

doi: 10.7690/bgzdh.2022.06.006

基于 Cesium 的 3 维态势显示系统

张富强

(中国人民解放军 91404 部队, 河北 秦皇岛 066001)

摘要: 针对现有 2 维态势显示不能满足现代战争需求的问题, 提出基于 Cesium 的实时 3 维态势显示系统的设计方案。简要介绍 Cesium 系统的特点与优势, 构建基于 Cesium 态势显示系统的架构, 对综合集成、实时数据驱动等关键技术进行了详细分析。结果表明: 该方案简便可行, 能较好地完成实时 3 维态势显示。

关键词: Cesium; 态势显示; GIS

中图分类号: TJ01 **文献标志码:** A

3D Situation Display System Based on Cesium

Zhang Fuqiang

(No. 91404 Unit of PLA, Qinhuangdao 066001, China)

Abstract: Aiming at the problem that the existing 2D situation display can not meet the needs of modern warfare, a design scheme of real-time 3D situation display system based on Cesium is proposed. This paper briefly introduces the characteristics and advantages of Cesium system, constructs the architecture of Cesium-based situation display system, and analyzes in detail the key technologies such as comprehensive integration and real-time data-driven. The results show that the scheme is simple and feasible, and can realize real-time 3D situation display.

Keywords: Cesium; situation display; GIS

0 引言

随着现代战争形态向陆海空天电等多维、多域演化, 现有 2 维态势显示形式已不能直观、形象地表达多维、多域、复杂的态势, 不能完全满足态势显示、分析的需求。随着信息技术发展, 特别是 3 维图形技术的发展使得 3 维形式的态势显示以及虚拟现实、增强现实等形式显示成为可能。

Cesium 是一个开源的 JavaScript 库, 用于 Web 浏览器中的创建地图服务。Cesium 广泛支持各种不同的地图数据源, 例如国外的 Google Maps、微软的 Bing Maps、开源的 OpenStreetMap, 国内的天地图, 以及任意基于 ArcServer 或 GeoServer 发布的 WMS 和 WMTS 服务, 也可以用简单的图片作为地图源。其优势有成本低、开发简单、支持多种数据可视化方法、2 维 3 维一体化、良好的跨平台性、计算精度较高等。通过 WebGL 的硬件渲染加速, Cesium 能顺利支持最新的 HTML5 浏览器, 不需要安装额外的扩展插件, 适合于 WebGIS 轻量级用户的开发需求, 可有效提高用户体验度^[1]。

作为开源的 3 维 GIS 系统, Cesium 支持完整的 GIS 服务功能, 提供包括地理坐标系与笛卡尔坐标

系之间的转换、矩阵运算、动态投影等算法。以此为基础进行二次开发, 可大幅减少开发工作量, 专注于业务相关功能开发, 具有开发周期短、成本低等特点。笔者针对通用 3 维态势显示需求, 提出具体可行的方案, 对同类型研究具有一定的参考价值。

1 系统总体设计

1.1 需求分析

为表现战场真实的情况, 态势显示应是实时、动态变化的过程。除了地理信息等基础数据外, 其他信息应随时间的变化实时更新。3 维态势显示系统应支持以下功能^[2]:

- 1) 地理信息显示: 包括卫星地图、数字高程数据、地名地物等地理环境信息。
- 2) 实体模型显示: 作战平台等装备实体的 3 维模型, 并在数据驱动下实时更新位置(经度、纬度、高度)、姿态(航向、俯仰、滚转)、航迹等信息。
- 3) 战术信息描绘: 与作战行动相关的点、线、面、箭头等标绘。
- 4) 观察和显示操控: 提供不同视角下观察战场环境和实体模型跟踪观察功能。
- 5) 动画特效: 提供毁伤、火焰、爆炸、烟雾等

收稿日期: 2022-02-22; 修回日期: 2022-03-28

作者简介: 张富强(1983—), 男, 河南人, 硕士, 工程师, 从事舰艇装备试验研究。E-mail: zfq0111@163.com。

特效功能。

由于 Cesium 提供了完整的 GIS 功能支持，地理信息显示、实体模型显示、战术信息描绘及显示控制等功能只需在 Cesium 库基础上进行适当的二次开发即可实现。笔者以 Cesium 与具体的业务应用相结合实现通用态势显示为目标，重点分析系统架构设计及相关关键技术并提出具体可行的方案。

1.2 架构设计

系统架构设计采用了基于浏览器的 B/S 架构(浏览器/服务器)，由数据层、服务层、界面层 3 部分组成。数据层提供高清卫星图像数据、数字高程数据(digital elevation data, DEM)、实体 3 维模型数据以及驱动态势更新的实时态势数据。服务层提供基于 Http 协议的 Web 服务器和基于 WebSocket 的数据代理 2 种服务，Web 服务器用于提供图像、高程、模型等数据资源服务；数据代理负责实时数据的收发。界面层由浏览器负责实现 2 维、3 维态势实时显示及操控。系统架构如图 1 所示^[3]。

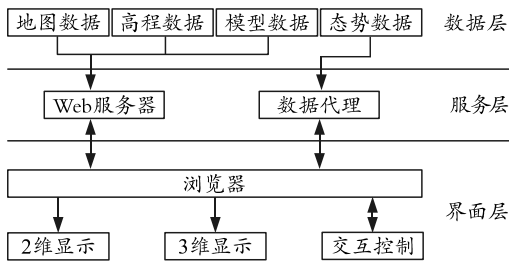


图 1 系统架构

这种基于 B/S 的架构，态势显示算法由 Web 浏览器实现，而事务处理算法由服务层实现，两者通过网络协议进行信息交互。用户端通过浏览器向 Web 服务器发出数据请求，Web 服务器根据用户端数据请求调用相应文件，解释执行后，将处理结果返回给浏览器，浏览器接收到返回的页面内容，进行动态解释执行后形成可视化的结果展示给用户。该模式具有分布式、易扩展、易部署(平台自带浏览器)、易维护(服务器端改变显示内容可实现用户端同步更新)、动态显示等特点。

为尽可能减少开发的难度及成本，提高效率，系统采用成熟技术和公开资源。数据层采用公开的卫星地图影像数据及 DEM 数据源。服务层 Web 服务器采用成熟的商用 Web 服务程序(如 Apache、Node.js、Tomcat 等)，也可按照 Http 协议自主开发相关功能模块。界面层采用 Cesium 开源库，实现基础的 GIS 功能，并在此基础上采用 JavaScript 进行二次开发，进行功能的拓展。

由于 Cesium 本身采用了 B/S 架构，区别于现有仿真系统、推演系统等常用的 C/S 架构，导致无法直接嵌入现有业务系统。这种架构上的不兼容性成为了制约 Cesium 在态势显示领域广泛应用的主要原因。为了解决此问题，系统采用基于 WebSocket 协议的数据代理方式，打通 Cesium 态势显示模块与现有业务系统的数据关联接口，实现 Cesium 态势显示模块与业务系统数据实时/准实时双向交互和显示模块的灵活接入，实现与现有业务系统综合集成。

1.3 信息流程

信息交互主要通过服务层的 Web 服务器和数据代理实现。

Web 服务器提供卫星地图、数字高程、实体模型等资源服务。用户端通过浏览器观察 3 维态势并进行态势交互控制。浏览器根据用户控制及视角变化，动态向 Web 服务器请求并加载所需数据。

数据代理提供实时态势数据交互支持。数据代理首先与业务系统建立连接并获取基础数据，用户端浏览器通过 WebSocket 协议与数据代理建立持久连接，数据代理保存并维护连接状态，将基础数据分发给各用户端。用户端根据所接收数据进行初始化，加载相关地图及模型数据并渲染形成 3 维态势。而后每个数据采样周期或仿真运行周期，数据代理实时接收动态态势数据，通过 WebSocket 向用户端点播/广播，用户端接收数据后进行滤波、外推等处理后实时驱动实体模型运动。同时，用户通过操作界面对装备、实体进行操控，通过数据代理实时反馈到业务系统，并进入下一周期循环。

2 关键技术

由于 Cesium 提供了通用的地理信息显示控制等功能支持，笔者重点分析了 Cesium 态势显示模块与业务系统综合集成、实时数据驱动、3 维军标标绘等需要自主研发、影响部署应用的关键技术。

2.1 综合集成方法

由于 Cesium 态势显示模块采用 B/S 架构，无法直接嵌入现有业务系统，就需要采用数据代理这种“中间件”的方式实现与现有业务系统综合集成。支持 B/S 架构的数据代理协议主要有 HTTP 协议和 WebSocket 协议^[4]。

HTTP 协议具有格式简便、实现容易、传输速度快等特点，被广泛用于浏览器与服务端间的数据

通信。HTTP 协议使用“请求—响应”模式，用户端获得数据前，必须向服务器主动发送请求。该特性使得服务器不能实时主动将数据推送给用户端，以致出现数据传输延迟。为避免这种延迟，一般采用 Ajax 轮询的模式。Ajax 轮询就是浏览器以一定时间间隔不断的发送请求，造成了用户端和服务器的资源消耗，同时对服务器并发性和处理速率的要求也较高。

WebSocket 协议同样在 TCP 协议基础上实现，在完成初始化的握手过程后，就通过 TCP 建立持久连接，实现用户端和服务器的全双工异步通信。异步通信就是用户端或服务端在监听 WebSocket 对象的事件(如建立连接和接收消息等)时，无需阻塞等待监听结果，而是等事件发生时，利用回调函数的方式进行处理。全双工异步通信使得 WebSocket 协议网络数据延迟更低，而且无事件发生时，可以去处理其他事务而不是阻塞等待。基于这一特点，利用 WebSocket 协议服务器能实现真正的推送(Push)技术和信息实时传输。HTTP 协议和 WebSocket 协议对比如图 2 所示。

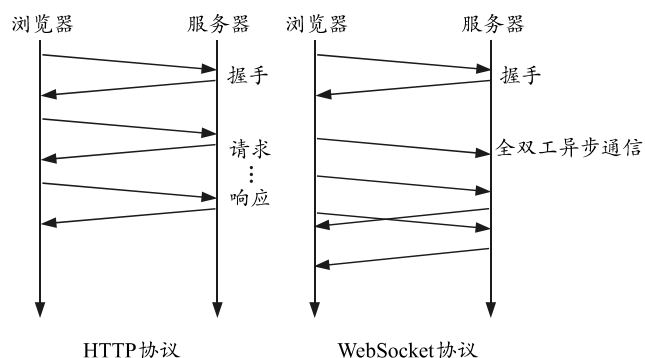


图 2 HTTP 协议和 WebSocket 协议对比

由于 WebSocket 协议的全双工异步通信和真正意义的服务器推送技术，使得 B/S 架构也具备了 C/S 架构的实时通信能力。本系统采用 WebSocket 协议实现数据代理与 Cesium 态势显示模块实时数据交互，数据代理再通过现有业务系统的协议与业务系统达成实时数据交互，进而实现与现有业务系统综合集成。

2.2 实时数据驱动方法

现有业务系统的真实采集数据或仿真系统生成数据通常是离散的，且具有一定周期。尤其是平台位置姿态等数据，对态势的显示至关重要。这些周期性数据的离散性对于 3 维态势显示的影响较 2 维更为明显，尤其是 3 维态势中实体模型位置、姿态

等的变化会出现明显的跳变、抖动现象，对态势观察分析产生不利影响。产生跳变、抖动一般有 2 方面原因：1) 数据的周期性；2) 数据随机误差。

为减少这种影响，需将驱动 3 维实体模型运动的有异常值、帧频低、随机误差大的实时测量/仿真模拟的轨迹数据进行平滑滤波、插值外推等处理，使其能驱动 3 维实体模型平稳逼真地进行运动模拟。数据接收模块通过网络接收到的各类平台离散的轨迹数据解析转换后存于共享缓存区。而后，由数据处理模块进行平滑滤波和插值外推，实现平台位置、姿态的平滑过渡和变化。平滑滤波用于减少数据异常值、随机误差等影响，为兼顾流畅性和准确性，可采用卡尔曼滤波或交互式多模型滤波等方法剔除异常值和消除误差^[5]。插值外推用于实时计算关键帧之间某一时刻平台的瞬时运动及姿态参数。首先根据历史数据中平台的运动要素，构建其运动动力学模型；再利用最近时刻的运动要素进行插值外推算出当前时刻平台的运动参数。平台作直线运动时采用线性插值可以理论上接近真实数据，效果较为理想；对于曲线运动，还需考虑运动要素的变化情况，近似地模拟出平台运动效果^[6]。

该方法短时间内误差较小，但随着时间积累，外推数据的误差也会不断积累逐渐超出可接受范围；因此，还需结合实际情况，提高业务系统源数据的精度和帧率，兼顾精度和效率的情况下不断优化态势显示的效果。

2.3 3 维军标标绘

军标标绘是采用军队标号这种抽象的图形符号来表现不同类型武器装备及其作战行动。军标标绘作为战场态势一种重要表现形式在 3 维态势显示中也是不可或缺的。军队标号均是由一些基础的图形、文字组合而成。为提高灵活性，标绘过程中可将军标图元数据生成与军标显示分别处理，当需要调整参数修改军标形状、颜色时，只需重新生成军标图元数据。军标显示，只需对抽象的各种图元数据进行处理，这样也提高了处理速度。

军队标号主要有点状军标、线面状军标^[7]。点状军标的特点是不依比例尺标绘，即军标作为一个整体，其大小不随地图缩放而变化。根据这一特点，标绘点状军标首先按类型生成对应图元数据，然后在 3 维场景中以“公告板”形式在指定位置按设定的比例、旋转等参数加载并显示；线面状军标的特点是依比例尺标绘，图形与地图贴合并随地图缩放、

旋转。线面状军标标绘要先确定控制点，可通过鼠标点选方式，或加载文件方式确定。而后将控制点的位置数据，通过相应模型计算生成该军标的图元数据，最后加载并显示在 3 维场景中。其中图元数据计算是这一过程的关键，以箭形军标为例介绍计算过程。

根据箭形军标的形状特征，将其分为箭头和箭身 2 部分，设置 8 个控制点：1 个箭头点，2 个箭耳点，2 个箭颈点，2 个箭尾点和 1 个中心点，对应图 3 中的 A~H 点。除箭身的 BG、AF 2 条边为 Bezier 曲线外其余均为直线。为保证操作的简便，需要根据输入的箭头点 D 和 2 个箭尾点 A、B 这 3 个控制点来采取相应算法计算其他控制点坐标^[7]，并可通过拖动控制点方式来对形状进行调整，达到合适的视觉效果。控制点生成算法如图 3 所示。

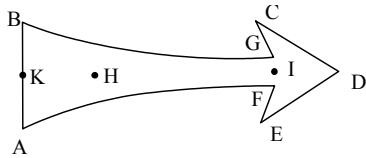


图 3 箭形军标控制点

1) 通过 A、B 2 点坐标(Ax, Ay)、(Bx, By)计算 AB 中点 K 点坐标(Kx, Ky);

2) 中心控制点 H 与 K 点距离设定为整个箭身长度 KD 的 1/5, 可计算出中心控制点 H 坐标为(Hx, Hy);

3) 箭头长度(D 点到线段 GF 中点 I 的距离)设定为箭身长度(D 到线段 AB 中点 K 的距离)的 1/5, 可计算出点 I 坐标(Ix, Iy);

4) 箭耳宽度 CE 设定为整个箭头长度的 2 倍, 可计算出点 C 坐标(Cx, Cy)、点 E 坐标(Ex, Ey);

5) 箭颈宽度 GF 为箭耳宽度的 1/2, 可计算出点 G 坐标(Gx, Gy)、点 F 坐标(Fx, Fy)。

3 系统实现

3 维态势显示的核心是将战场态势用 3 维形式进行展示。基于以上分析，系统开发中重点实现了 3 维地理信息加载与显示、3 维模型显示与控制、军标标绘等功能。

3.1 地理信息加载与显示

在服务端，利用 HttpServer 软件将卫星地图瓦片数据、数字高程数据等地理信息集中存储于服务器，通过文件服务形式提供给用户端。Cesium 模块自动通过 Http 协议加载相应的地理信息数据。基础

信息显示效果如图 4 所示。

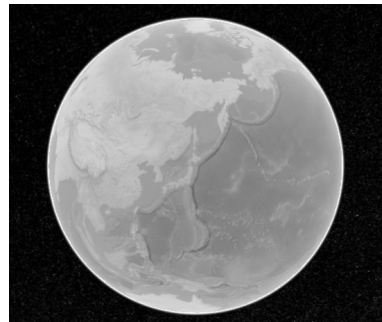


图 4 地理信息显示效果

用户端可通过鼠标进行旋转、缩放等基本操作，通过菜单选择不同的卫星地图。通过脚本控制相机位置、旋转等参数，从而控制视角变化。

3.2 3 维模型显示与控制

Cesium 支持 gltf 和 glb 这 2 种基于 web 的 3 维模型行业标准格式，常用模型格式可通过专用软件转换成 gltf/glb 格式。系统初始化时，通过添加实体(entity)并修改其模型(model)属性的方式，向场景中添加 3 维模型；而后每个数据周期，根据实时接收态势数据修改实体的位置、方向、旋转、缩放等参数，使模型状态与数据保持一致。模型显示效果如图 5 所示。



图 5 3 维模型显示效果

3.3 军标标绘

军标标绘是通过抽象的图形符号表现不同类型武器装备及其行动。其中：点状军标以“公告板”形式在指定位置按设定的比例、旋转等参数加载并显示；线面状军标，采用动态生成图元数据的方式加载到 3 维地图场景中。

标绘过程首先以鼠标在地图上点选的方式增加控制点，鼠标移动时根据鼠标所在位置作为控制点动态生成“公告板”或图元数据在地图上显示，双击结束绘制。根据控制点参数通过脚本或鼠标点选方式可对军标进行控制点修改、旋转、缩放等操作。军标显示效果如图 6 所示。