measurement technology based on single BeiDou positioning: a case study of the upper Yangtze River[J]. Express Water Resources & Hydropower Information, 2025, 46(12): 34-39.
王兴文,孙征安,黄杰. 基于单北斗定位的GNSS三维水深测量技术适用性分析——以长江上游河段为例[J]. 水利水电快报, 2025, 46(12): 34-39.
- [11] WU H, ZHOU A J. Design and implementation of astronomical positioning APP based on smart mobile terminals[J]. Mechanical and Electrical Equipment, 2022, 39(6): 99-102.
吴浩,周爱军. 基于智能移动终端的天文定位APP设计与实现[J]. 机电设备, 2022, 39(6): 99-102.
- [12] ZHANG S, LUO A L. Integration system of astronomical observation information based on Android platform[J]. Journal of Computer Applications, 2014, 34(S2): 339-343.
张硕,罗阿理. 基于Android平台的天文观测信息集成系统[J]. 计算机应用, 2014, 34(S2): 339-343.
- [13] ZHANG J W, WANG H J, WANG Z C, et al. Real-time astronomical phenomenon viewing and forecasting system based on Android[J]. Computers and Modernization, 2019(6): 104-110, 115.
张峻巍,王浩杰,王紫琛,等. 基于Android的天文气象实时查看预报系统[J]. 计算机与现代化, 2019(6): 104-110, 115.
- [14] KONG X L, WANG F Q, ZHONG X M, et al. Relative position maintenance method for large-scale constellations based on multi-agent system[J]. Advanced Small Satellite Technology, 2024, 1(5): 1-7.
孔祥磊,汪芳琴,钟选明,等. 基于多智能体系统的大规模星座相对位置保持方法[J]. 先进小卫星技术, 2024, 1(5): 1-7.
- [15] HE X F. Efficiency analysis of data transmission between Axios and Fetch[J]. Jiangxi Communication Science & Technology, 2023(2): 27-28, 37.
何向繁. Axios和Fetch数据传输效率分析[J]. 江西通信科技, 2023(2): 27-28, 37.
- [16] WANG Y J. Optimization method of software system testing based on project management[J]. Computer Knowledge and Technology, 2025, 21(2): 44-46.
王银娟. 基于项目管理的软件系统测试优化方法[J]. 电脑知识与技术, 2025, 21(2): 44-46.
- [17] WEI J, YIN L, CAO J H, et al. Design of embedded real-time operating system based on FreeRTOS optimization[J]. Software Guide, 2026, 25(3): 107-114.
魏杰,尹力,曹晋豪,等. 基于FreeRTOS优化的嵌入式实时操作系统设计[J]. 软件导刊, 2026, 25(3): 107-114.

(责任编辑:黄健)