

IT 系统结构关系可视化系统^①

曹瑞斌^{1,2}, 廉东本²

¹(中国科学院大学, 北京 100049)

²(中国科学院 沈阳计算技术研究所, 沈阳 110168)

摘要: IT 系统结构关系可视化系统是一个利用三维可视化技术构建业务至 IT 设施的可视化管理模型, 以立体、仿真、实时互动的方式, 呈现系统业务、系统、流程以及支撑的物理设备之间关系的 Web 系统^[1]. 可以直观的展示不同层次应用节点的业务关系、应用与组件之间所属关系、逻辑组件与物理实体的对应关系、应用业务流向关系、系统监控信息和按条件显示拓扑等, 并且支持针对某一节点发生故障的定位及影响分析、故障回溯功能. 可视化系统数据来源于配置管理模块, 后端使用 Struct2+Hibernate4+Spring3 技术, 前端可视化模块使用 Unity3D 引擎. 整个可视化系统具有使用方便、扩展性好、面向服务、支持配置化、数据呈现直观等特性.

关键词: 系统结构关系; 可视化管理模型; 三维可视化技术; 面向服务; Web 系统

Visual System for IT System Structure

CAO Rui-Bin^{1,2}, LIAN Dong-Ben²

¹(University of Chinese Academy of Sciences, Beijing 100049, China)

²(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

Abstract: IT systems' structural relationship visualization system is a web system, which utilizes three-dimensional visualization technology to build the business to IT visual management model and presents the relationship among the system's business, systems, processes and physical devices supported by three-dimensional, simulation and real-time interaction^[1]. It can visually show business relationship in different levels of application nodes, owner, relationship between the application and the relevant components, correspondence relationship between the logical and physical components of the entity, flow relationship in business applications. Moreover, it can achieve system monitoring information, display topology according to the condition and so on. In the meantime, it supports for a failure's node position, influences' analysis and fault's backtracking function. Visualization system data is from the configuration of the management module including its backend using Struct2+Hibernate4+Spring3 technology and its front visualization module using Unity3D engine. The whole visualization system has the characteristics of easy to use, good expansibility, service-oriented, configuration support and intuitive data presentation.

Key words: systems' structural relationship; visual management model; three-dimensional visualization technology; service-oriented; web system

1 引言

系统结构关系是指系统内部各组成要素之间的相互联系、相互作用的方式或秩序, 即各要素在时间或空间上排列和组合的具体形式^[2]. IT 系统结构关系表现为子系统之间的面向业务的协作关系、系统与各要素之间的组成关系、各要素之间的相互联系与相互作用.

用. IT 系统结构关系可视化系统致力于 IT 系统结构关系的可视化, 对系统管理者对系统的全面认知具有重大意义.

随着我国提出的信息化战略的逐步实施, 我国企业的信息化水平不断提高, 逐步建立了各种 IT 系统, 各系统之间关系错综复杂, 对系统管理者维护系统提

① 基金项目: 国家水体污染控制与治理科技重大专项(2012ZX07505004)

收稿时间: 2016-04-21; 收到修改稿时间: 2016-05-26 [doi:10.15888/j.cnki.csa.005545]

出了巨大的挑战. 当前国内外大多数呈现系统结构的方式是以文档的方式来记录系统结构关系, 以人为的方式查看、分析、判断系统应用之间的关系, 不能实时监控系统业务负载量, 不能准确的定位故障, 不能快速地分析故障对其它组成要素的影响程度, 而且人力成本高, 效率低^[3]. 鉴于此, 本文设计了 IT 系统结构关系可视化系统. 针对 IT 系统结构关系提出利用三维可视化技术构建业务至 IT 设施的可视化运维管理模式, 能够实时、准确地呈现 IT 系统结构关系及监控系统业务^[4].

IT 系统结构关系可视化系统具有重要应用意义, 首先, 该系统构建了业务至 IT 设施的可视化管理模型, 为系统管理者提供了便利. 其次, 该系统将可视化技术应用于结构关系呈现领域, 对可视化应用具有研究价值. 最后对于该系统运用层次信息可视化技术直观的呈现系统结构, 提供故障影响分析, 故障物理定位, 监控系统业务, 并且具有可拓展性的优点, 对今后的系统结构关系分析应用具有重大意义^[4].

2 系统结构关系可视化系统介绍

2.1 可视化系统功能介绍

系统结构关系分析系统主要分为系统管理、配置管理、IT 应用可视化三大模块, 如图 1 所示.

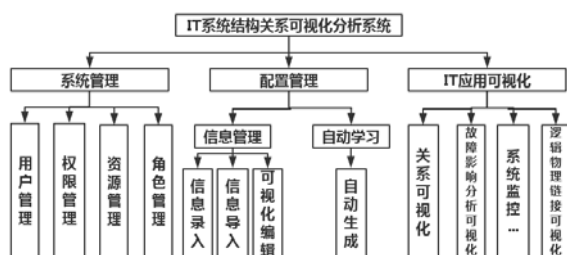


图 1 系统功能图

(1) 系统管理

系统管理模块负责管理系统用户、角色、资源的分配与管理; 用户分为系统管理员(root)和普通用户(游客)两类: 系统管理可以管理所有资源, 普通用户可以可视化系统资源, 但是不能作增删改操作, 对系统起到保护的作用.

(2) 配置管理

配置管理模块提供系统输入和配置功能, 在录入方面: 本系统采用智能方式, 利用相关机器学习领域

的算法自动识别系统结构并录入数据库, 但为了整合用户原有 IT 系统的目的, 也提供了手动录入, 文本导入, 图形编辑录入等常见功能. 配置方面: 本系统可以自由配置样式, 配置显示方案.

(3) IT 应用可视化

IT 应用可视化模块是本系统的核心模块, 它提供了一个利用三维可视化技术构建业务至 IT 设施的可视化管理模型. 它可以从逻辑到物理两方面呈现 IT 系统的运行支持结构, 可以监控业务动态, 支持故障影响分析, 支持故障回溯; 它清晰全面的为系统管理者提供了 IT 系统逻辑结构地图, 使得系统管理者可以实时监控业务, 全面了解系统, 并最终为决策作指导.

2.2 实施方案

2.2.1 系统数据交互过程

本系统采用 Struct2+Hibernate4+Spring3(SSH)框架搭建后台服务, 利用 Unity3D 引擎进行可视化展示, 以 Web 方式提供与用户的交互界面. 其交互过程如图 2 所示.

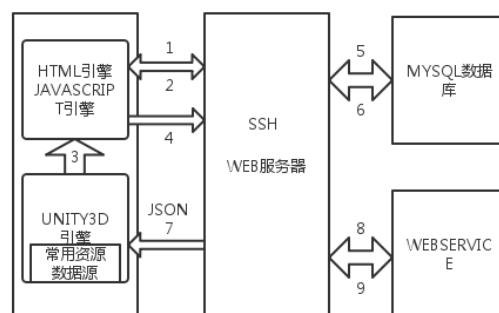


图 2 交互过程

图 2 中数字说明如下:

(1) 1-2

为一般的 Web 交互过程, 利用 Http 协议完成浏览器与服务器的交互. 主要功能为实现用户系统管理与配置管理模块的交互过程.

(2) 3

利用 Unity3D 提供的与 JavaScript 接口完成 IT 应用可视化模块的数据调用.

(3) 4

利用 Ajax 异步根据 Unity3D 所需请求服务器数据, 同时也是 Web 异步请求数据接口. 此处两个接口有点耦合, 但是实现简单.

(4) 5-6

利用 Hibernate4 实现与数据库交互. 此处主要交互一些相对固定的数据(例如节点要素名称, 负责人等).

(5) 7

封装服务器与 Unity3D 的 JavaScript 交互接口 (SendMessage()函数), 利用 Json 数据格式进行交互.

(6) 8-9

对于故障、监控信息等实时数据通过调用 Webservice 获取.

本系统将数据分为三类: 实时数据, 常更改数据, 不常更改数据. 实时数据利用实时调用 Webservice 获取, 常更改数据封装到 Unity3D 运行环境中, 不常更改数据存储在数据库中, 从而极大的提高了数据获取速度.

2.2.2 系统架构

本系统采用模型(model)+视图(view)+控制器(controller) (简称 MVC)模式, 具有简洁, 耦合度低的特点. 其系统架构如图3所示.

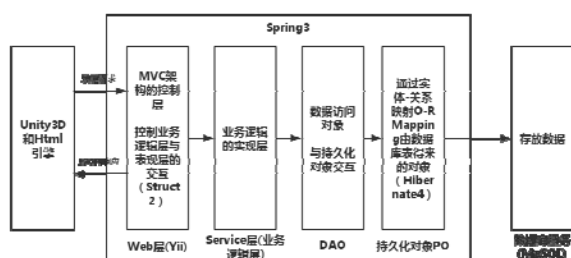


图3 系统架构图

本系统架构借助 MVC 模式的优点, 完成组件之间的解耦, 可以很好的进行拓展, 很贴切的面向服务进行开发.

3 系统结构关系可视化系统技术点说明

3.1 关键技术点

本系统使用 J2EE 开发, 中间整合 Unity3D 进行三维呈现. 整个系统的难点分为三部分:

- (1) Unity3D 与 JavaScript 之间数据交互接口;
- (2) 结构拓扑布局算法的设计和实现;
- (3) 布局节点动态调整功能的实现.

3.1.1 Unity3D 与 JavaScript 之间数据交互接口

(1) Unity3D 调用 JavaScript

Unity3D 引擎支持 JavaScript, 在引擎内部实现集成了调用 JavaScript 系统函数 Application.ExternalCall 来交互数据.

(2) JavaScript 调用 Unity3D

JavaScript 调用 Unity 的数据内容采用封装的 SendMessage()函数, 该函数封装了 Unity3D 的相关对象 UnityEngine.Object 调用, UnityEngine.Object 类提供了 getUnity() 函数提供支持.

通过 Unity3D 与 JavaScript 之间的数据交互接口, 使得 JavaScript 为 Unity3D 提供运行环境及数据提供功能, Unity3D 负责三维显示功能, 从而使得两部分功能明确, 耦合度降低, 利于系统维护.

3.1.2 结构拓扑布局算法的设计和实现

现有的布局算法主要包括直交布局算法、树型布局算法、射线型布局算法、层次性布局算法和混合型布局算法等^[5]. 本系统采用层次性布局算法, 并在原来算法的基础上进行了改进, 主要是: 原来层次化算法在节点量比较大, 并且每层分布不均匀时会出现头重脚轻的状况^[6], 所以本系统进行了螺旋形改造: 即先根据节点个数判断层级大小(用节点个数的开平方作为层级的边长); 当节点增加时, 将会一圈一圈螺旋增长, 从而有效的避免了原有层次性布局面对不均匀数据时出现的头重脚轻问题^[7].

具体代码实现如下:

```
JsonData jd=JsonMapper.ToObject(jsonString);  
//将 Json 字符串进行封装为模型对象  
for(Object: jd.Count){
```

```
GameObject Cube1 //创建层级板
```

```
Cube1.AddComponent<blurChilds>();//增加虚化功能
```

```
Cube1.AddComponent<样式>();//增加层级样式
```

```
for(Object: Cube1){
```

```
Point.transform.position=new
```

```
Vector3(Cube1.transform.position.x-Cube1.collider.boun  
ds.size.x*0.5f+(1.5f+(k-1))*3*Point.collider.bounds.size.
```

```
x,Cube1.transform.position.y,
```

```
Cube1.transform.position.z-Cube1.collider.bounds.size.z  
*0.5f+(1.5f+(j-1))*3*Point.collider.bounds.
```

```
size.x);} //利用布局算法设置位置
```

```
}
```

代码说明: 通过解析 Web 服务器提供的 Json 数据, 获取层次及节点信息, 并依此创建对象, 增加对象功

能,而节点对象的位置通过螺旋式增长方式获得.螺旋式增长方式即:代码中根据原有节点位置增加或减少一定间距的方式获得,环绕原有节点动态增长.

3.1.3 布局节点动态调整功能的实现

布局节点动态调整指当系统初始化完成后,针对用户个性化需求,要求进行定制化布局的一种实现方案.即在布局初始化后用户可以拖动节点位置,并且保存现有位置,使下次生成时会按用户要求生成.

实现伪代码如下:

```
lines[i]=new  
VectorLine("name",pos[i],LineMaterial,3f,  
LineType.Discrete, Joins.Weld);  
//用数组存下选中节点关系线的对象,但是线的位置是用 pos[i](位置数组)来存储的  
pos[i]=this.gameObject.transform.position;  
//根据鼠标位置实时更改位置数组的数值  
void Update () {  
    for(line: lines){  
        line.draw3D;  
    }  
}  
//每帧绘画关系
```

利用上述实现,可以使得所有与当前选中节点的关系线会随着选中节点的移动而移动,从而实现节点动态调整的功能.

4 可视化系统运行效果

IT 系统结构关系可视化系统的运行效果如图 4 所示.

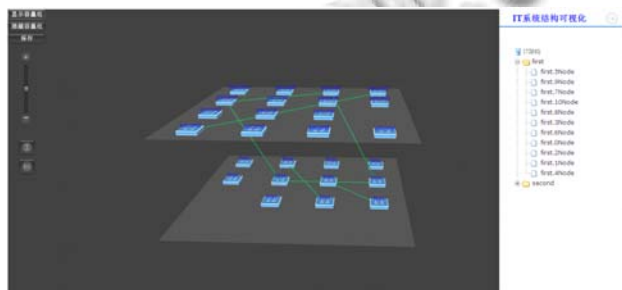


图 4 运行效果图

本系统利用三维的方式构建业务至 IT 设施的可视化管理模型,以层次信息可视化技术呈现业务、系统、

物理设备之间关系,展示了各要素之间的关联关系和所属关系,各层节点布局灵活、清晰,直观地展现了 IT 系统要素之间的层次关系以及应用的所属结构.并且实时监控各要素节点的业务运行状态,对业务节点提供检索和定位功能,支持各节点对其他节点的影响分析,从而为系统管理提供信息支持.本系统是可视化技术在系统管理方面的应用,对今后拓展可视化技术的应用价值有重要意义.

5 结语

本文论述了 IT 系统结构关系可视化系统的功能结构、系统架构、实施方案和关键技术.该可视化系统现针对 IT 系统开发,但本文讲述的技术路线不局限于 IT 系统,对其他系统的结构可视化依然有效^[8].该可视化系统构建业务至 IT 的可视化管理模型,对今后系统管理方面依然存在重大意义.

参考文献

- 1 张驰,罗铁坚,王相根.基于 Web 的信息可视化系统的设计与实现.计算机系统应用,2009,18(12):5-9.
- 2 布里顿.IT 体系结构与中间件.北京:人民邮电出版社,2003.
- 3 高屹.信息可视化技术研究分析.科技创新与应用,2016,(1):59-60.
- 4 洪文学,王金甲.可视化和可视化分析学.燕山大学学报,2010,34(2):95-99.
- 5 窦长威.层次信息可视化技术的一种实现方法.工程地质计算机应用,2007,(2):11-15.
- 6 刘金明,万明祥.基于节点度分层的路由器级拓扑布局算法.计算机技术与发展,2015,25(1):100-106.
- 7 杨如意,刘东苏.基于大众标注的层次信息可视化算法研究.现代图书情报技术,2014,(7):71-76.
- 8 任国栋,陈林华,陶学锋,等.基于 Unity3D 的虚拟博物馆信息可视化系统.计算机系统应用,2013,22(9):86-90.
- 9 Bohunsky P, Gatterbauer W. Visual structure-based web page clustering and retrieval. International Conference on World Wide Web, WWW 2010. Raleigh, North Carolina, USA. April, 2010. 1067-1068.
- 10 Kjellin A, Pettersson LW, Seipel S, et al. Evaluating 2D and 3D visualizations of spatiotemporal information. ACM Trans. on Applied Perception, 2010, 7(3): 483-496.