

智能知识图谱平台

1. 目录

智能知识图谱平台

- 1. 目录
- 2. 引言
 - 2.1 目的
 - 2.2 范围
 - 2.3 参考文献
- 3. 总体描述
 - 3.1 项目前景
 - 3.2 项目功能
 - 3.3 用户特征
 - 3.4 约束
 - 3.5 假设与依赖
- 4. 详细需求描述
 - 4.1 对外接口需求
 - 4.1.1 用户界面
 - 4.1.2 硬件接口
 - 4.1.3 软件接口
 - 4.1.4 通信接口
 - 4.2 功能需求
 - 4.2.1 用户上传CSV格式的知识图谱
 - 4.2.1.1 特征描述
 - 4.2.1.2 刺激响应序列
 - 4.2.1.3 相关功能需求
 - 4.2.2 用户查看知识图谱
 - 4.2.2.1 特征描述
 - 4.2.2.2 刺激/响应序列
 - 4.2.2.3 相关功能需求
 - 4.2.3 用户编辑知识图谱
 - 4.2.3.1 特征描述
 - 4.2.3.2 刺激/响应序列
 - 4.2.3.3 相关功能需求
 - 4.2.4 用户导出知识图谱为数据文件
 - 4.2.4.1 特征描述
 - 4.2.4.2 刺激/响应队列
 - 4.2.4.3 相关功能需求
 - 4.2.5 用户导出知识图谱为图片
 - 4.2.5.1 特征描述
 - 4.2.5.2 刺激/响应队列
 - 4.2.5.3 相关功能需求
 - 4.2.6 允许用户自定义图元属性
 - 4.2.6.1 特征描述
 - 4.2.6.2 刺激/响应队列
 - 4.2.6.3 相关功能需求
 - 4.2.7 自适应展示布局
 - 4.2.7.1 特征描述
 - 4.2.7.2 刺激/响应队列
 - 4.2.7.3 相关功能需求
 - 4.2.8 图元搜索
 - 4.2.8.1 特征描述
 - 4.2.8.2 刺激/响应队列

- 4.2.8.3 相关功能需求
- 4.2.9 数据持久化（含布局）
 - 4.2.9.1 特征描述
 - 4.2.9.2 刺激/响应队列
 - 4.2.9.3 相关功能需求
- 4.2.10 显示模式切换
 - 4.2.10.1 特征描述
 - 4.2.10.2 刺激/响应队列
 - 4.2.10.3 相关功能需求
- 4.2.11 展示效果调节
 - 4.2.11.1 特征描述
 - 4.2.11.2 刺激/响应队列
 - 4.2.11.3 相关功能需求
- 4.2.12 从用户提供的文本中构建人物图谱
 - 4.2.12.1 特征描述
 - 4.2.12.2 刺激/响应队列
 - 4.2.12.3 相关功能需求
- 4.2.13 根据用户手动修改改善自动人物图谱构建
 - 4.2.13.1 特征描述
 - 4.2.13.2 刺激/响应队列
 - 4.2.13.3 相关功能需求
- 4.3 非功能需求
 - 4.3.1 安全性
 - 4.3.2 性能
 - 4.3.3 可靠性
 - 4.3.4 易用性
 - 4.3.5 可移植性
 - 4.3.6 约束
- 4.4 数据需求
 - 4.4.1 数据定义
 - 4.4.2 默认数据
 - 4.4.3 数据格式要求
- 4.5 其他

2. 引言

本文档详细描述了 nioCoin 知识图谱可视化系统的需求。该文档为需求及软件系统开发的根据。

本文档仅面向开发者，不将终端用户作为目标读者。

2.1 目的

该系统用于对给定格式知识图谱的可视化，可以帮助人类或机器对大规模数据中包含的特征进行发现。

2.2 范围

该文档包含 nioCoin 项目的总体描述，系统的对外接口，功能性需求与非功能性需求。

2.3 参考文献

1. IEEE 需求规格文档标准
2. [软件开发的技术基础](#)
3. [软件工程与计算\[三\]——团队与软件开发实践](#)

3. 总体描述

3.1 项目前景

同时随着社会的发展，越来越多的数据对公众可见，如何利用好这些数据，分析并得出有意义的指导结论，帮助使用者的各类决策，成为一个不可忽视的问题。本项目立足于数据分析中常用的知识图谱技术，提供了简单易用的可视化接口，帮助用户完成实体和实体关系的抽取，预览，存储，提供高效的图片查询乃至补全推理，助力社会公共知识体系的建设。

3.2 项目功能

当前系统包括了数据的导入导出，数据上传，数据可视化，简易的实体/关系编辑功能。

3.3 用户特征

用户是无（或不普遍具备）计算机背景知识，但期望从数据中获取直观知识的人，系统设计应满足简单易用，封装不必要的复杂选项。

3.4 约束

CON1：系统使用 Java 开发。 **CON2**：系统使用迭代式开发。 **CON3**：系统应该基于 Web。 **CON4**：在每个迭代周期中，开发者要提交并维护

1. 计划文档
2. 软件需求规格说明文档
3. 设计描述文档
4. 测试报告

3.5 假设与依赖

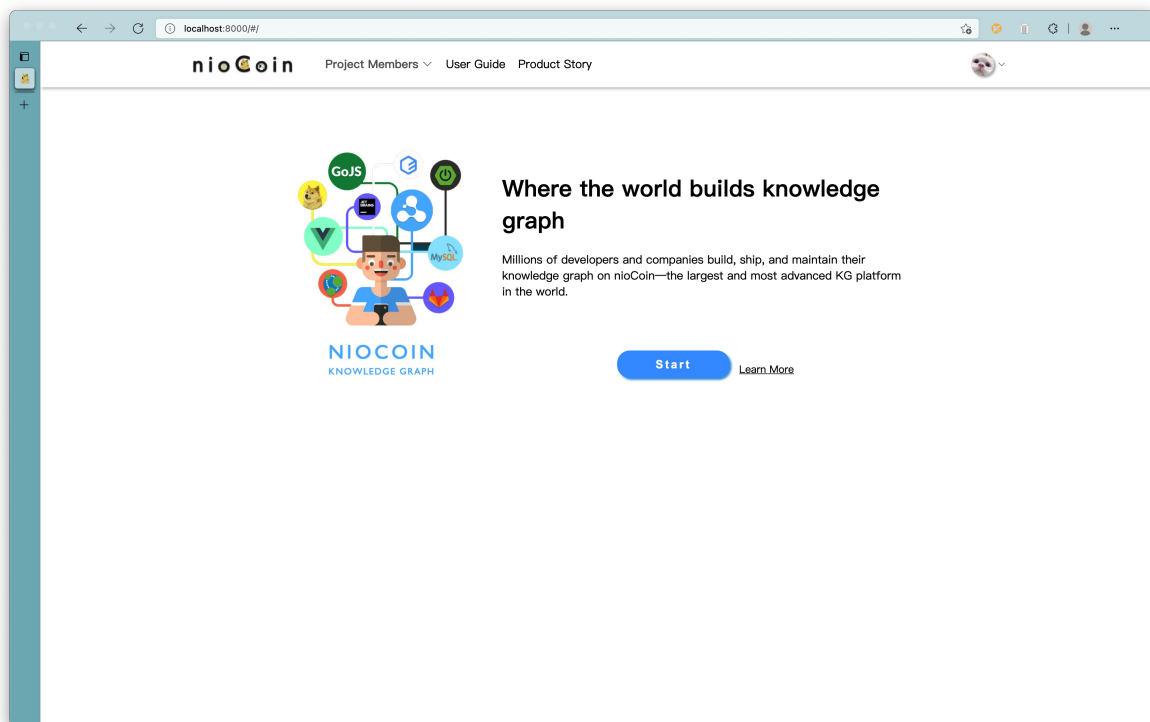
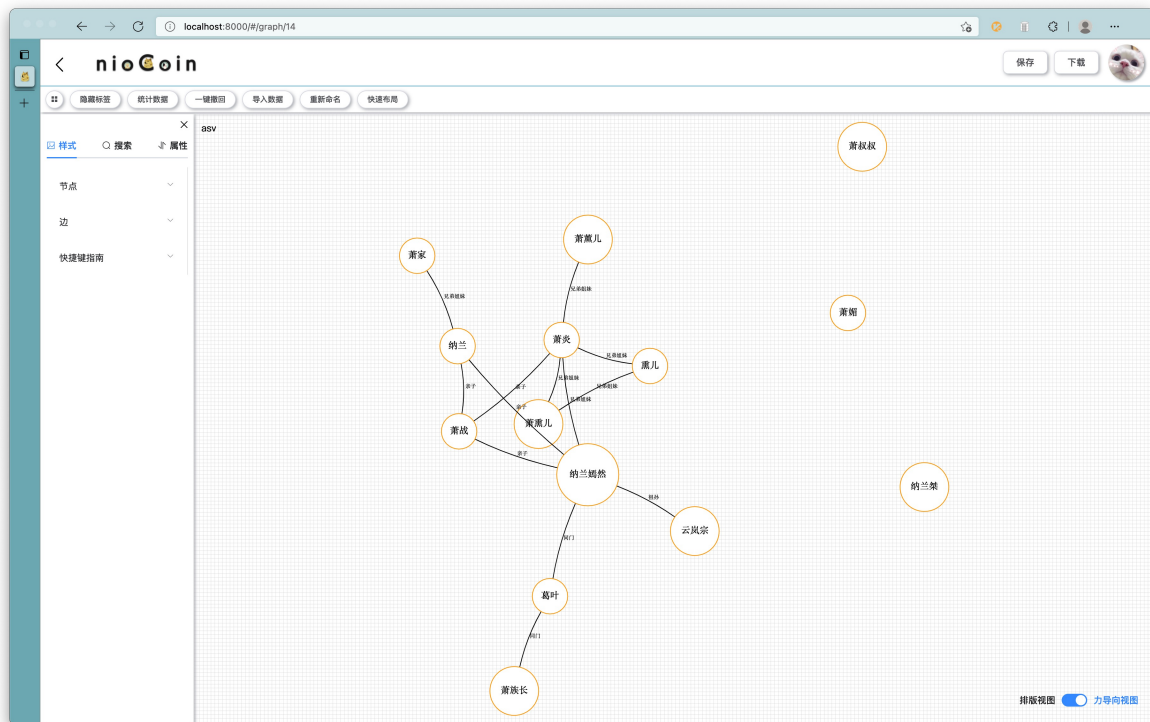
AS1：用户在流畅（ $BandWidth \geq 10M$ ）的网络环境下使用本系统。

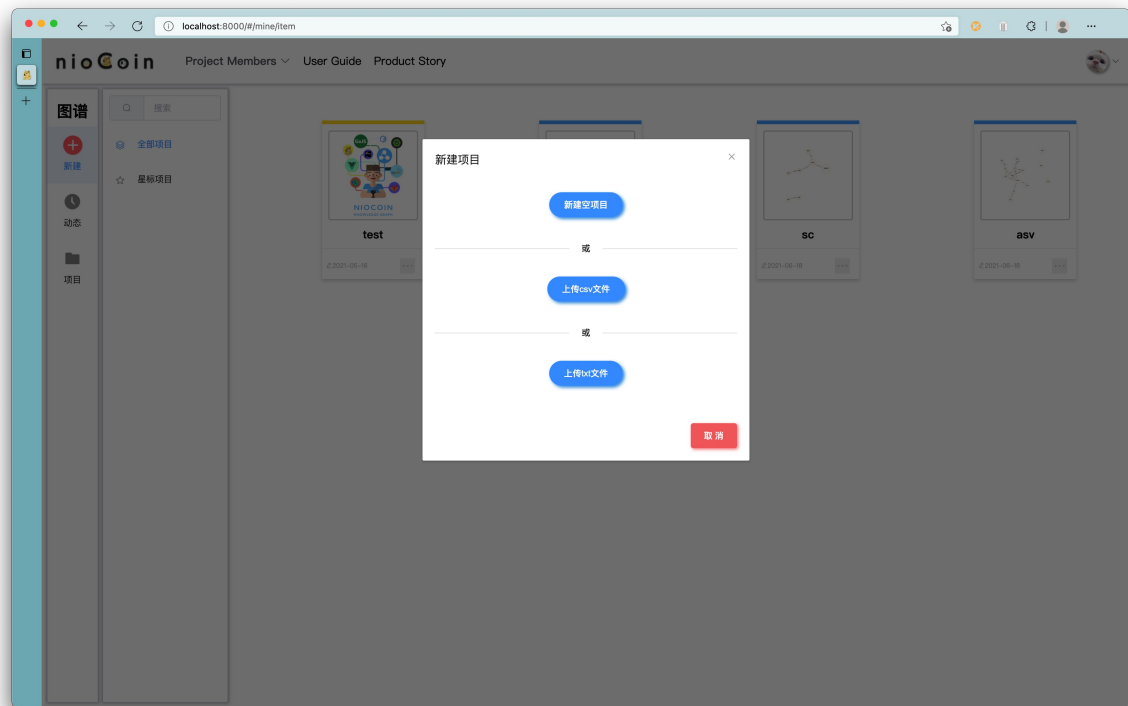
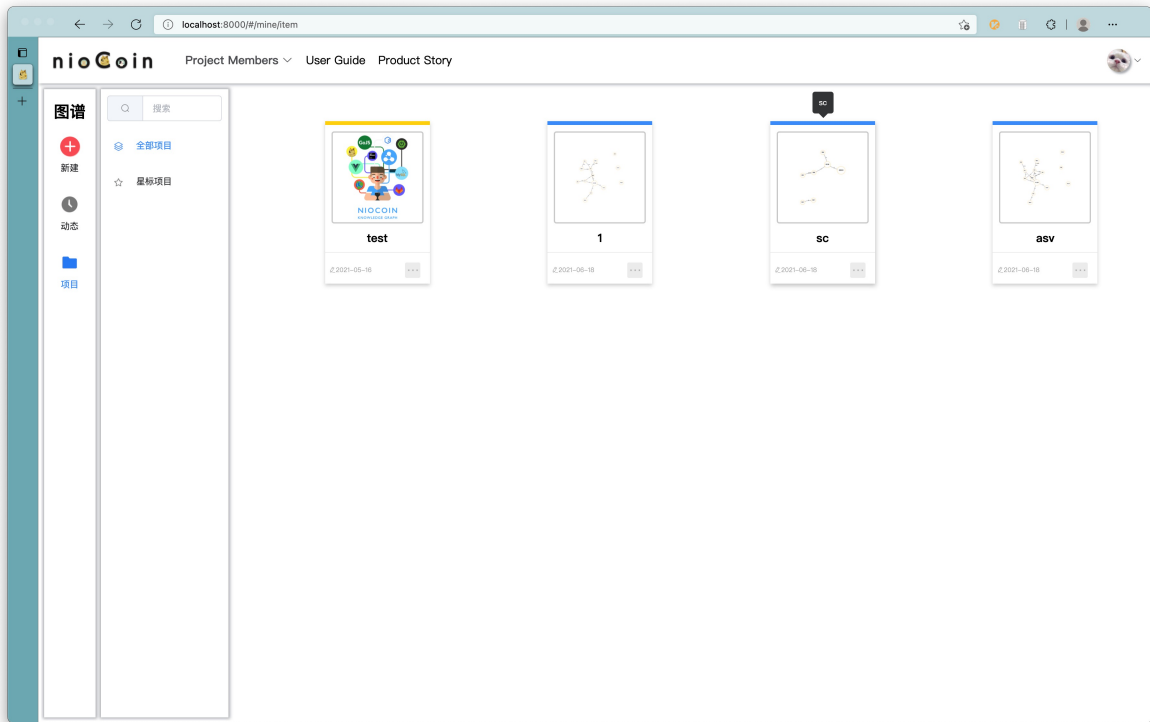
AS2：用户不具有过大规模（ $|Nodes| \geq 10000$ ）的可视化请求。

4. 详细需求描述

4.1 对外接口需求

4.1.1 用户界面







4.2.1.3 相关功能需求

编号	描述
kgbackend.convertJSON	将 CSV 文件转化为 go.js 期待的模型格式
kgbackend.graph.updateGraphName	更改图谱名称
kgbackend.graph.updateGraph	更新图谱内容

4.2.2 用户查看知识图谱

4.2.2.1 特征描述

用户上传知识图谱后，获取知识图谱所有的节点和关系信息

4.2.2.2 刺激/响应序列

刺激：用户选择上传的知识图谱

响应：系统返回用户可视化的知识图谱

刺激：用户移动光标到具体的节点或关系

响应：系统返回用户节点或关系的相关信息

4.2.2.3 相关功能需求

编号	描述
kgbackend.transJSON	将Neo4j数据库中存储的节点和关系信息返回

4.2.3 用户编辑知识图谱

4.2.3.1 特征描述

用户获取到知识图谱后，可以通过文本编辑实现对知识图谱的增删改

4.2.3.2 刺激/响应序列

刺激：用户选择编辑知识图谱

响应：系统返回可编辑的文本形式的知识图谱界面

刺激：用户选择增加节点或关系

响应：系统增加数据库中的节点或关系并更新

刺激：用户选择删除节点或关系

响应：系统删除数据库中的节点或关系并更新

刺激：用户选择修改节点或关系

响应：系统修改数据库中的节点或关系并更新

4.2.3.3 相关功能需求

编号	描述
kgbackend.modify	修改Neo4j数据库中的节点或关系状态
kgbackend.transJSON	将Neo4j数据库中存储的节点和关系信息返回

4.2.4 用户导出知识图谱为数据文件

4.2.4.1特征描述

用户可以在操作界面将知识图谱的内容导出为xml格式文件

4.2.4.2刺激/响应队列

刺激：用户选择界面操作

响应：系统展示可进行的操作

刺激：用户选择导出数据文件

响应：系统自行导出下载xml文件供用户本地查看

4.2.4.3相关功能需求

编号	描述
kgbackend.getGraphData	从neo4j请求图谱数据，以便前端进行导出

4.2.5 用户导出知识图谱为图片

4.2.5.1特征描述

用户可以在操作界面将知识图谱的内容导出为png格式图片

4.2.5.2刺激/响应队列

刺激：用户选择界面操作

响应：系统展示可进行的操作

刺激：用户选择导出图片

响应：系统自行导出下载png图片供用户本地查看

4.2.5.3相关功能需求

编号	描述
kgbackend.getGraphData	从neo4j请求图谱数据，以便前端进行导出

4.2.6 允许用户自定义图元属性

允许用户为图元设计非预设的属性

4.2.6.1 特征描述

刺激：用户选择所需编辑属性图元（点元或边元，该设置可作用于全局图元）

响应：系统展示表单收集目标属性（属性名，默认值）

刺激：用户填写表单

响应：系统更新受编辑图元属性并回显

4.2.6.2 刺激/响应队列

4.2.6.3 相关功能需求

编号	描述
kgbackend.setAttribute	更新当前图谱全局或特定图元的属性

4.2.7 自适应展示布局

4.2.7.1 特征描述

根据当前窗口尺寸与比例自适应调整展示布局

4.2.7.2 刺激/响应队列

刺激：用户变更窗口大小并达到临界值

响应：图谱布局方式变化

4.2.7.3 相关功能需求

编号	描述
kgbackend.reLayout	响应式调整展示布局

4.2.8 图元搜索

4.2.8.1 特征描述

根据用户规格描述搜索满足要求的图元

4.2.8.2 刺激/响应队列

刺激：用户展开搜索界面

响应：展示可选的搜索选项

刺激：用户填写选项并搜索

响应：高亮满足条件的图元

刺激（可选）：用户选择过滤不满足图元

响应：不满足图元暂时隐藏

4.2.8.3 相关功能需求

编号	描述
kgbackend.search	用户可以搜索具有指定特征的图元

4.2.9 数据持久化（含布局）

4.2.9.1 特征描述

用户可以保存图谱中的数据及其排版布局

4.2.9.2 刺激/响应队列

刺激：用户选择保存数据（本地或云端）

响应：若为本地，下载图片数据

响应：若为云端，更新用户图谱数据

刺激：用户打开图谱数据（本地或云端）

响应：若为本地，要求用户图片数据

刺激：选择本地文件

响应：若为云端，要求用户选择其拥有的图片

响应：显示被读取的数据

4.2.9.3 相关功能需求

编号	描述
kgbackend.persist	存储图谱的内容与布局信息
kgbackend.store	从持久化数据中恢复图谱的内容与布局信息

4.2.10 显示模式切换

4.2.10.1 特征描述

用户可选择可编辑的排版模式与自动力导图模式

4.2.10.2 刺激/响应队列

刺激：用户切换模式

响应：切换至另一显示模式

4.2.10.3 相关功能需求

编号	描述
kgbackend.layoutMode	允许用户切换排版模式与力导图模式

4.2.11 展示效果调节

用户可调整图谱展示选项，包括字体大小，图元形状。

4.2.11.1 特征描述

4.2.11.2 刺激/响应队列

刺激：用户选中待编辑的图元，选择展示设置

响应：弹出展示选项列表

刺激：用户设置展示选项

响应：该图元对应变化

4.2.11.3 相关功能需求

编号	描述
kgbackend.viewOption	允许用户传递展示选项定制图谱

4.2.12 从用户提供的文本中构建人物图谱

4.2.12.1 特征描述

允许用户上传 TXT 文件，根据文本内容分析人物及其关系并据此生成图谱

4.2.12.2 刺激/响应队列

刺激：用户选择从 TXT 文本导入

响应：系统要求用户选择文件

刺激：用户选择符合格式的文件

响应：系统开始解析部分文本获取初步结果

4.2.12.3 相关功能需求

编号	描述
kgbackend.python.ltp	使用神经网络解析文件识别实体
kgbackend.python.nre	使用神经网络解析文件抽取关系
kgbackend.graph.updateGraph	更新图谱内容

4.2.13 根据用户手动修改改善自动人物图谱构建

4.2.13.1 特征描述

用户基于自己的认知对初步解析结果进行编辑，系统学习并构建出更准确的知识图谱。

4.2.13.2 刺激/响应队列

刺激：用户在初步解析的结果内操作（节点的修改，增加与删除，关系的删除，修改）

响应：系统记录用户修改，等待下一个操作

刺激：用户提交修改

响应：系统根据存储的操作历史，对完整文本进行解析，并构建知识图谱

4.2.13.3 相关功能需求

编号	描述
kgbackend.python.ltp	使用神经网络解析文件识别实体
kgbackend.python.nre	使用神经网络解析文件抽取关系
kgbckend.python.filter	根据用户操作历史过滤三元组
kgbackend.graph.updateGraph	更新图谱内容

4.3 非功能需求

4.3.1 安全性

Safety1：系统对用户应该只能开放节点的名称编辑

Safety2：系统对用户只能开放联系的名称的编辑

Safety3：系统应该只对有权限的用户开放

4.3.2 性能

Performance1：系统获取知识图谱的时间不能超过500ms

Performance2：系统导入知识图谱的响应时间不能超过3s

Performance3：系统的超时错误率不能超过0.1%

4.3.3 可靠性

Reliability1：如果在客户交互时，网络出现故障，系统不能出现故障

Reliability2：数据库的数据实时备份，在丢失或者破损之后可以自动回复

4.3.4 易用性

Usability1：无需用户使用手册或者专业培训即可直接使用系统

Usability2：用户编辑知识图谱或是导入知识图谱不会超过三个模态框

4.3.5 可移植性

Portability1：系统或部件可以部署到Windows以及Linux服务器中

Portability2：数据库可以在Windows，Linux，MacOS中进行多平台移植

4.3.6 约束

C1：系统采用分层模型开发

C2：系统使用neo4j数据库存储数据

C3：系统使用Java的Springboot+Vue3进行开发

4.4 数据需求

4.4.1 数据定义

DR1：系统需要存储用户的基本注册信息及登录信息

DR2：系统需要存储知识图谱的基本信息，包括节点信息、边信息以及图谱标识符

DR3：系统需要存储用户导入的图谱数据

DR4：系统需要存储知识图谱的布局信息和视图位置详细信息

DR5：系统需要存储用户对于图谱进行编辑后的图谱信息，包括增删改查

DR6：系统需要存储用户在一段时间内的操作和编辑记录

DR7：系统删除的所有数据需要继续存储一段时间，以保证历史记录完整性

4.4.2 默认数据

在没有数据的时候默认在数据库存入字符串 null

4.4.3 数据格式要求

Format1：用户注册和登录信息数据统一使用json进行传输

Format2：用户导入的图谱数据统一用csv格式文件进行导入，用csvString格式数据进行传输

Format3：节点和关系的编辑数据统一使用json进行传输

Format4：图谱的布局 and 位置数据精确到整数

Format5：用户操作编辑记录用json格式进行存储

Format6：系统中的时间精确到天，时间格式为yyyy-mm-dd，用户编辑的时间精确到分，格式为mm-dd-hh-mm

Format7：SpringBoot 后台与 Flask 后台通信时，使用 "<==>" 分割参数，首参数为需要调用的命令

Format8：节点相关属性在后台通过 "<==>" 平均后存储在单字符串中

4.5 其他

系统安装部署的时候需要同时导入一些数据，保证系统在安装部署之后用户可以直接进行一定的试用体验。