目录

[第1 章 绪论 7](#_Toc3560740)

[1.1 国内外研究现状 7](#_Toc3560741)

[1.1.1核电数据库管理系统的发展及现状 7](#_Toc3560742)

[1.1.2国内外对于项目管理系统的探索 8](#_Toc3560743)

[1.1.3核电数据库管理系统的重要意义 10](#_Toc3560744)

[1.2 课题背景—TMSR熔盐泵实验平台数据库管理的背景 11](#_Toc3560745)

[1.3主要研究内容及组织架构 13](#_Toc3560746)

[第二章 关键技术 14](#_Toc3560747)

[2.1系统设计方法 14](#_Toc3560748)

[2.2 开发体系结构的选择 16](#_Toc3560749)

[2.3 数据库的选择 18](#_Toc3560750)

[2.4系统开发平台 22](#_Toc3560751)

[2.4.1 Node.js 22](#_Toc3560752)

[2.5 本章小结 23](#_Toc3560753)

[第三章 系统总体设计 24](#_Toc3560754)

[3.1 系统开发目标 24](#_Toc3560755)

[3.2 管理系统需求分析 24](#_Toc3560756)

[3.2.1 系统业务需求 24](#_Toc3560757)

[3.2.2 系统功能需求分析 25](#_Toc3560758)

[3.1.3 系统非功能性需求 28](#_Toc3560759)

[3.3 系统设计 30](#_Toc3560760)

[3.3.1 总体架构设计 31](#_Toc3560761)

[3.3.2 功能设计 33](#_Toc3560762)

[3.4.1需求分析 33](#_Toc3560763)

[3.4.2 数据结构 35](#_Toc3560764)

[3.5 视图结构设计 37](#_Toc3560765)

[3.5.1页面结构及功能 37](#_Toc3560766)

[3.6 本章小结 40](#_Toc3560767)

[第四章 熔盐泵设计管理平台的实现 40](#_Toc3560768)

[4.1 系统开发环境及配置 40](#_Toc3560769)

[4.1.1 硬件环境 40](#_Toc3560770)

[4.1.2 软件环境 40](#_Toc3560771)

[4.2系统开发架构 41](#_Toc3560772)

[4.3 关键技术实现 41](#_Toc3560773)

[4.4 关键模块实现 44](#_Toc3560774)

[4.4.1系统管理模块 45](#_Toc3560775)

[4.4.2设备静态信息管理模块 46](#_Toc3560776)

[4.4.3文档信息管理模块 47](#_Toc3560777)

[4.4.4 设备动态信息管理模块 49](#_Toc3560778)

[4.5 本章小结 51](#_Toc3560779)

[第五章 系统测试 51](#_Toc3560780)

[5.1 测试环境 51](#_Toc3560781)

[5.2 系统功能测试 51](#_Toc3560782)

[5.3 本章小结 51](#_Toc3560783)

[第六章 总结与展望 51](#_Toc3560784)

[6.1 总结 52](#_Toc3560785)

[6.2 特色与创新 52](#_Toc3560786)

[6.3 工作展望 54](#_Toc3560787)

摘 要

Abstract

# 第1 章 绪论

## 1.1 国内外研究现状

### 1.1.1核电数据库管理系统的发展及现状

上世纪九十年代末，在八五期间，从数据中心的筹建到建库计划的指定，从管理系统的选型到各种标准规范的制定，从“八五”计划的落实到研制合同的验收，实现了核电工程数据库从无到起步这样零的突破。在这期间，进行了组织分工，讨论出了建库方针、路线和方法；建立了核电工程数据库总体结构框架；制订出了若干标准规范和规章制度；初步建成了涉及核电工程设计、项目管理、运行、材料、经济等方面的七个示范性数据库子系统，完成了国家“八五”攻关项目《核电工程数据库研究》专题（专题编号85-213-01-14）[1]。其中初步建成的七个示范性数据库子系统是：核电运行数据库（包括国内外核事件定级、国外核电厂异常事件、国外核电运行可靠性数据库的通用数据库和秦山核电厂运行异常事件库）、核电工程材料数据库（包括反应堆及反应堆主冷却剂系统设备材料、电器及控制系统材料、焊接材料和燃料组件等子库）、核电项目管理核库（包括计划投资管理、文件收发管理、设备采购、进度网络管理等数据库）和300MW核电设计数据库（包括核电站参数子系统、设备子系统）、核电工程经济数据库（包括核电厂经济、核燃料循环经济、核电规划与环境经济数据库）、核电工程文献数据库（包括国内外压水堆核电厂选址、审批、设计、调试、运行、维修、事故等方面的文章）。

近年来，为保障核电厂的有效安全运行，保证设备的可靠性，通过分析设计可靠性保证大纲（D-RAP）的流程和主要步骤，设计了管理D-RAP过程中文件、数据、信息的D-RAP数据库。该数据库能够辅助运行经验（OPEX）报告的定量化评估，并将评估过程中产生的数据和文件储存在数据库中；考虑了概率安全分析（PSA）相关软件接口，为PSA分析以及电厂运行的风险监测提供数据输入；将D-RAP设备相关的信息、文件、数据、报告等内容进行整合关联，实现方便查询，并能整合打包作为O-RAP的输入[2]。

### 1.1.2国内外对于项目管理系统的探索

国外目前广泛用于大型工程设计的商业软件有英国AVEVA公司的PDMS和美国Intergraph公司的Smart Plant Foundation。其中PDMS是英国AVEVA公司自1977年生产了第一个自主设计的PDMS的商业软件，目前最新的版本为PDMS 12.1.其具备的特征是：1）具体参照实际所见而建立的三维空间设计下的建模；2）在互联网的帮助下，完整体现实际的环境，各个不同的专业组共同协调统筹建成附带具体数据的工厂模型。而设计人员通过实时的模型都可以随时了解自身的工作；3）在整个人机交互的设计程序中，PDMS利用立体的三维碰撞检测的功能实现PDMS的全方位设计的终极目标；4）因为在PDMS中采用独立的数据库结构，因此元件或者设备的数据存储在各自的元件库和设备库中，规避了因大型数据库信息处理的限制；5）软件支持多种编程语言，实现多种数据和程序间的良好连接。PDMS软件逐步的发展下，出现了不断的进步已经从最初的三维设计工具逐步发展成最新的功能包含设计、管理、施工的建筑平台数据软件。

Intergraph公司早些年研发设计了新的Smart Plant Enterprise软件，该软件的研究设计是为了更好掌控公司的业主、工程以及运营商设计等信息的信息管理软件。SPF信息管理软件为工厂企业提供全面监控工厂数据的储存厂库，对企业的数据进行全面覆盖，从工厂的设计、施工以及修整以及革新等不同的阶段进行实时的信息监控，有助于管理者对工厂进行信息系统管理。Smart Plant Foundation是一个对工程信息进行储存、处理、整理汇总的一个信息处理平台。Smart Plant Foundation的功能涵盖了工程设计软件的信息集成框架及工作流程管理，使用Smart Plant Foundation可以更好、更快地执行工程总承包项目（EPC）。

国内，近年来也有多家高等院校、科研设计院所、工程设计单位在工作流程信息化管理、设计工作标准化、信息化、自动化方面做了许多的探索和改进，开发了可以满足于某一特定功能的信息、数据、工作管理平台，如北京广利核公司开发的数字化仪控系统设计配置管理工具，通过对商用工具的扩展以及再次开发，建立了对仪控的专用配置平台。将软件工程方法论与配置管理的相关技术有效结合在一起，在核电项目管理领域取得了较大的成果。该系统首先不再采用人工管理模式，减少了开发所需要的时间，以及解决了沟通方式不畅、版本控制不规范等问题，同时也使该软件在应用时可以更加高效且人性化，较大程度地满足了实际工作的需要。

针对工程设计管理系统方面的探索，在勘察设计领域，基于“项目设计计划管理、项目设计输入管理”等基础环节需要，采用B/S结构模式，JAVA相关的开发工具、J2EE等技术，加上对管理系统的数据库设计，设计开发出的满足功能需求的设计管理系统。

除此之外，基于工作流和MOSS开发的工程综合管理系统、教学管理系统项目审批管理系统等也在一些高校、工程管理单位等得到广泛应用。

### 1.1.3核电数据库管理系统的重要意义

核电工程数据中心：是核电工程数据库建设的组织协调管理机构，建库工作周期长、任务重，协调管理的内容十分复杂，有必要对数据库建设过程中的规划、计划、开发合同、文件、文档、标准、数据库等内容进行系统保存、管理。

核电工程数据库规划和开发管理系统的研究目的：是为核电工程数据中心提供一套对入库的文档和数据库、生效合同、经费预决算、会议组织和合同管理规程、计划和规划、标准规范和规章制度、文件、大事记等内容进行动态管理和报表处理的数据库系统。研制并应用该系统的意义在于：对核电工程数据库开发组织工作进行规范化、系统化、科学化、现代化管理，为调整开发方案、制定开发规划提供依据，在一定程度上辅助数据中心决策、管理。同时便于核电工程数据库开发的组织、协调管理工作的自然承接，利于吸取教训、扬长避短，提高工作效率和管理水平。

## 1.2 课题背景—TMSR熔盐泵实验平台数据库管理的背景

中国科学院与2011年1月启动首批五项战略性先导专项，其中钍基熔盐堆（Thorium-based Molten-Salt Reactor，TMSR）核能系统专项依托于上海应用物理研究所实施，旨在通过20多年的时间研发出第四代反应堆核能系统。TMSR于2012年研制了小流量高温熔盐泵，即熔盐泵原理样机。在304不锈钢预样机的研制中，开展了水介质试验，热态空气介质的干运转试验。由于水与熔盐介质的物性不同，熔盐泵和水泵在结构和密封方式上也不同；常温与高温的环境工况，对于泵的运行情况也有很大不同。熔盐泵的成功研制需要大量的运行试验验证和工程验证。因此，熔盐泵测试平台的建设是在熔盐泵的研制过程中必不可少的阶段。熔盐泵熔盐试验回路目的是为熔盐泵及样机提供高温熔盐试验平台，主要任务是对泵进行熔盐介质下的高温试验研究。实验内容主要包括：型式试验和运行试验，为主循环泵及样机提供定型依据。依据鉴定大纲，进行鉴定试验。数据库是熔盐泵试验测试平台的必不可少的组成部分，是利用熔盐泵台架进行型式试验和鉴定试验的重要保障。 数据库管理系统是一种对数据库进行操作和管理的软件，其用途包括对数据库进行查询、修改和删除等操作，简称DBMS。根据数据库管理系统可以实现统一管理和操作数据库，以最大限度保证数据的完整性。随着B/S结构的高速发展，基于WEB开发的数据库应用系统越来越普及，它能够实现不同的操作人员，从不同的地点，以不同的接入方式访问和操作共同的数据库，同时能够有效地保护服务器数据平台。

随着核电工程项目的增多，设计输入和接口数据成倍增加，各项目的设计数据和要求各有不同，设计过程也受供贷商以及电厂现场变化影响大，各种修改量极大。虽然目前已经实现计算机化办公，但所使用的软件功能有限，不能很好地与目前核电厂系统设计工作结合，设计人员在应用时需要做大量的非技术性基础工作，造成时间浪费，花费增多。因此，采用信息化技术，设计更先进的设计管理平台，提升工作效率，降低人因失误概率，是目前最行之有效的解决方法。在此背景下，本论文选择研究了设计管理平台，分析了该平台的功能需求和详细设计，并实现了该系统。

课题研究的意义是可以提高设计部门的设计和管理的效率，提高设计工作的标准化和信息化程度，实现合理利用人力资源呢，降低人因失误，保证设计质量和任务进度，具有较大的应用意义和价值。整个设计管理平台集成了人员、技术、设计、协同工作等多项支持功能，主要特点为“信息化、标准化和自动化”。信息化主要体现在：在设计周期内的所有数据，包括设计输入、参考数据、责任人员、待定项目、设计输出、进度管理要求等各种信息集成到设计管理平台中，实现信息资源共享利用，确保任何设计人员可以在平台中快速知道自己的工作任务以及完成这些设计任务已经获得的设计输入、未解决的问题等综合信息。标准化主要体现在：将设计活动中的输入、过程、产出进行规范，利用结构化的模型进行管理，统一形式、统一认识和统一平台，确保高效有序地工作。自动化主要体现在：将设计活动中自动追踪并记录文件、数据的修改历史，实现设计过程可控。

## 1.3主要研究内容及组织架构

本论文调研了国内外现今核电厂数据库管理平台的发展和现状，切实分析了TMSR熔盐泵实验平台数据库管理系统的实际需求，调研了合适的数据库系统及开源技术，详细研究内容如下：

1. 调研和分析国内外核电数据库管理平台技术的发展历史、现状和趋势，阐述了TMSR熔盐泵实验平台数据库管理系统的研究背景，论述了基于TMSR熔盐泵实验平台的数据库管理系统的意义所在。
2. 研究分析了架构、数据库技术、前端框架、数据可视化技术等关键技术，包括关键技术的选择。
3. 结合第二代钍基熔盐堆核能项目实施的实际背景，对设计管理平台进行需求分析和功能设计，构建系统架构。

（4）开发设计TMSR熔盐泵实验平台数据库管理系统，实现各功能模块，最终模拟应用在钍基熔盐堆仿真验证装置上，通过设计文件的计划清单管理，开口项管理和接口管理等活动，证明了该系统数据模型设计的可行性。

（5）系统测试。

本文的组织架构分为七章，各章的主要内容如下：

第一章介绍国内外核电数据库技术的发展及项目管理系统的发展趋势，立足于本课题研究对象TMSR项目中的熔盐泵实验平台的研究，明确开发熔盐泵实验平台数据库管理系统的意义和目的所在。

# 第二章 关键技术

## 2.1系统设计方法

软件工程是一项从概念到具体，从设计到实现的过程。在软件工程中，将涉及到很多相关技术。比如软件的设计技术、软件的研发技术。本章将研究熔盐泵设计管理平台在实现时，使用的关键技术，包括系统设计方法、系统实现方法和数据库技术等，

软件工程领域，常见的软件开发方法有面向过程的开发方法，面向数据结构的开发方法和面向对象的开发方法。其中面向对象开发方法的思想是对问题空间进行自然分割，以更接近人类思维的方式建立问题域模型，以便对客观实体进行结构模拟和行为模拟，从而使设计出的的软件尽可能直接地描述现实世界，构造出模块话的，可重用的，维护性好的软件，同时限定软件的复杂性和降低开发维护费用。它是通过面向对象的分析（OOA）、面向对象的设计（OOD）和面向对象的程序设计（OOP）等过程，将现实世界的问题空间平滑地过渡到软件空间的一种软件开发过程。目前，典型的面向对象的开发方法是UML和统一开发过程（RUP）。本文使用具有包括统一标准、面向对象、可视化及表达能力强大、独立于过程和容易使用掌握特点的UML设计方法（Unified Modeling Language）。

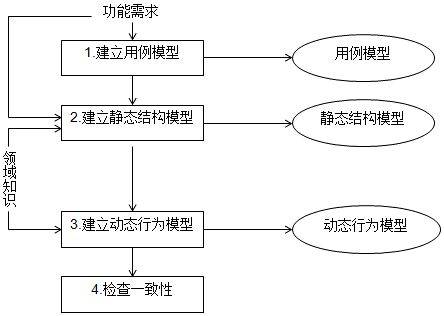


图 UML建模方法流程

优秀的UML建模可以为将来的系统开发节省大量的时间和精力。UML建模语言具有一整套完备的系统设计方案，涵盖了系统需求分析、业务流程分析、系统功能设计、系统类图设计等。在UML的实际使用中并不一定需要实现其全部的设计方案，单独使用UML的某一种设计方法也很常见。UML建模方法虽然流程较多，但从本质上就是建立系统的功能用例模型，系统的静态结构模型和系统的行为模型。上图展示了此种建模流程，通过建立这些模型，基本描述出系统由谁使用，系统能完成什么事情，完成这些事情需要哪些数据，完成这些事情需要经过哪些流程。

用例模型：用例模型描述了系统的用户是谁，他们使用系统完成什么功能，以及完成这些功能的基本步骤。建立用例模型时。首先需要站在系统外部考虑系统对外需要提供哪些功能，这时需要建立顶层的用例模型，然后将系统内部的功能进行细分，确定每个功能的使用者和其涵盖的子用例，并详细介绍每个用例的实现过程，这时需要建立各功能模块的用例模型。UML中经常使用例图的方式来完成用例模型的建立过程。

静态结构模型：静态结构模型描述了系统中所包括的各种实体的静态结构，通常使用类图的方式来表示。不同静态结构之间的关系有：关联、依赖和聚集。通过对用例的分析，可以得到不同静态结构之间信息交互的内容和方式。使用类图建立的静态结构可以大大减少系统开发时的工作量。

动态行为模型：动态行为模型描述了系统各个功能或业务流程在系统程序中的实现过程、数据流转方式等。UML中定义了顺序图、状态图、活动图、序列图等描述方式，不同的图的侧重点不同，例如顺序图主要描述业务的运行过程，而时序图重点描述系统实现时各个实体之间数据流转的时间顺序。

建立完三种模型以后，需要检查三种模型的一致性。用例模型、静态结构模型和动态行为模型需要一脉相承。一致性检查通过后，UML建模完成。

## 2.2 开发体系结构的选择

目前信息系统的开发体系结构主要包括C/S和B/S两种。每种方式都有自己的优缺点和应用场景。本章旨在根据系统的特点，选择合适的开发体系结构。

在软件开发领域中，常见的有B/S和C/S架构两种开发模式。其中B/S中的B指的是Brower，S指的是Server，也就是说它是浏览器和服务器组成的系统。B/S的兴起和互联网技术分不开，在B/S结构出现之前，C/S是最常用的开发框架体系。这种开发方式对于客户端的软硬件配置有一定的要求，需要在客户端系统中安装一定的软件运行环境，因此需要一定的硬件配置水平。每个客户机都需要有运算】展示等功能，这种方式在安全性方面有一些优势，但是C/S结构缺乏交互性，稳定性较差，而且系统的维护工作量巨大。B/S与C/S最大的不同在于，它不用再客户本地安装软件程序，客户端只要可以连接互联网，拥有浏览器，就可以使用浏览器访问系统，将数据请求通过互联网传送给服务器，服务器经过运算处理后，通过互联网将信息传递给客户端，客户端只负责信息的展示。这种方式大大减轻了客户端的负担，降低了对客户机的要求，系统稳定性增强。但是在响应速度方面，需要依靠互联网的带宽和质量，

B/S的主要优点有开发简单、后期维护方便，因此开发效率高，开发周期短，可以说B/S的重点在于服务器端，对于客户端的的要求非常低，只需要拥有浏览器，B/S以其特有结构降低了客户端电脑负荷，又有效地保护了数据库的安全，减少了资源的浪费。基于B/S软件结构的这些优点，现在大至门户网站，小至企业管理软件，都在广泛地使用B/S。因此本系统采用B/S结构来开发实现熔盐泵设计管理平台。

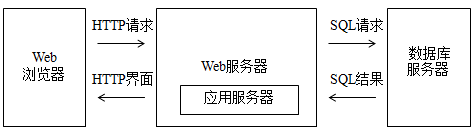


图 B/S的三层架构体系

B/S开发结构也被称为Web开发技术，国内将这种开发结构成为三层开发体系。如上图所示，基于B/S的软件系统可以分为三部分：Web浏览器、Web服务器和数据库服务器。信息在B/S结构中的传递方式如下：首先客户使用Web浏览器向Web服务器发送Http请求，Web服务器接收请求，并根据业务逻辑向数据库服务器发送SQL请求，数据库接收到请求后，查询数据库并将查询结果返回给Web服务器。Web服务器通过业务逻辑，将查询结果展示给Web浏览器。B/S可以处理多样式的媒体结构，比如声音、视频、图片、文件等。

因此本文将基于B/S结构来开发熔盐泵设计管理平台。

## 2.3 数据库的选择

SQL是英文Structured Query Language的缩写，也就是结构化查询语言。SQL语言的主要功能就是要与数据库之间保持必要的关联，并进行相应的沟通工作。按照ANSI的相关规定，SQL为关系型数据库中的标准语言。SQL语句可以用于执行各类操作，如对数据库中的数据进行更新，以及提取数据库中的数据等。

自1970年以来关系数据库便开始出现，经过了40多年来的演化，如今的关系型数据库具备了强大的存储、维护、查询数据的能力。关系型数据库，是指采用了关系模型来组织数据的数据库，常见的关系型数据库有Oracle,Mysql,sql server,PostgreSql等。关系型数据库的优势在于其易于理解的二维表结构，是非常贴近逻辑世界的概念，关系模型相对网状层次等其他模型来说更容易理解；通用的SQL语言使得操作关系型数据库非常方便；丰富的完整性（实体完整性、参照完整性和用户定义完整性）大大降低了数据冗余和数据不一致的概率。NoSQL非关系型数据库是2009年由Eric Evans在一场关于开源分布式数据库的讨论中提出的，NoSQL用于指代那些非关系型的、分布式的且一般不保证遵循ACID（Atomic原子性，Consistency一致性，隔离性，Durability持久性）原则的数据存储系统。非关系型数据库以键值对存储且结构不固定，每一个元组可以有不一样的字段，每个元组可以根据需要增加一些自己的键值对，这样就不会局限于固定的二结构。但由于非关系型数据库具有很少的约束，它不能够提供像SQL所提供的where这种对于字段属性值情况的查询，并且难以体现设计的完整性，它只适合存储一些较为简单的数据，对于需要进行较复杂的数据，SQL数据库更为合适，基于以上原因，本系统采用关系型数据库。

目前主流的关系型数据库主要包括Oracle,Mysql,sql server,PostgreSql，下面一一比较优劣：

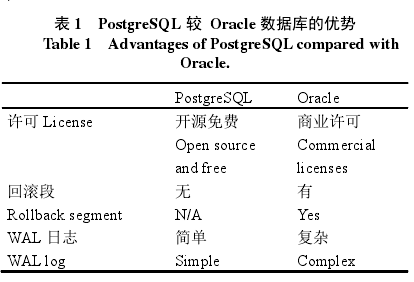
1）Oracle Database，又名Oracle RDBMS，或简称Oracle，是目前世界上流行的关系数据库管理系统，系统可移植性好、使用方便、功能强，适用于各类大、中、小、微机环境。它是一种高效率、可靠性好的 适应高吞吐量的数据库解决方案。但对硬件的要求很高，价格比较昂贵，管理维护麻烦一些，操作比较复杂，需要技术含量较高。

2）SQL Server 是Microsoft 公司推出的关系型数据库管理系统。具有使用[方便](http://baike.baidu.com/view/142800.htm)可伸缩性好与相关[软件](http://baike.baidu.com/view/37.htm)集成[程度](http://baike.baidu.com/view/644590.htm)高等[优点](http://baike.baidu.com/view/1465058.htm)，可跨越从[运行](http://baike.baidu.com/view/1026025.htm)Microsoft Windows 98 的膝上型[电脑](http://baike.baidu.com/view/2358.htm)到运行Microsoft Windows 2012 的大型多处理器的服务器等多种平台使用。缺点是只能windows上运行，没有丝毫开放性操作系统，系统稳定对数据库十分重要，WindowsX系列产品偏重于桌面应用，NT server只适合小型企业，而且windows平台可靠性、安全性和伸缩性非常有限。

3）MySQL 是一种关联数据库管理系统， 关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。其社区版的性能卓越，搭配 PHP 和 Apache 可组成良好的开发环境。缺点在于不支持热备份。MySQL不支持自定义数据类型 MySQL最大的缺点是其安全系统，主要是复杂而非标准，另外只有到调用mysqladmin来重读用户权限时才发生改变。MySQL对存储过程和触发器支持不够良好。MySQL对XML支持不够良好

而PostgreSQL是一个自由的对象-关系数据库服务器(数据库管理系统),支持大部分 SQL标准并且提供了许多其他现代特性：复杂查询、外键、触发器、视图、事务完整性、MVCC。同样，PostgreSQL 可以用许多方法扩展，比如， 通过增加新的数据类型、函数、操作符、聚集函数、索引。免费使用、修改、和分发 PostgreSQL。 PostgreSQL 不只是一个关系型数据库，还是一个面向对象数据库——支持嵌套，及一些其他功能。虽然PostgreSQL也具有其劣势，如对于简单而繁重的读取操作, 超过了 PostgreSQL 的杀伤力，可能会出现比同行（如MySQL）更低的性能。但这点劣势与其强大的优点相比可以忍受。

PostgreSql作为一种复杂的对象-关系型数据库管理系统，也是目前宫娥能最强大，特性最丰富和最复杂的自由软件数据库系统。【】PostgreSql数据库完全免费，遵循BSD（Berkeley Software Distribution）版权协议，在这个版权的范围内，允许任何人几乎可以做任何事情，包括增强和商业化，不会被其他公司控制。



PostgreSQL较Oracle的优势在于：1）Oracle通过CPU收费，它完全免费，且遵循BSD协议，不受任何公司控制。2）PostgreSQL没有回滚段，对于Oracle来说一旦回滚段损坏，会导致数据丢失，甚至数据库无法启动的严重问题。而没有回滚段，将旧数据记录在原先的文件中这样的方式使得PostgreSQL出现异常的几率更小，另外它的WAL日志比Oracle要更简单。

鉴于以上优势，本系统选择PostgreSQL数据库来开发熔盐泵设计管理平台。

## 2.4系统开发平台

### 2.4.1 Node.js

Node.js 是一个基于 Chrome V8 引擎建立的服务器端 Java Script 运行环境 ，可以用来快速搭建具有高响应速度和易于扩展的网络应用。Node.js 具有异步 I/O, 事件驱动、单线程等特性。其基于事件轮询的异步响应方式 ，使得 Node.js 在响应文件资源读取、数据库查询等请求时 ，无需等待硬盘 I/O 完成 ，在数据准备好后会通过事件机制通知 Node.js进行响应 ，这种编程模型在处理小文件并发请求时可同时满足高并发和高响应的要求。

2.4.2 AngularJs和BootStrap前端框架

MVC是一种UI架构模式，它的目的是将应用的功能分解到专门的模块，实现模块的重用性，减少模块间的耦合度，增强系统的鲁棒性。MVC模式主要分成三部分： Model: 用来存储系统数据 ；View: 用来实现系统的UI界面 ；Controller: 用于衔接Model和View。AngularJS是一个MVC框架，依赖于AngularJS的框架机制，实现三者（控制器Controller,界面view和数据Model）间的相互联动。 这里的框架指得是是一种特殊的、已经实现了的WEB应用，你只需要对它填充具体的业务逻辑。Angular JS (Angular.JS) 是一组用来开发 Web 页面的框架、模板以及数据绑定和丰富 UI 组件。它支持整个开发进程，提供 Web 应用的架构，无需进行手工 DOM 操作。AngularJS有着诸多特性，最为核心的是：MVC（Model–view–controller）、模块化、自动化双向数据绑定、语义化标签、依赖注入等等。

BootStrap属于前端UI库，可以快速搭建前端页面，还可以使用saas重新设计组件，具有跨设备，跨浏览器可以兼容所有现代浏览器的优势。bootstrap提供了一套响应式、移动设备优先的流式栅格系统。它可以根据用户屏幕尺寸调整页面，使其在各个尺寸上都表现良好。bootstrap预先定义了很多CSS类，使用的时候直接给class赋予对应的类名即可，如text-left，text-align，.table等。Bootstrap提供了很多实用性的JQuery插件，这些插件方便开发者实现WEB中各种常规特效。bootstrap提供了实用性很强的组件，包括：导航，标签，工具条，按钮等供开发者使用。综合来看是一项功能强大且极具优势的前端UI组件库。

## 2.5 本章小结

本章研究并学习了熔盐泵设计管理平台在开发时需要的关键技术，首先介绍了系统的设计方法，选择了面向对象领域常用的UML建模方法，然后选择了开发体系结构，重点比较分析了C/S和B/S两种开发架构的不同，以及本文采用B/S架构的原因。接着介绍了数据库开发技术，先比较了关系型数据库和非关系型数据库的使用场景，接着比较了主流的几种关系型数据库，最终本文选用Postgresql数据库管理系统。最后介绍了Web开发技术包括Node.js开发平台，AngularJs前端开发框架以及Bootstrap UI库，利用以上技术开发本系统应用。

# 第三章 系统总体设计

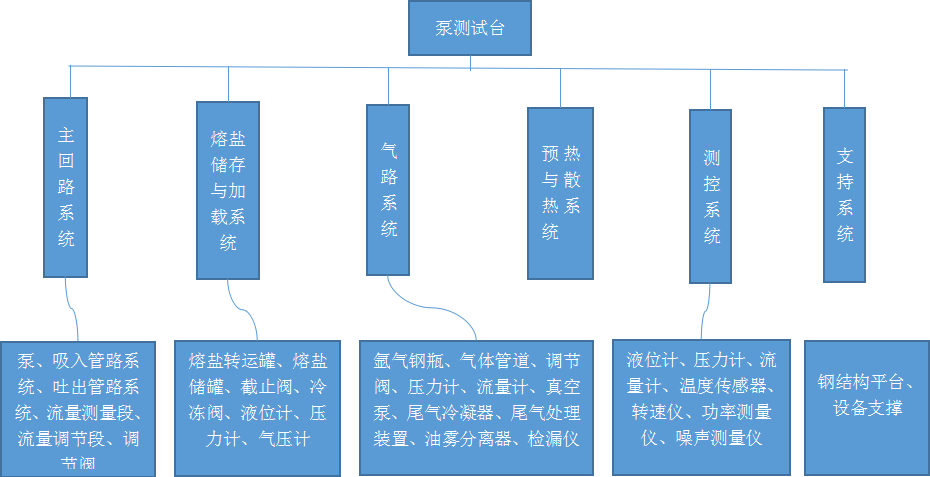
## 3.1 系统开发目标

本系统的目的是建立一套使用TMSR熔盐泵实验平台的管理平台，用计算机化、网络化的方式来完成整个熔盐泵实验平台在设计、建设、运行过程中产生的各类数据的存储和管理，其中包括完成实验平台建设阶段设备数量、基础信息、设备设计参数等信息的录入与查询；设备运行阶段通过EPICS IOC采集系统采集到的动态数据的归档数据的查询、图形绘制、下载。另外尤其是对实验平台整个方案预研、工程设计、施工设计、工程建造、工程验收等全周期内产生的项目文档，以项目计划管理、流程控制、质量监督为核心，以多项目、多级的管理体系以及计划管理、过程控制和质量监督三维一体的管理理，集成了人员、技术、设计、协同工作等多项功能，项目管理中的计划跟踪、设计输出、输出审核等工作，一方面实现项目管理软件和数据库管理平台的集成，另一方面实现核电设计项目管理、进度的管理。

## 3.2 管理系统需求分析

需求分析的主要任务是熟悉系统的业务，明确用户的需求，重点是调查和分析用户在信息管理中需要处理什么样的数据和如何处理数据两大问题。通过实际的需求分析得到，熔盐泵实验平台在研究、设计、建造、运行整个周期里需要处理的数据包括三大类：1）研究和设计阶段产生的项目文档数据，具体包括工程设计文件、工程联系文件、技术通知文件和工程制造文件等。2）建造阶段中涉及的设备的安装信息、设备的技术参数等。3）运行阶段，设备的一些控制信号，例如液位计读数等动态数据，这些动态数据通过EPICS IOC采集系统采集到，需要通过数据库引擎将实时数据的存档数据归档到关系型数据库PostgreSql数据库中。

熔盐泵实验平台数据库的设计就是围绕这三大类数据进行。对于设备安装信息、设备技术参数数据，需要实现的处理要求包括对设备安装信息、设备参数信息的保存、统计、在线查询，按照反应堆命名规则对这些数据命名，可视化展示出设备信息的树状结构以及系统间数据的联系。对于设备动态信息，需要实现的处理要求是对于动态数据的存档数据进行自定义查询、按照时间绘制图表、图表的下载等功能。对于项目文档数据，需要实现的处理要求包括项目文档中各类文件的保存、统计、在线查询；工程设计文件、工程联系文件、技术通知文件的上传和回复；工程设计文件以文件提交、校验、审核、会签、批准状态进行流程追踪。~~下图为设备信息按照系统划分的树形结构。~~

~~~~

~~下面按照这三大类数据具体说明。~~

### ~~3.2.1 系统业务需求~~

~~通过对TMSR熔盐泵实验平台业务需求的分析，本系统所实现的业务需求包括如下：~~

1. ~~设备管理，旨在将建造阶段产生的设备设计参数数据、设备安装信息等静态信息的保存、统计、在线查询。按照反应堆命名规则对这些数据命名，进行数据库的设计，在数据库中体现设备信息的树状结构以及系统间数据的关系。~~
2. ~~文档管理，研究和设计阶段产生的项目文档的保存、统计、在线查询；工程设计文件、工程联系文件、技术通知文件的上传和回复；以文件提交、校验、审核、会签、批准状态对工程设计文件进行流程追踪。以帮助设计人员更好得进行系统的设计。~~
3. ~~动态信息管理，设备运行阶段产生的运行数据，这些可以通过控制组获得，对这些运行存档数据合理建立数据库进行存储，进而进行上层应用的开发，如在web页面上实现检索和生成报表的功能。~~
4. ~~系统管理，系统管理是维护系统正常运行的基础。主要包括用户的管理、角色的配置、角色权限的管理等。~~

### 3.2.1 系统功能需求分析

通过上一章节的业务分析，本系统的主要功能包括设备管理、文档管理、动态信息管理、系统管理等，主要用户包括系统管理员、设计人员、项目管理人员、系统维护人员等。下面将分析详细的需求。

（1）设备管理

在熔盐泵实验平台的建造过程中，设备管理还未完全实施设备化管理，设备的信息查询还无法做到在web端随时随地得操作查询，尚未实现对于设备基础信息的保存、设备信息的修改、设备购置及报废后数量的更新，同时也缺少部门之间对于这些信息的有效沟通和共享，缺乏对设备的有效统筹和管理。因此设备管理包括了以下功能：

1. 设备基础信息例如设备编号、名称、所属系统、数量、设备参数等的录入；
2. 设备基础信息按照系统查询，设备参数信息按照系统、按照设备查询显示；
3. 设备基础信息、参数信息的增加、删除、更新。

（2）文档管理

文档管理管理的文档范围是熔盐泵实验平台设计的过程中，产生的一系列相关的文档，包括概念设计文件、方案设计文件、施工图设计文件。由于熔盐泵的设计过程是由多人协作完成的，设计过程产生多个设计文档，这些文档如何被多人处理，如何同步文档数据是文档管理所应重点关注的。文档管理功能包括：

1.建立设计计划管理、设计过程控制、质量监度三维一体的设计管理体系；

2.实现用户权限管理、时间进度管理，提供权限和时间纬度的支撑；

3.建立高效的、快捷的文档搜索功能，运用了数据库索引、缓存技术等；

4.实现多维度的分类、汇总、快速查询等功能；

5.实现设计文件版本控制，保证设计数据的安全性和协同性。

下面具体介绍流程追踪功能：TMSR熔盐泵设计过程需要多人共同协作完成，而且设计要经过若干个流程。每个人负责不同流程的不同部分，当其完成自己所负责部分流程后，只有相关人员审核通过后，才能继续到下一流程。因此流程管理控制仪控设计的全过程，对于熔盐泵实验平台设计的质量水平和时间进度起到关键的作用。设计过程控制包括：设计输入清单的编制、校核、审批。其中设计输入还包括不确定项的编制、校验和审批以及设计输出的编制、校核、会签、审批和质量审查管理等。设计输出则包括开口项的管理维护。

（3）动态信息管理

熔盐泵实验平台在运行过程中，为了保障设备的安全和正常运行，设备上将安装一些控制设备产生控制信号，这些控制信号产生的运行数据，将通过控制系统保存成存档数据，对这些存档数据进行数据库的开发，进而实现上层应用的开发，实现它们的管理功能。动态信息管理实现的功能包括：

1. 通过数据库引擎从EPICS动态数据数据库中采集到的设备运行数据归档到关系型数据库postgresql数据中；
2. 实现动态数据的历史存档数据按照时间查询；
3. 实现动态数据的历史存档数据绘制曲线图；
4. 实现按照功能自定义对动态历史存档数据的下载。

（4）系统管理

系统管理是维护系统正常运行的基础。主要包括用户的管理、角色的配置、角色权限的管理。系统管理功能包括：

* + 1. 用户管理：增加、删除一个用户、更新用户基本信息。
    2. 角色配置：对某一个用户赋予一定的权限，如设计人员、项目管理人员、系统维护人员等。

### 3.1.3 系统非功能性需求

熔盐泵设计平台除了在功能上要满足以上的需求以外，还应该在性能上有一些要求。

（1）先进性

系统在开发和设计时应重点考虑目前应用广泛的先进技术，既包括系统的设计技术，又包括系统开发技术，还包括系统实施部署技术。先进的技术有利于减少开发者的工作量，提高开发效率，并且在未来系统升级改进时更加便利。

（2）可靠性

系统应保持可靠。可靠不仅是系统可以随时完成用户需求，而且系统数据的存储要可靠，在出现系统崩溃等情况时，可以及时备份系统数据，防止数据丢失。

（3）可扩展性

扩展性也是系统的性能需求要求。良好的扩展性可以避免短时间内的重复开发，节省硬件和软件资源。从硬件的可扩展性来说，系统应使用兼容性良好的硬件设备，提供充足的硬件扩展端口，至少保证系统在10年内可顺利的更新换代。从软件的可扩展性来说，要预留系统功能扩展的功能接口和数据接口，当系统功能需要完善时，可方便调用相关的功能接口。这对于系统的持续演进具有重要的意义。

（1）系统的并发访问性能要求

本系统是单位内部的管理系统，用户的量级在百人上下。因此系统应保障至少几百人同时访问系统顺畅，不会出现系统瘫痪等情况。

（2）系统的安全特性包括了物理安全和数据安全。

物理安全要求机房在防火、防盗、防水、防断电等方面具有充足的应对措施，在出现问题时可以快速响应，及时解决。系统数数据安全要求建立数据的备份机制，除定期自动备份还要求按周手工备份。除本地备份外，还要求异地备份。在系统出现瘫痪、崩溃等情况时，可以及时恢复系统数据，降低系统损失。

## 3.3 系统架构设计

熔盐泵系统设计管理平台在设计中遵循的原则主要有：

统一规划，分步建设的原则：系统在设计时应与单位其他相关系统统一规划，考虑在数据接口、功能扩展方面的特性。系统建设时可以采取分步建设，逐步完善的方法。

可扩展性：可扩展性决定了系统的使用周期。为防止重复开发，造成软件和硬件的浪费，系统应在数据、功能、架构三个层次考虑系统的扩展性。

经济性：由于本项目为单位自建，在研发经费方面虽不充足，但有一定的保证。系统在设计时，应尽量选用质优价廉的技术，在硬件设备的设计上以够用、好用、易用为原则，严禁出现硬件性能过高造成不必要的浪费。

先进性：系统在开发和设计时应重点考虑目前应用广泛的先进技术，既包括系统的设计技术，又包括系统开发技术，还包括系统实施部署技术。先进的技术有利于减少开发者的工作量，提高开发效率，而且在未来系统升级改进时更加便利。

可靠性原则：系统应保证稳定、可靠。可靠不仅是系统可以随时完成用户需求，而且系统数据的存储要可靠，在出现系统崩溃等情况时，可以及时备份系统数据，防止数据丢失。

标准性和开放性：系统设计时应采用标准化的系统设计方法，方法应保障一定的开放性。

个性化：系统在设计时针对管理人员、普通用户等应设计不同的登录界面和主页，在系统的 UI 设计时应充分考虑不同用户在操作、浏览、功能方面的特点，尽量做到系统界面简洁、易用，又能满足用户的个性化需求。

### 3.3.1 总体架构设计

软件架构

系统采用的是Node.js、PostgreSQL、Bootstrap、ECharts技术，在Node.js的基础上，使用Express Web框架，构建整个应用。应用中设置路由解析，访问相应页面时，路由控制器处理请求，通过逻辑应用层对PostgreSQL数据库操作，然后通过Html模板引擎渲染好，返回给客户端[8]。

在Internet环境下，系统采用B/S结构，用户通过浏览器即可访问。整体架构如下：



**图2 系统架构**

如图2所示，该核能设计管理系统的体系结构分成四层：人机接口层，逻辑应用层，数据服务层，基础设施层。具体如下：

(1)人机接口层 （前端页面） 即Web页面，系统中综合使用Html模板，CSS3，Bootstrap模板布局等，采用ECharts数据可视化组件及前端Angular JS框架实现[9]。

(2)逻辑应用层（后端Web服务器）系统中采用的Node.js基础框架，其中的Express框架提供的路由模块作为控制器，用于实现业务逻辑，负责从视图读取数据，控制用户输入，并向模型发送数据[10]。

(3)数据服务层（关系型数据库）系统根据用户需求，综合比较了多种主流关系型数据库以及非关系型数据库，最终采用PostgreSQL作为核能文档数据的数据库管理软件，通过需求分析、实体分析、ERD图设计、数据库表设计、形成表间关系图一系列流程，完成设计管理系统的数据库的设计。

(4)基础设施层 包括硬件设施部分，如服务器、网络资源等。

整体采用MVC架构，（a）Module数据模型层，负责在以核能设计文件为数据建立的PostgreSQL数据库中存取文件数据，文件数据包括pdf格式的文档,jpeg,png格式的图片等。（b）View视图层，负责在Web网页上处理数据显示的部分，采用CSS3，Bootstrap模板布局，ECharts组件库实现图表的可视化展示。（c）Controller控制层，负责接收客户端用户的输入并调用模型和视图去完成用户的需求，其中发生了从视图读取数据，控制用户输入，并向模型发送数据一系列过程，Node.js负责控制层，将数据逻辑放在Node层进行处理。

### 3.3.2 功能架构设计

3.4 数据仓库设计

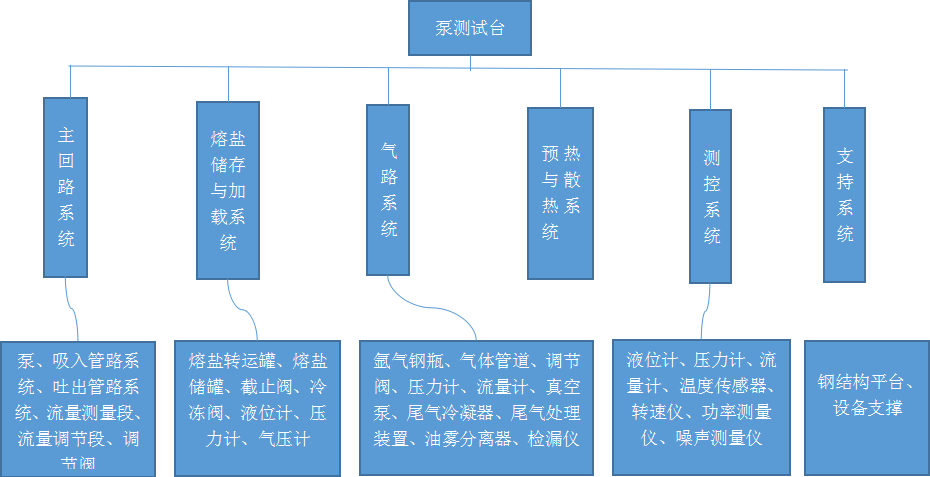
大多数的应用系统都属于数据库应用程序，都离不开数据库的支持。数据库设计过程是针对具体的应用环境，设计优化的逻辑模式，并根据所采用的数据库系统设计物理结构，最后建立应用程序的过程。数据库设计方案的优劣对于应用程序的运行至关重要。数据库的设计过程可以理解为提出问题、分析问题、解决问题的过程，具体包括四个步骤，需求分析、概念结构设计、逻辑结构设计、物理结构设计。

### 3.4.1数据结构分析

通过实际的需求分析得到，熔盐泵实验平台在研究、设计、建造、运行整个周期里需要处理的数据包括三大类：1）研究和设计阶段产生的项目文档数据，具体包括工程设计文件、工程联系文件、技术通知文件和工程制造文件等。2）建造阶段中涉及的设备的安装信息、设备的技术参数等。3）运行阶段产生的设备控制参数，例如液位计读数等动态数据，这些动态数据的存档数据可以从项目组下的控制组获得。下面按照这三大类数据具体说明。

### 3.4.2 数据结构

（1）设备静态信息的数据结构



熔盐泵实验平台按照系统功能可以划分为六个系统，分别是作为吸入管道系统连接泵进口起整流作用的主回路系统、将熔盐储存到熔盐罐中并加热成液态熔盐输送至主回路的熔盐储存与加载系统；运行前为回路清扫运行中为实验泵供气停机后对回路进行清扫的气路系统；为仪控系统提供参数检测设备运行状态对温度、流量、压力等参入进行过程控制的测量与控制系统；起到支持作用消除管道在高温下的热应力的支持系统。

（2）设备动态信息的数据结构

（3）文档信息的数据结构

数据库的逻辑结构任务是根据概念结构的E-R关系图转换为数据模型的过程，即在关系数据库中，将所有的实体和相应的联系转换为一系列的关系模型[12]。TMSR设计管理模型中，列表信息数据的关系模型如下：

**图3 TMSR设计平台的数据模型**

**Fig.3 The data model of the design platform of TMSR**

①信息统计项目概况（包括计划进度信息、开口项信息、会签文件数信息、质量监督信息、责任人信息）

②员工信息（包括身份ID、员工姓名、机构名称、课题名称、专业序号、细分专业序号、电话邮箱）③项目信息（包括项目ID、项目名称、责任人）④质量管理员（用于权限管理，包含身份ID、项目ID）

⑤文件清单(包括文件编号、文件名称、版本、部所、文件进度、设计阶段、实际出所时间、出院计划、编制人、校核人、审核人、审定人、质保等级）

⑥输入不定项（包括不定项编号、参考文件编号、参考文件名称、版本、不稳定描述、反馈人、反馈时间、反馈不说、责任部所、项目编号、状态）以及输出开口项（包括开口项的编号、原因、描述，文件图纸编号、版本，反馈人，反馈时间，项目编号，部所，管理状态等）

⑦技术协调（包含编号、事项描述、归属项目、归属部门、提出部门/专业、提出人、提出时间、审批状态、期待解决时间、实际解决时间）

⑧会议纪要（包含编号、会议议题、主控部门、会议地点、会议日期、会议主持、会议议题数量、后续行动项数量、纪要起草人、纪要审核人、纪要批准人、审批状态）

⑨质量监督（包括编号、信息类型、监督形式、外部文件号、主题、相关责任方、进度、发出日期、要求回复日期、满足要求回复日期、计划完成时间、实际完成时间、质量评估人）。

在规范化理论的基础上对上述关系模式进行分析，得出上述关系模式中没有多余的函数依赖，是规范程度很高的关系模型。该模型中除第一项统计信息外,其它都对应一个列表，列表中每行对应一个文件，所以整个项目元数据信息归结为：“类”列表数据和“对象” 文件数据的管理。

## 3.5 视图结构设计

设计管理系统主要分为两部分：（1）内容管理：对无流程的文档进行信息整理、列表统计、排序、模糊搜索等（2）流程控制：对有流程的文档进行流程追踪、版本控制、状态更新等。其中流程控制是文档管理的重要功能，包括对文件的提交、校验、审核、会签、批准等各种操作的控制，全面记录和跟踪文档出现的变更，对中间修改过程的文件版本进行保存管理，使得文档更加健全。

### 3.5.1页面结构及功能

****

页面结构分为四个层级，最外层是登陆页面，登陆验证通过后进入一级页面，主要包括项目概览、设计管理、质量监督、会议纪要、个人中心五个侧边导航栏；导航栏下包含八个二级页面，分别是统计信息详情、设计管理文件清单、输入不定项、输出开口项、技术协调、设计文件质量记录、会议纪要列表、员工信息页面。具体页面功能介绍：

（1）统计计划总体进度（总计划、已完成、进度完成比）、开口项（包括准开口项数、开口项、本月已关闭）、会签数（包括待会签数、本月完成数）、质量监督（包括打开数、关闭数、关闭率、责任人）等总体信息；

（2）列表展示（文件编号、名称、版本、部所、设计阶段、实际出所时间、出院计划、编制人、审核人、质保等级等）；

（3）列表展示、检索、排序输入不确定项；

（4）列表展示、检索、排序输出开口项

（5）列表展示、检索、排序技术协调事项；

（6）展示和统计包括文件监督形式、主题、相关责任方、进度、发出日期、请求回复日期、计划完成时间、质量评估人等设计文件质量记录；

（7）展示会议议题、部门、地点、日期、主持、议题数目、行动项数目、审核人、批准人、审核状态等会议纪要列表；

（8）展示员工信息包括身份、机构、可以、专业、电话、邮箱等。

三级页面包括文件清单页面下的新增文件页面和追踪文件流程的文件时间轴页面；输入不确定项和输出开口项页面下的项目详情页面，技术协调事项下的详情页面，质量文件记录下的质量分析页面，会议纪要列表下的新增会议记录页面。四级页面主要是文件时间轴页面下的变更页面，技术协调详情页面的事项变更页面，包含提交和回复文件的操作，以及质量分析页面下的绩效考核。综合以上页面，构成实现内容管理和流程控制的设计文件管理系统。



**图6 设计文件清单管理的操作**

**Fig.6 Design document inventory management operations**

图6为设计文件的操作页面，页面功能有：设计文件清单列表、状态显示；设计文件版本时间轴的展示和修改；元数据的提交、修改；文件的上传、下载和预览。实现了设计元素据以“点”上传，在时间轴线上管理数据，最终促进系统设计的“面”。

## 3.6 本章小结

本章完成熔盐泵实验平台的数据库管理系统的系统设计工作。先介绍目前系统的现状和基本功能需求，分析系统用户及各自的功能需求，系统主要功能概括为系统管理、设备管理、文档管理、动态信息管理等。从先进性、可靠性、可扩展性、安全性等角度描述了本系统的非功能性需求。其次进行系统的架构设计，总体采用MVC（Module-View-Controller）三层架构

# 第四章 熔盐泵设计管理平台的实现

## 4.1 系统开发环境及配置

### 4.1.1 硬件环境

1) 计算机：系统开发所用计算机配置为Dell i3-2350M CPU，4G内存，64位操作系统 ，1G 独立显卡，集成网卡。）

2) 服务器：（例如IIS6.5 及以上。）

### 4.1.2 软件环境

软件环境配置条件：

1) 编程语言：JavaScript、Html、CSS；

2) 开发平台：Node.js、Microsoft Visual Studio 2008；

3) 数据库：PostgreSql；

4) 操作系统：win 7 及以上；

5) 浏览器：IE8、谷歌浏览器。

## 4.2系统开发架构



## 4.3 关键技术实现

数据库接口：

为了与数据库建立连接，基于Node.js的 PostgreSQL 接口软件包，开发了PostgreSQL数据库访问接口文件pgconn.js：

var pg = require('pg');

var config = {

user: 'blog\_user', //env var: PGUSER

database: 'tmsr\_doc', //env var: PGDATABASE

password: '\*\*\*\*\*\*', //env var: PGPASSWORD

host: 'XXX.XXX.XXX.XXX', // Server hosting the postgres database

port: 5432, //env var: PGPORT

max: 10, // max number of clients in the pool

idleTimeoutMillis: 30000, // how long a client is allowed to remain idle before being closed

};

var pool = new pg.pool(config);

var query=function(sql,callback){

pool.connect(function(err, client, done) {

if(err) {return console.error('error fetching client from pool', err);}

client.query(sql, function(err, result) {

done();

if(err) {return console.error('error running query', err);}

callback(result);});});

};

exports.query = query;

前端先根据“ 文件名称”向后端发送数据请求，并把获取的数据赋给Angular全局变量$scope.request：

var title = $(document).attr("title");

var myUrl2 = 'http://web\_url/data/lf\_contact\_req/' + title + '?callback=JSON\_CALLBACK';

$http.jsonp(myUrl2).success(function (response) {

$scope.request=response;

});

后端服务器监听到数据查询请求，触发数据库para\_requirement表查询，并把查询结果以JavaScript 对象简谱（JavaScript Object Notation，JSON）格式给前端：

var pg1 = require('./pgconn');

router.get('/lf\_contact\_req/:post\_name', function (req, res, next) {

sql = 'select \* from param\_requirement where name= \'' + req.params.post\_name + '\';';

console.log(sql);

pg1.query(sql, function (result) {

console.log(result.rows);

res.jsonp(result.rows);

});

});

综合运用Angular数据绑定技术、jsonp跨域请求服务和ECharts可视化组件，实现了对文件信息的获取和可视化。数据的获取过程围绕Angular控制器及其依赖项展开。在前端页面创建Angular控制器，并注入两个依赖项：用于数据获取的$http对象和用于数据绑定的$scope对象。为了规避浏览器的同源策略对跨域访问数据的限制，借助$http对象的jsonp方法向后端服务器发起jsonp请求；服务器根据该请求向数据库获取数据，并以JSON格式将数据返回给$scope对象的指定变量，实现数据与变量的绑定。同时，$scope对象的变量与页面标签或Echarts组件绑定，完成数据从模型到页面视图的传递。例如前端页面Angular数据$scope.request的绑定，代码如下:

<ul class="timeline">

<li ng-repeat="x in request">

<div class="timeline-badge success">

<i class="fa fa-search"></i>

</div>。



图 7 信息统计前端可视化页面图

Fig. 7 Information statistics front-end visualization page

## 4.4 关键模块实现

将按照系统管理模块、设备管理模块、文档管理模块、动态信息管理模块几大重要模块的实现页面和功能实现的关键代码。

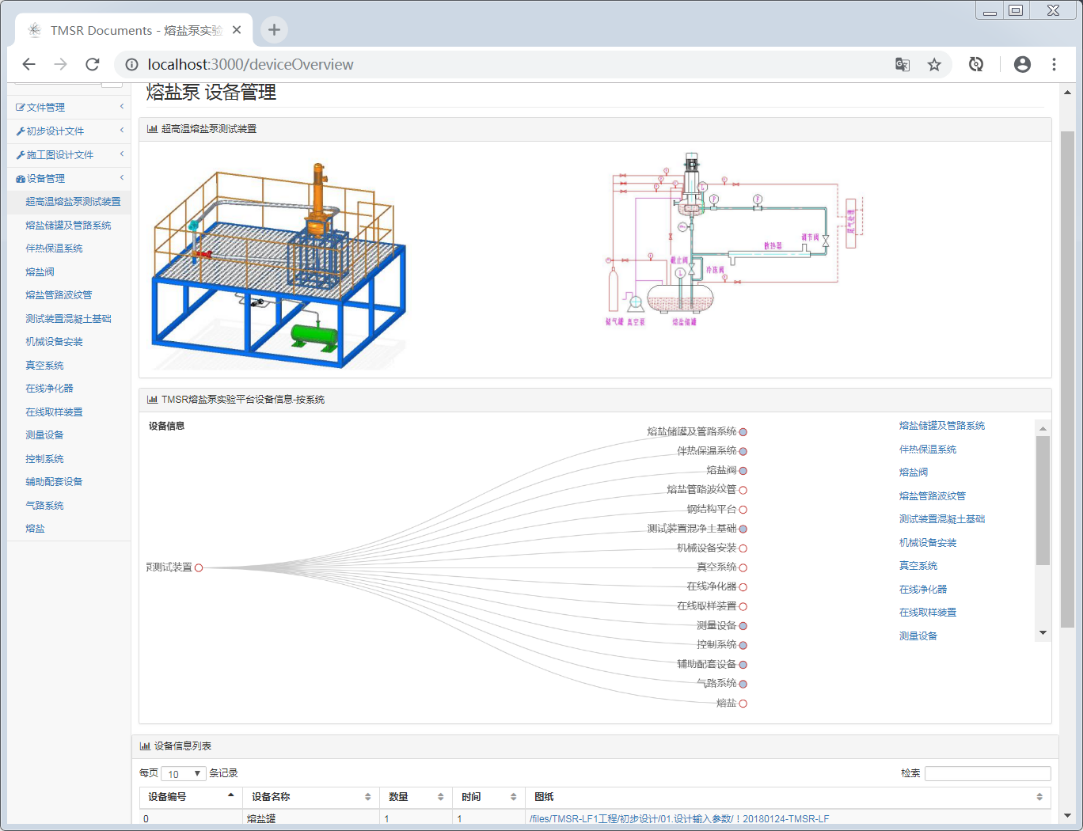
### 4.4.1系统管理模块

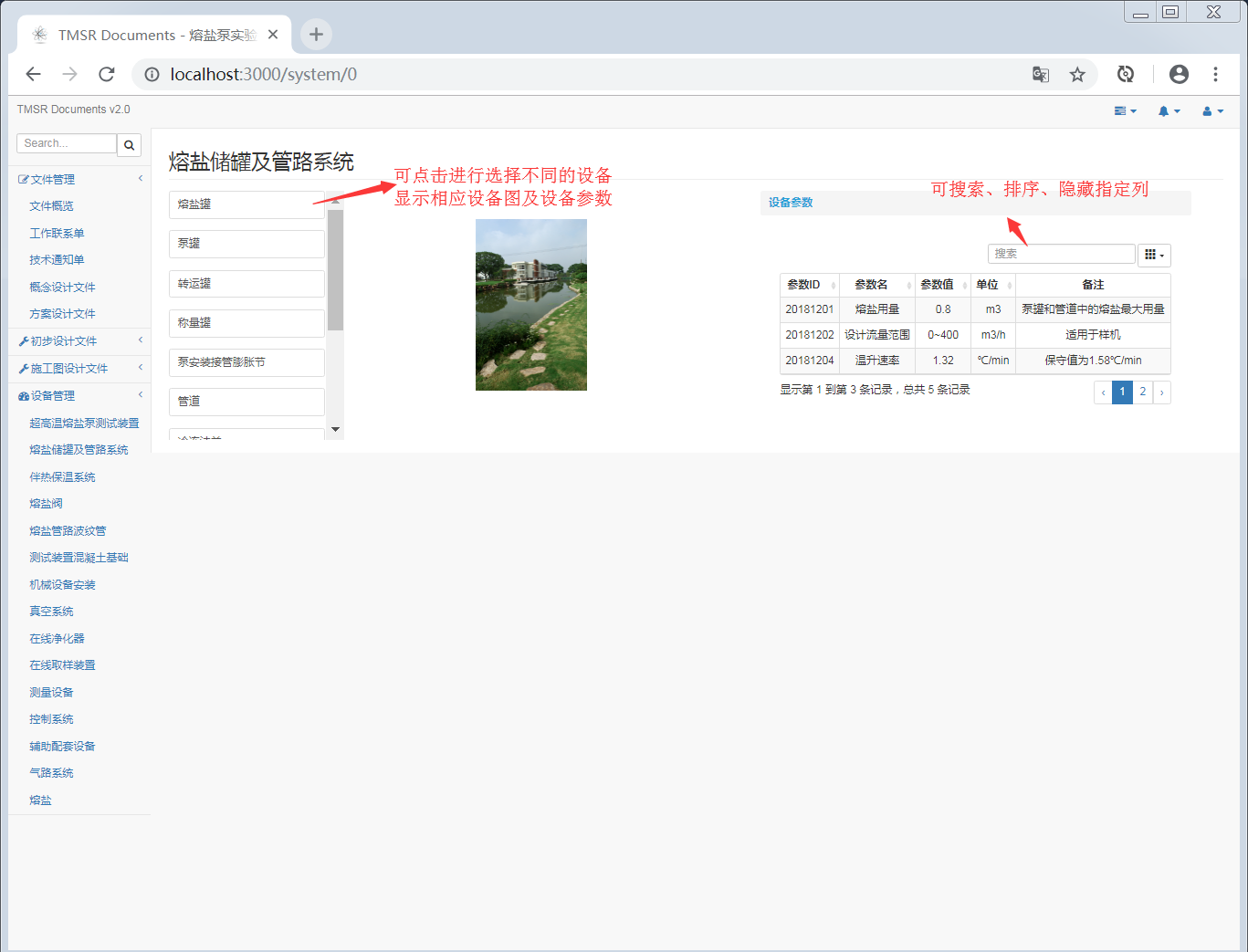


由于开发的基于TMSR熔盐泵实验平台数据库管理系统属于内部系统，具有一定的隐私性，需要对用户进行管理访问，另外系统的功能包括设备信息的增改删除等操作，以及文档管理具有文档的流程追踪操作，动态信息管理属于对动态信息的存档历史信息进行查询等操作，对于普通用户、设计人员、管理人员相应地有不同的权限能否执行这些操作。系统管理模块就是通过用户管理，给不同的用户添加相应的权限，再通过权限管理，不同的权限控制访问不同的操作，来实现对用户身份认证和操作限制/允许的功能。所有用户可以按照所在的同一类权限，不同的组内的用户有不同的访问系统权限。

### 4.4.2设备静态信息管理模块







如图所示，进入设备管理界面，可以实现

### 4.4.3文档信息管理模块



如上图所示，文档管理包含的数据信息分为两类：列表信息数据和文件数据。在项目的设计过程中，可以看做是面向对象的，而这两类数据实际上是面向对象的设计过程中类和对象的关系。其中 文件对象的类包括：设计文件、开口项描述文件、不确定项描述文件、会议纪要等。

TMSR熔盐泵实验平台在设计阶段为了解决设计人员、施工人员的沟通不畅，文件版本无法进行统一控制造成人力、资源的浪费，同时也是为了保障工程的质量和优化进度，建立起以多项目、多级的管理体系以及计划管理、过程控制和质量监督三维一体为管理理念【】的文档管理。其中涵盖的数据信息范围结合实验平台设计和建设过程中 的实际情况，包含了方案预研、初步设计、工程设计、施工设计、工程建造、工程验收全周期。另外还考虑到在不同的阶段核心数据会发生转移，例如在施工和验收阶段，更注重于关于系统测试、工程进度跟踪、工程施工等数据的管理。为此开发了包括高效快捷的文档搜索功能，多维度的分类、汇总、快速查询功能，用户权限管理、时间进度管理功能、设计文件版本控制功能的文档管理模块。

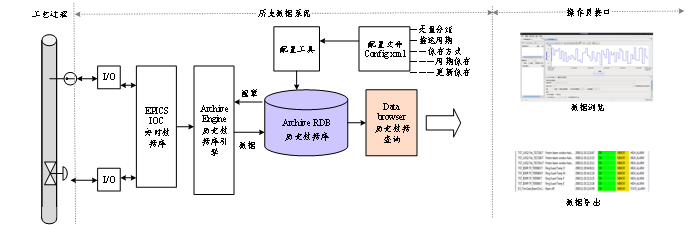




如图所示，

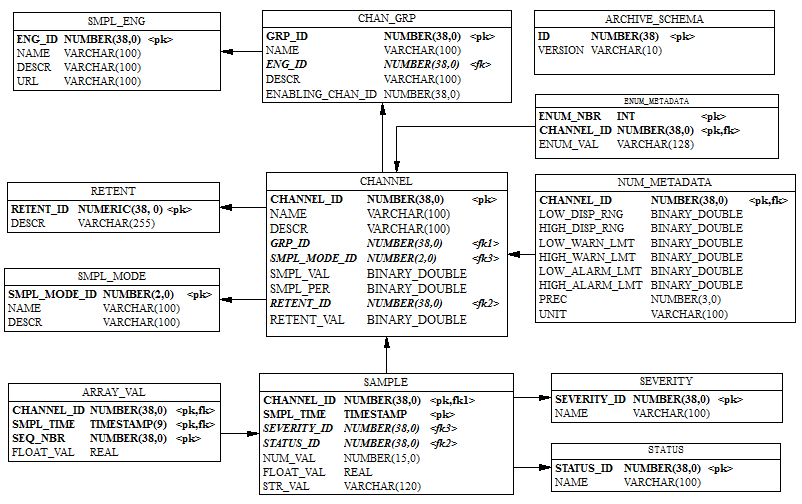
### 4.4.4 设备动态信息管理模块

设备动态信息指的是在熔盐泵实验平台使用过程中，设备运行期间，与设备运行相关的数据，即设备的控制参数例如熔盐泵泵罐液位计的读数，泵罐气路系统压力计的压力读数，泵的上轴承温度下轴承温度等。这些动态数据实际上是通过EPICS IOC采集到设备产生的实时数据，在通过ArchiveEngine历史数据库引擎将采集到的过程变量存储到关系型数据库上，本系统选用的是Postgresql关系型数据库，最后利用Nodejs、JavaScript、Angular等前后端开发技术，实现数据的检索、图形的绘制以及数据的导出。



通过AchiveEngine历史数据库引擎将采集到动态数据存储到关系型数据库中，实际上获得的是动态实时数据的历史归档数据。本系统就是将这些历史归档数据实现网页数据检索、图形绘制以及数据导出功能。

历史归档数据最终保存到Postgresql数据库中，主要包含12张表，分别是采集引擎表、通道组表、采集模式表、通道表、报警级别表、报警状态表、采样表、归档版本表、采样数组表、元数据预留表、数值报警表以及枚举通道表，这些表格在数据库中的命名分贝是SMPL\_ENG、CHAN\_GRP、SMPL\_MODE、CHANNEL、SEVERITY、ARRAY\_VAL、RENTENT、NUM\_METADATA、ENUM\_METADATA。下面介绍最关键的几张表，SAML\_MODE表定义了数据的记录方式，有两种方式，一种使周期扫描—scan，另一种是实时监测数据更新—monitor。其中scan扫描模式会接收到所有更新数据，但只记录特定间隔最近的数据；而Monitor监视模式接收所有的PV值变化，每个接收到的数据和前值进行比较，当差值超过预先设定阈值是才被存储。Channel表记录了需要采集的变量名称，以及规定它们的数据采集方式（SMPL\_MODE\_ID）、采集周期（SMPL\_PER）、以及采集阈值设定（SMPL\_VAL）等信息；CHAN\_GRP表对于所有PV变量（通道）进行分组管理；SAMPLE表存储采集数据。如下图所示：



## 4.5 本章小结

# ~~第五章 系统测试~~

## ~~5.1 测试环境~~

## ~~5.2 系统功能测试~~

## ~~5.3 本章小结~~

# 第六章 总结与展望

## 6.1 总结

本文结合TMSR研究堆项目的实际，建立起多阶段的工作流模式、组织架构、质量规范体系和生命周期管理，综合使用了Node.js软件平台，PostgreSQL数据库，AngularJS框架和最先进的基于HTML5标准的前端技术，开发出一套以TMSR项目计划管理、流程控制、质量监督为核心的设计数据管理系统[13]。该系统通过定义两类元数据：列表信息数据和文档元数据，实现了设计过程的追踪和管理[14]。基于MVC架构实现了设计数据管理平台前后端的数据交互。最后应用在钍基熔盐堆仿真验证装置上，通过进行计划清单管理，开口项管理和接口管理等活动，验证了可行性，为设计及开发堆项目的设计管理系统提供参考[15]。

## 6.2 特色与创新

创新点：

代替传统的EXCEL、Word或单独靠独立工具的记录或管理方式，将内部流程或者数据资源全部串联整合起来，打破各个部门的信息孤岛，让管理者可以迅速并准确了解到外部对于自身管理的影响，如工程部门可以快速了解到采购进度、施工进度对于总控进度的影响，财务可以快速了解到花费部门对于预算、资金的影响，现场管理者可以快速了解到供应商到货对施工进度的影响。不仅减少了大量重复发送、修订、接收文件的工作，降低了部门与部门，人员与人员之间的关系压力，同时也大大提高了项目资源的整个能力以及管理控制能力。

本管理系统将代替原有的周、月报等管控方式，让管理者可以随时随地了解到各个项目的进度情况，同时系统实施质量监督的功能，责任分配到人。

熔盐泵实验平台的设备动态信息，实际上是设备运行时通过CA从EPICS IOC RTDB中定期取得的存档数据，是各个设备的控制数据。EPICS（实验物理及工业控制系统，Experimental Physics and Industrial Control System）实际上是客户端属于B/S架构，通过本系统采用的C/S架构可以随时随地查询到设备的动态存档数据。

手工做报表的形式繁琐且容易出错，避免不了人力的浪费和人工的差错。用管理系统的报表功能，可以自定义要求，节省人力物力，满足全方面需求打印出来。

EPICS即“实验物理及工业控制系统”，作为大型控制软件系统，专门开发用于大型实验物理设备。EPICS的软件结构包括两个主要的部分，IOC（Input Output Control）层实际上是EPICS实时控制的核心，EPICS将控制的具体对象和对它们的操作抽象为一系列记录对记录的操作，IOC的核心是常驻内存的数据库。运行数据库（Run-time Database,简称RTDB）分布在各自的IOC中，RTDB数据库为了便于实时访问，数据结构采用简单flat表。静态的数据库通常将数据存放在外部海量存储设备中，而动态的RTDB为了便于实时访问，将数据存放在内存中。对于分布式系统，RTDB分布在系统的各个节点中，供操作员提取。

## 6.3 工作展望

参考文献

1. 核电工程数据库的现状及发展设想，《核动力工程》
2. 核电厂D-RAP数据库设计，上海核工程研究设计院，《发电设备》