

分 类 号 _____

密 级 _____

U D C _____

单位代码 _____ 10151 _____

港口码头集装箱生产作业管理系统

陈 鹏

指 导 教 师 赵广利 职 称 副教授

企 业 副 导 师 马全华 职 称 副高工

学位授予单位 大 连 海 事 大 学

申请学位级别 工程硕士 学 科 与 专 业 计算机技术

论文完成日期 2010 年 9 月 论文答辩日期 2010 年 11 月

答辩委员会主席 邓安生

1901395



Y1836854

Container Port Production Operations Management System

A thesis Submitted to

Dalian Maritime University

In partial fulfillment of the requirements for the degree of

Master of Engineering

By

Chen Peng

(Computer Technique)

Thesis Supervisor: Associate Professor Zhao Guangli

Assistant Supervisor: Ma Quanghai

September 2010

大连海事大学学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明：本论文是在导师的指导下，独立进行研究工作所取得的成果，撰写成硕士学位论文“港口码头集装箱生产作业管理系统”。除论文中已经注明引用的内容外，对论文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文中不包含任何未加明确注明的其他个人或集体已经公开发表或未公开发表的成果。本声明的法律责任由本人承担。

学位论文作者签名：陈鹏

学位论文版权使用授权书

本学位论文作者及指导教师完全了解大连海事大学有关保留、使用研究生学位论文的规定，即：大连海事大学有权保留并向国家有关部门或机构送交学位论文的复印件和电子版，允许论文被查阅和借阅。本人授权大连海事大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，也可采用影印、缩印或扫描等复制手段保存和汇编学位论文。同意将本学位论文收录到《中国优秀博硕士学位论文全文数据库》（中国学术期刊（光盘版）电子杂志社）、《中国学位论文全文数据库》（中国科学技术信息研究所）等数据库中，并以电子出版物形式出版发行和提供信息服务。保密的论文在解密后遵守此规定。

本学位论文属于： 保 密 ☐ 在____年解密后适用本授权书。

不保密 ☒ （请在以上方框内打“√”）

论文作者签名：陈鹏 导师签名：赵广利

日期：2010年11月4日

2
1
1
2

2
1
1
2

摘 要

滨海新区开发开放步伐的加快,为天津港吞吐量的不断增长注入了强劲动力。在进一步加快滨海新区开发开放的新形势下,在京津冀、环渤海及中西部地区的经济发展中,天津港要进一步增强服务功能,兼顾各种货类的协调发展,坚持综合性港口的发展方向。只有这样,才能更加适应腹地经济发展的需要,才能进一步增强对区域经济的辐射力和影响力,更好地发挥滨海新区的辐射带动作用。努力把天津港建设成为我国北方对外开放的门户、高水平的现代制造业和研发转化基地、北方国际航运中心和国际物流中心。

本论文课题的设计与开发,主要是针对天津港集装箱码头现行业务需求和操作流程,做了比较详细的业务调研和需求分析,研发了符合天津港码头操作规范的集装箱码头生产作业管理系统。系统由六大功能模块组成:1)操作便捷的 EDI 报文收发系统;2)功能完备的基本操作系统;3)动态实时的智能操作系统;4)高效全面的商务费收系统;5)灵活简洁的权限控制系统;6)及时准确的数据统计系统。系统主体框架采用 C/S 两层结构,并以三层结构为辅助。系统前端开发工具采用 PowerBuilder 和 VC++,后台数据库采用 Oracle,通过中间件为载体实现消息实时传递,大量采用图形化技术和数据库的存储过程、触发器,增加了数据处理的实时性。

目前,系统已在天津港联盟国际集装箱码头、天津港太平洋国际集装箱码头、天津港欧亚国际集装箱码头投入正式运行。系统能够满足码头正常生产作业的需要,显著的降低了人工成本和运行成本,提高的码头作业效率,而且改善了码头堆存的有效利用率,得到了使用系统码头的好评和认可。随着应用的不断深入,该系统将做进一步完善,力争进一步提高管理水平和服务质量。

关键词: 集装箱; 生产作业管理; 存储过程; 装卸船算法设计

ABSTRACT

Binhai district, speed up the pace of tianjin port throughput for the constant growth of injected strong power. In further accelerate the openness and development of binhai new area, under the new situation of the bohai sea in jingjinji, and the economic development in the Midwest, tianjin port to further enhance the service function, balancing the coordinated development of various goods, adhere to the comprehensive port development direction. Only in this way can we better suited to the hinterland of the need of economic development, so as to further strengthen the regional economic radiation and influence, play a better role in driving the binhai new area of radiation. Efforts to build a port in north China opening doors, a high level of modern manufacturing and r&d transformation base, northern international shipping center and international logistics center.

Subject of this paper the design and development, mainly for Tianjin Port Container Terminal current business needs and operational processes, do more detailed research and business needs analysis, research and development operations in keeping with standard Tianjin Port Container Terminal Management System for production operations. System consists of six functional modules: 1) that are easy to send and receive EDI message system; 2) complete the basic operating system functions; 3) dynamic real-time intelligent operating system; 4) a comprehensive business cost and efficient collection system; 5) flexible simple access control system; 6) timely and accurate statistical data system. The main framework of the system using C / S two-story structure, and to three-tier structure for aid. System using PowerBuilder front-end development tools and VC, the background database using Oracle, through real-time middleware for the carrier to achieve message passing, a lot of technology and the use of graphical database stored procedures, triggers, increased data processing in real time.

At present, the system has been in the Tianjin Port Alliance International Container Terminal, Tianjin Port Pacific International Container Terminal, Container Terminal in Tianjin, Hong Kong, Europe and Asia into formal operation. System can meet the needs of terminal normal production operations, significantly reducing labor costs and operating costs, improved efficiency of terminal operations, and improve the effective utilization of terminal stockpiling, use the system terminal has been well received and

approved. With the deepening of the application, the system will further improve and strive to further improve the management level and service quality.

Key Words: Container; Manufacturing operations management; Stored procedures; Handling vessels Algorithm

目 录

第1章 绪论.....	1
1.1 问题的提出及背景.....	1
1.2 国内外的研究现状.....	1
1.3 论文的主要研究内容.....	2
1.4 本文的章节结构.....	2
第2章 系统开发环境.....	3
2.1 VC++6.0.....	3
2.2 SYBASE POWERBUILDER 简介.....	3
2.2.1 Sybase PowerBuilder 说明.....	3
2.2.2 PowerBuilder 特点.....	3
2.3 数据库平台选择.....	4
2.4 系统运行模式与环境的简要说明.....	6
第3章 系统需求分析.....	8
3.1 系统需求分析.....	8
3.1.1 用户需求.....	9
3.1.2 系统调研.....	10
3.2 业务流程.....	10
3.2.1 进口卸船流程.....	11
3.2.2 出口装船流程.....	15
3.3 系统数据流例图.....	17
3.3.1 进口船图数据维护用例.....	17
3.3.2 出口配载用例.....	17
3.4 系统数据流图.....	19
第4章 系统设计.....	21
4.1 系统的设计目标.....	21
4.2 系统设计原则.....	21
4.3 系统设计逻辑结构与系统机构图.....	22
4.4 系统功能模块划分.....	25
4.5 系统功能模块设计.....	26
4.5.1 EDI 收发子系统功能模块设计.....	27
4.5.2 基本操作子系统功能模块设计.....	27
4.5.3 计划实时控制子系统功能模块设计.....	28

4.5.4 商务计费子系统功能模块设计	28
4.5.5 权限管理子系统功能模块设计	29
4.5.6 统计报表查询子系统功能模块设计	29
4.6 系统数据库设计	29
4.6.1 数据库设计	29
4.6.2 数据库设计特点与要求	29
4.6.3 数据结构	30
4.6.4 表与表之间关联	33
第 5 章 系统主要功能实现	36
5.1 系统卸船作业功能的实现	36
5.1.1 船舶动态维护	36
5.1.2 EDI 报文导入——进口船图	37
5.1.3 EDI 报文导入——进口舱单	37
5.1.4 进口船图数据维护	37
5.1.5 舱单数据维护	37
5.1.6 船舶作业线分配	37
5.1.7 堆场计划的制定	39
5.1.8 机械区域计划的设定	40
5.1.9 卸船作业指令发送	40
5.1.10 岸边理货员手持卸船操作	41
5.1.11 集卡全场调度	41
5.1.12 场桥落场确认	42
5.1.13 系统卸船回退功能的实现	42
5.1.14 系统卸船找位算法的实现	44
5.2 系统装船作业功能的实现	47
5.2.1 出口装载清单的导入	47
5.2.2 出口装载清单维护	47
5.2.3 出口堆场计划设定	48
5.2.4 闸口收箱	48
5.2.5 出口箱放行操作	49
5.2.6 出口船图配载	52
5.2.7 集装箱三态显示与作业队列四层模型	52
5.3 消息服务器	54

第 6 章 总结与展望.....	56
6.1 总结.....	56
6.2 展望.....	56
参考文献.....	57
攻读学位期间公开发表论文.....	59
致 谢.....	61

第1章 绪论

1.1 问题的提出及背景

信息技术是当代最具潜力的新的生产力,从工业经济到信息经济,从工业社会到信息社会,在这个动态演进过程中,信息产业逐步上升成为推动世界经济和社会全面发展的关键因素,成为人类进步的新标志^[1]。近年来,天津港在建设中国北方国际航运中心的进程中,始终保持着高速的成长性,与国际知名船公司及码头运营商合资兴建的现代化专业化集装箱码头比比皆是,这些新码头的建设投产使天津港的岸线长度、泊位数量、堆场面积、机械能力等有了很大的提升,为接卸超五代、第六代集装箱船舶提供了强有力的物质保障。在全球经济危机的大环境下,我国政府高度关注包括港口在内的基础产业,出台了一系列重要政策和应对举措,指导港口业的发展,就港口应对危机、促进经济发展做出重要指示。国务院相继出台了“物流业调整振兴规划”、交通运输部还将研究出台“拓展港口服务功能的指导意见”,这有利于我们在贯彻落实的过程中,扩大视野,创新港口经营管理模式,进一步提高我国港口应对危机的能力和管理水平^[2]。

企业管理信息化是企业核心竞争力的源泉^[3],在天津港信息化建设快速发展的大环境下,尽快使港口集装箱生产插上数字化的翅膀,提高港口集装箱处理能力,提高码头运行效率,提高集装箱码头的生产经营和信息服务的水平,是我们研究的主要目的。

1.2 国内外的研究现状

目前国际上较好的几个集装箱码头管理系统开发商在全球港口已经占据了近60%的市场,然而大多数港口只购买其产品核心,而外围产品如财务、计费、EDI传输、统计、决策支持等都要二次开发,每年支付巨额的系统维护费,由于厂商的控制,用户只有使用权,没有原代码,致使用户在应用和发展等方面都受到相当大的制约。天津市委八届八次全会明确要求天津港要“加快港口建设,增强服务功能,搞好科学管理,提高吞吐能力,使天津港成为世界一流大港”^[4]。因此,天津港集团公司在对国内外集装箱码头管理系统的了解和研究的基础上,吸取先进经验,开发有自己特色的,适合天津港集团发展需要的集装箱码头自动化作业

管理系统，实现天津港内部码头信息资源整合，避免信息孤岛情况的发生变得十分必要，意义重大，同时该项目的研发成功是天津港进一步整合现有信息资源的必要条件

集装箱码头自动化作业管理系统功能覆盖港口集装箱码头生产作业的全部领域，包括码头基本操作系统、智能计划系统、智能操作系统、箱站管理系统、费收管理系统、EDI 报文转换系统、RFID 技术应用、GPS 卫星定位技术应用、WEB 客户服务系统、决策支持系统、可视化作业控制系统、维护服务系统等子系统。涵盖码头生产的全过程，包括面对生产第一线人员的实时生产指挥系统，体现工艺创新；面向经营管理人员的评价决策系统，体现管理创新；面向码头各类客户的客户服务系统，体现数字化港口为服务创新理念；面对技术人员的计算机数字安全系统，体现整个系统以安全为本的理念。

1.3 论文的主要研究内容

天津港集装箱码头作业要求的作业管理系统（CTOS），已经在天津港联盟国际、太平洋国际、欧亚国际投入正式应用。CTOS通过对劳动力、堆场空间和机械设备的实时使用情况进行优化，使码头效率、产量和经济效益实现最大化。系统主要研究内容如下：装卸船作业算法、堆场找位算法、拖车调度、无线终端以及消息机制等。

1.4 本文的章节结构

针对上述研究内容，可将本文分为六个章节，每个章节的内容安排如下：

第一章：绪论。主要介绍论文的研究背景、研究意义及主要工作。

第二章：系统研究的理论性、开发工具以及运行环境的选择。主要介绍管理信息系统相关概念、介绍 C/S 模式以及系统运行环境与开发工具介绍。

第三章：系统需求分析。从集装箱码头业务介绍入手，全面阐述了业务流程调研，系统客户化需求分析，并形成业务流程及数据流程，通过图例进行分析。

第四章：系统设计。从系统设计目标性和逻辑性以及相应的业务结构、应用结构入手，对系统中各个模块的功能进行详细阐述。

第五章：系统主要功能实现。

第六章：总结和展望。

第 2 章 系统开发环境

本章主要对集装箱码头生产作业管理系统架构和研发过程中所涉及的开发工具、开发平台以及相关的理论依据进行说明和介绍。

2.1 VC++6.0

VC++是一个功能强大的可视化软件开发工具。自 1993 年 Microsoft 公司推出 VC++1.0 后,随着其新版本的不断问世,VC++已成为专业程序员进行软件开发的首选工具。VC++6.0 不仅是一个 C++编译器,而且是一个基于 Windows 操作系统的可视化集成开发环境(integrated development environment, IDE)。VC++6.0 由许多组件组成,包括编辑器、调试器以及程序向导、类向导等开发工具。这些组件通过一个名为 Developer Studio 的组件集成为和谐的开发环境^[5]。

集装箱码头操作系统,需要极好的兼容性和稳定性,因此选择 VC++作为开发平台完成图形化窗口的开发。

2.2 Sybase PowerBuilder 简介

2.2.1 Sybase PowerBuilder 说明

Sybase PowerBuilder 是易于使用的、可伸缩的、并经实践证明的快速集成开发环境,包括窗口、菜单、事件逻辑、数据库存取、报表、图形等都能在 PowerBuilder 内完成开发,相对于其他开发工具 PowerBuilder 可以使开发人员的工作速度更快,开发成本更低,产品功能更强。PowerBuilder 在客户机/服务器结构应用中具有描述多个数据库连接与检索的前端能力,其应用开发的快捷性、简便性以及先进性得到了广大用户认可^[6]。

2.2.2 PowerBuilder 特点

1、可视化的开发环境: PowerBuilder 是一个用于传统客户机/服务器应用、分布式应用和 Web 应用的集成完全可视化的开发环境。使用 PowerBuilder,可以用一种可视直观的方式来创建应用程序的用户界面和数据库接口。PowerBuilder 应用程序通常包含用户界面和应用处理逻辑。

2、面向对象、时间驱动的开发工具：PowerBuilder 变革了面向对象技术的使用，把它应用到用户界面对象中，具有面向对象编程的各种特性，与传统的开发工具有着本质的区别。数据窗口是基于 C/S 结构或分布式结构 PowerBuilder 应用程序中的 PowerBuilder 控件^[7]。PowerBuilder 的窗口、菜单、数据窗口等称为对象。用户可以创建构造自己的对象。对象都有一组与之关联的系统预定义的事件、属性和函数。在 PowerBuilder 中，可以利用面向对象的编程技术的特点，如多态性、封装和集成来提高系统的重用性、可扩展性和健壮性。

3、功能强大的编程语言：可以非常方便的使用 PowerBuilder 的高级编程语言 PowerScript 为每个对象的事件编写脚本，就象使用 BASIC 语言编程一样简单。PowerScript 同时也提供了丰富的内置系统函数，用于对应用进行操作。

4、开发的系统：PowerBuilder 可以访问当今任何一个数据库系统。它本身也带一个数据库 Adaptive Server Anywhere。PowerBuilder 为各类数据库提供了多种方便的访问方法。PowerBuilder 提供了通用的前台用户界面，开发人员可以无需关心每个数据库的特性，而集中精力进行前台应用程序的开发。

5、跨平台的开发环境：PowerBuilder 支持跨平台的开发和发布。

PowerBuilder 开发环境现在可以支持 Microsoft Windows, Sun Solaris, I-P-UX, IBM AIX 和 APPLE Macintosh 等操作系统。

6、支持多种开发解决方案：PowerBuildery 应用程序可以是传统的客户机/服务器两层访问数据库服务器的应用。也可以用 PowerBuilder 建立分布式多层应用。PowerBuilder 应用程序也可以基于 Web 来专门为 Internet 开发客户端的 Web 应用，或者由已有的 PowerBuilder 应用程序扩展^[8]。

2.3 数据库平台选择

Oracle9i 是由 Oracle 公司开发的、面向 Internet 计算的、支持关系对象模型的分布式数据库产品。作为一个通用的关系型数据库系统，它具有较完整的数据管理功能，而且是一个十分优秀的完备关系产品；作为分布式数据库系统，它实现了简单的分布式处理功能；作为一个现代化的软件产品，它具有很好的开放性。Oracle 良好的安全性和数据一致性更是令人赞叹不已，正因为 Oracle 拥有

如此之强大的功能，所以被政府企业部门广泛使用^[9]。总的来说，Oracle 具有以下特征：

1) Oracle 支持所有可用于客户端/服务器的操作系统，包括：MS DOS，WINDOWS，OS/2，以及大多数 UNIX 风格的操作系统。

2) Oracle 具有开发者可用的一些特征，开发人员利用这些特征，可使网络中客户端/服务器之间的输入/输出降低到最少。

3) Oracle 具有易于管理复杂的客户端/服务器系统的特征^[10]。

鉴于天津港生产相关的管理信息系统大部分采用 Oracle 公司系列产品，为便于未来天津港信息资源共享，天津港集装箱码头作业管理系统仍然选择 Oracle9i 作为数据库开发平台。

Oracle 存储过程介绍及应用：

1) 存储过程的概念

存储过程 (Stored Procedure) 是一组为了完成特定功能的 SQL 语句集，经编译后存储在数据库中。用户通过指定存储过程的名字并给出参数 (如果该存储过程带有参数) 来执行它。存储过程是数据库中的一个重要对象，任何一个设计良好的数据库应用程序都应该用到存储过程。存储过程是由流控制和 SQL 语句书写的过程，这个过程经编译和优化后存储在数据库服务器中，应用程序使用时只要调用即可。在 ORACLE 中，若干个有联系的过程可以组合在一起构成程序包^[11]

2) 存储过程的优点

① 存储过程的能力大大增强了 SQL 语言的功能和灵活性。存储过程可以用流控制语句编写，有很强的灵活性，可以完成复杂的判断和较复杂的运算。

② 可保证数据的安全性和完整性。

(一) 通过存储过程可以使没有权限的用户在控制之下间接地存取数据库，从而保证数据的安全。

(二) 通过存储过程可以使相关的动作在一起发生，从而可以维护数据库的完整性。

③ 再运行存储过程前，数据库已对其进行了语法和句法分析，并给出

了优化执行方案。这种已经编译好的过程可极大地改善 SQL 语句的性能。由于执行 SQL 语句的大部分工作已经完成,所以存储过程能以极快的速度执行。

④ 可以降低网络的通信量。

⑤ 使体现企业规则的运算程序放入数据库服务器中,以便集中控制。当企业规则发生变化时在服务器中改变存储过程即可,无须任何应用程序。企业规则的特点是要经常变化,如果把体现企业规则的运算程序放入应用程序中,则当企业规则发生变化时,就需要修改应用程序工作量非常之大(修改、发行和安装应用程序)。如果把体现企业规则的运算放入存储过程中,则当企业规则发生变化时,只要修改存储过程就可以了,应用程序无须任何变化^[12]。

在本系统的研发设计阶段,从保证系统稳定性、灵活性、便捷性以及执行力角度出发,大规模的使用了数据库存储过程,因为考虑到业务的繁杂程度,许多地方都需要依靠表之间的关联来实现功能,且码头生产时的特点是数据量大、实时、多点同步性,数据库访问量相当大,如果单单的依靠 SQL 语句与数据库的交互,无形中增加服务器的负载,这无疑是资源的浪费,也是不科学的。若将算法和逻辑实现功能局限于客户端,无论是在 PB 中,还是在 VC++ 环境里,代码的复杂程度可想而知,出现问题也不利于调试,并且在生产运行的时候,就客户端而言,从性能上会带来很大的冲击和考验。于此同时上千上万的 SQL 语句与数据库之间的交互访问,必然似的服务器的工作效率和合理利用率大打折扣,得不偿失,进而也会使得系统与现实隔离性加大,在重合点和叠加性上出现偏差,软件失去了指导操作的意义^[13],也降低了码头作业的的生产效率。

目前 CTOS 系统在天津港集团使用投产的三家集装箱码头使用的数据库服务器为 IBM P590, P595 级别,且采用 RAC 方式运行,数据处理能力一流,倘若将大批量的数据处理转移到服务器端进行解决,大大降低了客户端配置需求,节省了不必要的开支。不难得出结论,采用存储过程在后台处理算法问题,客户端的工作仅仅是简单的传递参数和接收返回值,何乐而不为。

2.4 系统运行模式与环境的简要说明

1) 系统运行模式:天津港集装箱码头作业管理系统具有数据量大,使用点多和使用频率高,择优选取 C/S 和 B/S 结构各自特点,为保证系统的运行速度、保证

系统的实时控制、保证系统的客户化需求，系统基本操作功能使用 C/S 结构模式运行，WEB 网站发布码头相关信息功能使用 B/S 模式运行。

2) 开发运行环境：

数据库：Oracle9i

开发工具：PowerBuilder 10, Visual C++

系统运行环境如图 2.1 所示。

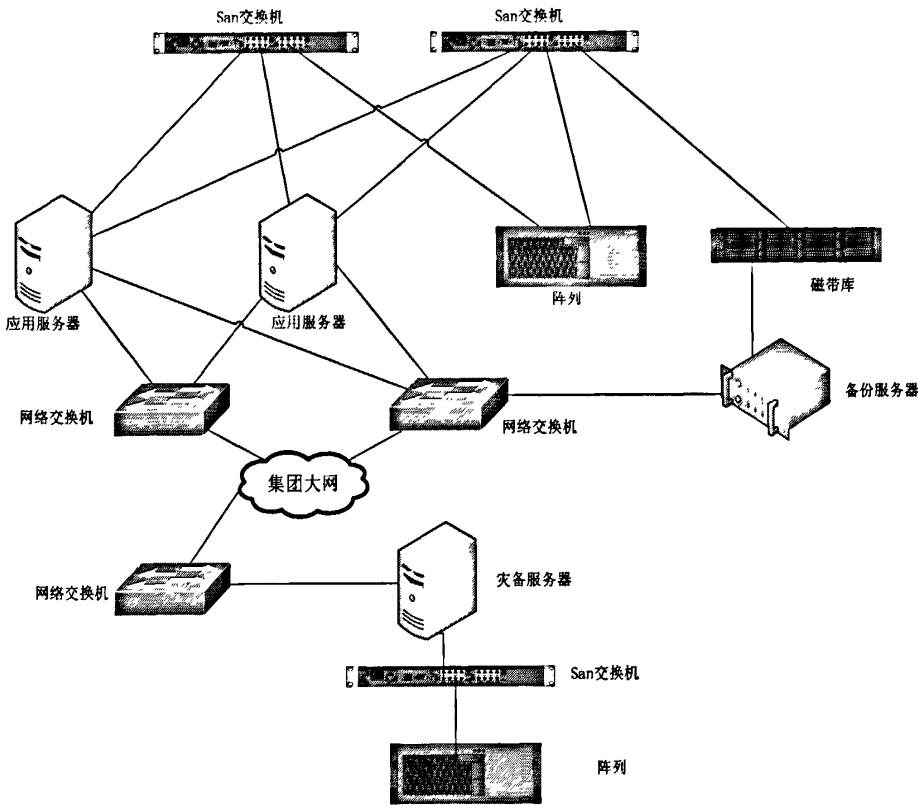


图 2.1 系统运行环境规划图

Figure 2.1 System operating environment plan

第3章 系统需求分析

什么是系统需求？简而言之，系统需求就是从用户的应用特点出发，依据其总体规划和目标，对系统进行抽取细化，就系统性能、规格、行为等方面总结出来的期望。更进一步，用户需求可以分为两类：系统需求和软件需求。前者是整个系统所提出的需求，后者则是系统中软件部分所涉及的需要。如何进行需求调研？需求调研涉及三个问题。一是如何确定调研对象，二是如何确定被调研对象，三是采用何种调研方法。调研对象的组成应以互补为原则，至少要有三类人员组成：技术人员、业务专家和管理者。被调研对象主要是人员和业务两类，其间主要涉及人与人、人与事物、事物与事物等三种关系。其中，关键是确定调研范围。调研范围包括关键域和关键活动。而关键活动又由关键流程加关键点构成。找到关键域，明确关键流程和关键点，对需求调研至关重要，需要专家或咨询顾问介入。而能否把握这一时机并找准需求提炼的关键点，是考验需求调研人员的重要方面。优秀的需求调研人员不仅能认识问题之所在，还能藉此获取足够多的知识，最后成为问题领域的专家。程序的设计研发是建立在系统需求调研和分析的基础之上，详细、全面的需求分析会使得程序设计的更加合理、全面、人性化。本章主要是根据集装箱码头已经形成的比较系统的操作模式，对天津港集装箱码头的业务流程和客户需求，进行详细的阐述和说明。

3.1 系统需求分析

开发阶段的第一步，也是最重要的一步为捕获需求，对用户需求的准确捕获对后面的开发工作、设计工作带来深远的影响，它是整套系统开发工作的根基，在很大程度上决定着系统开发能否成功^[14]。就集装箱作业本身而言，在需求调研、需求分析阶段，许多问题不单单是限于表面，为了调研而调研，称 CTOS 系统是个工程，并不过分，它所包含的业务形态与流程很多很杂，且每一个部分都是必不可少的。为了使得 CTOS 系统能够满足天津港当前的业务生产需要，稳定、实时、高效、智能、灵活这是这个系统的关键点。因而对 CTOS 系统业务处理、现场形态、数据结构等等，必须深入的分析，积极的应对，全面的理解。系统的成功研发与实施要走的路很长，是一个继承与创新的过程。为了保证系统需求调研、分析、

设计的准确性、有效性、合理性、便捷性，作为系统设计人员应该做到理论联系实际，并且遵循以下几个原则：

一、抓业务重点和难点，分主次进行规划、整合系统涉及的业务形态与范围：

切实依照 CTOS 中基本操作模块、实时计划模块、堆场编辑模块、船舶编辑模块这四个关键模块涉及业务重点，详细深入全面的进行 CTOS 需求调研与需求分析。

二、系统需求以人为本，建立健全业务流程模型

充分利用计算机的强大功能，指定完备的业务流程，抽取业务对象，通过优化和重组，极大限度的满足用户的客户化需求。

三、面向业务对象，分析实际业务发生过程模型

利用面向对象的方法来分析与设计业务，并分析业务 workflow 模型及其业务数据模型。

3.1.1 用户需求

天津港是中国最早拥有集装箱专用码头的港口（1980 年），经过三十来年的发展，她的集装箱码头运作日趋成熟。特别是进入新千年以来，天津港为了实现“与全球同步发展，建设世界一流大港”的战略目标，大力推进了集装箱码头的建设发展，2000 年后成立了诸如以下的新码头：2004 年五洲国际 (FICT) 集装箱码头成立、2006 年联盟国际 (TACT) 集装箱码头成立、2008 年太平洋国际 (TPCT) 集装箱码头成立、2010 年欧亚国际 (TECT) 集装箱码头成立。新码头的建设成立，无论是在泊位吃水能力，还是码头岸线长度都符合世界一流标准，同时引入了世界上最先进的集装箱作业机械，为天津港集装箱开辟新航线新市场以及逐年增长的集装箱吞吐量都做好了充分准备。硬件条件上去了，进而对码头装卸船的作业效率提出了更高的要求。天津港目前的集装箱码头使用的系统有两种：COSMOS 系统和 CTMS 系统，这些码头目前使用的仅仅是产品的核心功能，而附带模块包括泊位计划、EDI 系统接收传输、商务计费、数据统计以及辅助决策支持和 WEB 查询等等，都需要码头进行二次开发，且使用的开发工具很多，大多使用的接口形式，而且有的系统是全英文界面，从而使得系统的完整性不强、灵活度低，功能扩展和系统维护都存在很大的阻碍，势必会对码头生产效率起到反作用，发展创新才是硬

道理，软件升级是一种必然。为了天津港码头集装箱系统的能够做到统一完备，在集团公司领导的大力倡导与监督下，本着适应天津港集装箱业务需求，研发了天津港特有的 CTOS 系统软件。该软件的投产，加快了港口码头集装箱的信息化建设，在联盟国际、太平洋国际、欧亚国际使用该系统的基础上，集团领导积极推进全港集装箱的统一管理，现在正在进行天津港集装箱码头公司，天津港五洲国际码头公司和天津港东方海陆码头公司集装箱系统的软件升级的变更工作。

3.1.2 系统调研

通常情况下，用户并不能准确把握系统调研的意义，也就是需求的含义，只是把想到的，通过直白的语言，叙述想要实现的那些功能，或者说用户是资料的储备和数据的来源。集装箱码头业务人员，他们有着丰富的工作实践经验，码头作业如同家常便饭，轻车熟路，而对于计算机这种工具而言，知之甚少。因此系统开发的项目想要把系统做好，应该具备良好的沟通能力与对集装箱业务的全面了解，能够从用户提供的种类繁多的信息中提取设计所需的内容，进行需求的归类 and 提炼，最终简明扼要的阐述出来，让用户理解和满意，才能为系统开发奠定坚实的基础。

3.2 业务流程

集装箱码头业务可以用“装卸船、集疏港”来概括，另外根据集装箱海运的流向和流程过程又可分为水来陆走（进口业务）、陆来水走（出口业务）、水来水走（中转业务）。综合以上业务划分的标准，按照集港、装船构成出口业务主线，卸船、疏港构成进口业务主线和两种业务都涉及衍生的中转业务。

在实现以上进口、出口、中转过程中衍生其他多项子业务，紧紧围绕这两种业务进行，穿插在进口、出口、中转业务过程中为其服务，我们统称为综合辅助业务。而以上业务又可划分为多种子业务。如图 3.1 所示。

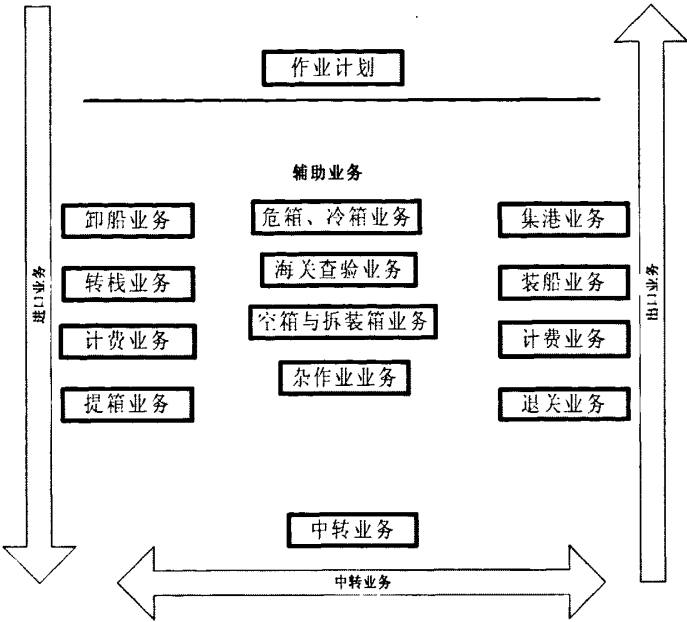


图 3.1 集装箱码头业务划分

Figure 3.1 Business of container terminal

3.2.1 进口卸船流程

(1) 进口业务的基本步骤

基本步骤是制定作业计划然后进行卸船作业，进而可能会有转栈业务，进行提箱前需要计费。如图 3.2 所示。

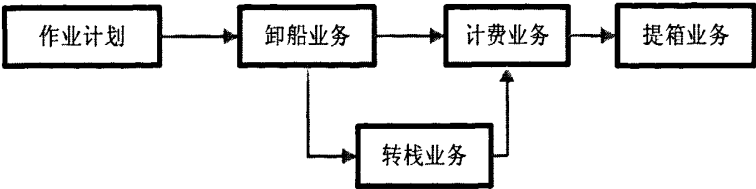


图 3.2 进口卸船基本步骤

Figure 3.2 Basic steps to import ship unloader

(2) 卸船业务简介

卸船业务是指按照预定计划把抵港的集装箱船舶上的集装箱从船上卸下的过程。具体涉及船舶靠泊计划、卸船计划、集装箱卸船场地计划和执行卸船作业。卸船的集装箱及集装箱的信息，经过信息传递和集装箱运输后，码头获得卸船集装箱信息，且集装箱经船舶运输转移到码头堆场或客户提箱拖车的。

(3) 卸船操作详细描述

① 确认船舶预报动态

泊位计划员通过调度系统，了解未来 24 小时本码头的船舶动态，联系船代或者船公司，对船舶的抵港时间进行核实确认，得到船舶本次作业的大概的进出口箱量，录入一定的计划信息数据，生成《昼夜作业计划表》，为码头作业提供指导性。

② 船舶报文信息导入

单证员在确定船期预报的情况下接收 EDI 船图和舱单文件，通过验证，查找报文中的错误信息，及时与船公司或者船代联系，进行更正，确认箱数、箱公司、尺寸、箱类型等信息无误后，最终导入船图和舱单。

③ 堆场策划

场地宏观计划员根据单证员导入的报文信息，制定出相应船舶的堆场计划。场控员制定计划的时候，必须依照集装箱的箱型、尺寸、流向以及提单号等因素来合理分配计划区域，保证系统自动给位的准确性。

④ 卸船作业

航运调度员协调作业安排和机械调度，装卸工拆加固，岸边理货人员，分配岸桥以及单桥的作业量，根据系统实时策划员做的卸船场地计划在系统中做区域计划，使场桥和堆场绑定，同时进行机械设备组的调配，安排港内拖车，并使设备组与岸桥进行绑定，生成作业队列。中控员（船控员）发送作业指令，通过消息服务器，拖车司机的车载终端会收到安桥下进行卸船的作业指令。当拖车司机行驶到目的地后，岸边理货员根据实际情况，把岸桥卸下来的箱子确认到拖车上。拖车的车载终端提示将集装箱拖至指定场位。拖车司机按照系统提示箱位并核对拖车箱信息，无误后，按照给出的指令，行驶到堆场给出的位置，场桥将集装箱

卸至指定位置并通过系统进行落场指令的确认，周而复始的往复循环，直到卸船作业完毕。

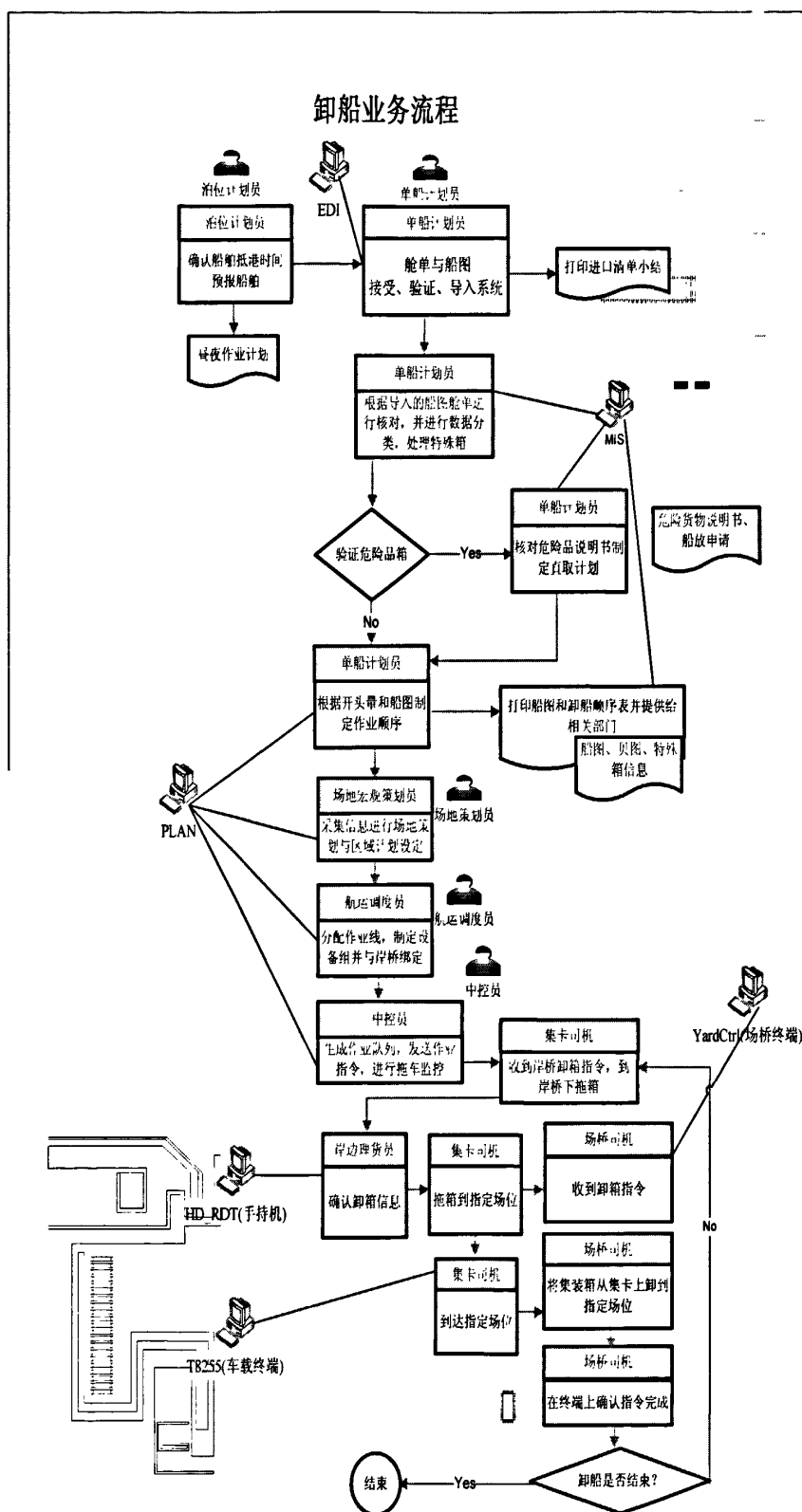
⑤ 卸船中特殊箱子处理

(a)溢短卸：溢短卸可分为溢卸和短卸。何为溢卸？溢卸的产生分两种情况，第一种岸桥卸下的箱子，船图和舱单信息不存在，场控员按照溢卸箱规则制定堆场计划。第二种岸边理货员输入箱号错误，系统提示箱信息不存在，没有与中控联系，强制进行了溢卸操作；什么是短卸？短卸是箱信息在船图舱单里可以找到，但是实际的到港箱确实没有匹配的箱信息。一般情况下短溢可以同时出现，但是不一定是完全的对称。

(b) 同船禁移箱（过境箱）转为翻捣箱。翻捣箱可以分为船上翻捣和舱外翻捣。舱内翻到目前不走系统的指令，维护好翻到计划后即可。舱外翻捣时，分两种情况，第一，需要制定相应的翻到箱的堆场测试，并实际卸船作业，在出口的时候，对该箱进行配载，并进行实际的装船作业。第二，如果提前能够晓得落地翻到箱的出口船上位置，则可以在翻到计划中录入改箱的船上目的位置，不需要通过指令进行真正的卸船作业。

(c)危险品箱：通常情况下，危险品箱不会进行落场操作，舱单船图数据导入以后，对危险品箱进行审批，由港外拖车进门后，理货员直接确认到港外拖车上，拉到港外的化工库。

(d)中转箱：集装箱虽然经过卸船操作，接卸到本码头，但是该箱子的目的港口非本港箱，或者在本码头内部进行中转(也可称为内转)，或者转到别的堆场、码头(也可称为集团内中转)。进口卸船详细业务流程如图 3.3 所示。



3.2.2 出口装船流程

(1)出口装船的基本步骤

如图 3.4 所示。

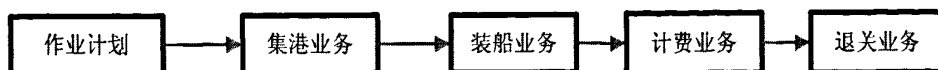


图 3.4 出口装船基本步骤
Figure 3.4 Basic steps to export shipment

(2)出口业务简介

装船指把集装箱转载到船上的过程。从计划到实施，中间关联多个岗位，共同协作完成装船业务。从装船准备的昼夜作业计划、单船配载计划、安排作业线，到作业过程中的指令发送、作业线装船作业，异常情况处理，一直到装船作业完成。在实际装船过程中有普通装船和直装两种情况。直装是指集装箱不经过堆场存放，直接从外部车队或港内车队装载到船上。一般危险品集装箱做直装业务

(3)装船作业详细描述

①船舶报文信息导入

单船计划员进行出口清单(NCL)或者订舱报文(BOOKING)的接收、验证、导入。

②堆场策划和场地收箱

堆场策划员进行集港场地的策划，并制定区域计划。闸口人员进行收箱操作。

③组织配载数据

收箱完毕后，单船计划员进行三单核对即：出口清单、场站收据、在场箱信息的核对。核对通过后，由单船计划员进行出口箱的配载。

④装船作业

配载完毕后，船控员安排作业岸桥、拖车，并制定设备组，使得拖车与岸桥进行作业绑定。中控或者值班调度组织进行装船指令的发送，开始装船作业，同时进行拖车指令的监控。通过消息机制，拖车终端会收到相应的到场地托箱的指令，拖车到达后，场桥司机将场地箱装载到拖车上以后，在场桥终端上进行确认，

拖车终端会显示到岸桥下装船的具体指令，拖车司机会按照指令到岸桥边进行装船，直到装船结束。

⑤装船中特殊箱子处理

(a)危险品直装箱：危管员进行危险品审批后，产生直装计划和委托，三单核对后，单船进行配载，待箱子进门口，再次进行三单核对，进行箱子的出口放行，然后进行正常的装船作业。

(b)落地到箱：进口舱外翻倒的箱，进行配载后，和正常的出口箱一样，也是通过发送指令后，进行装船。出口装船详细业务流程如图 3.5 所示。

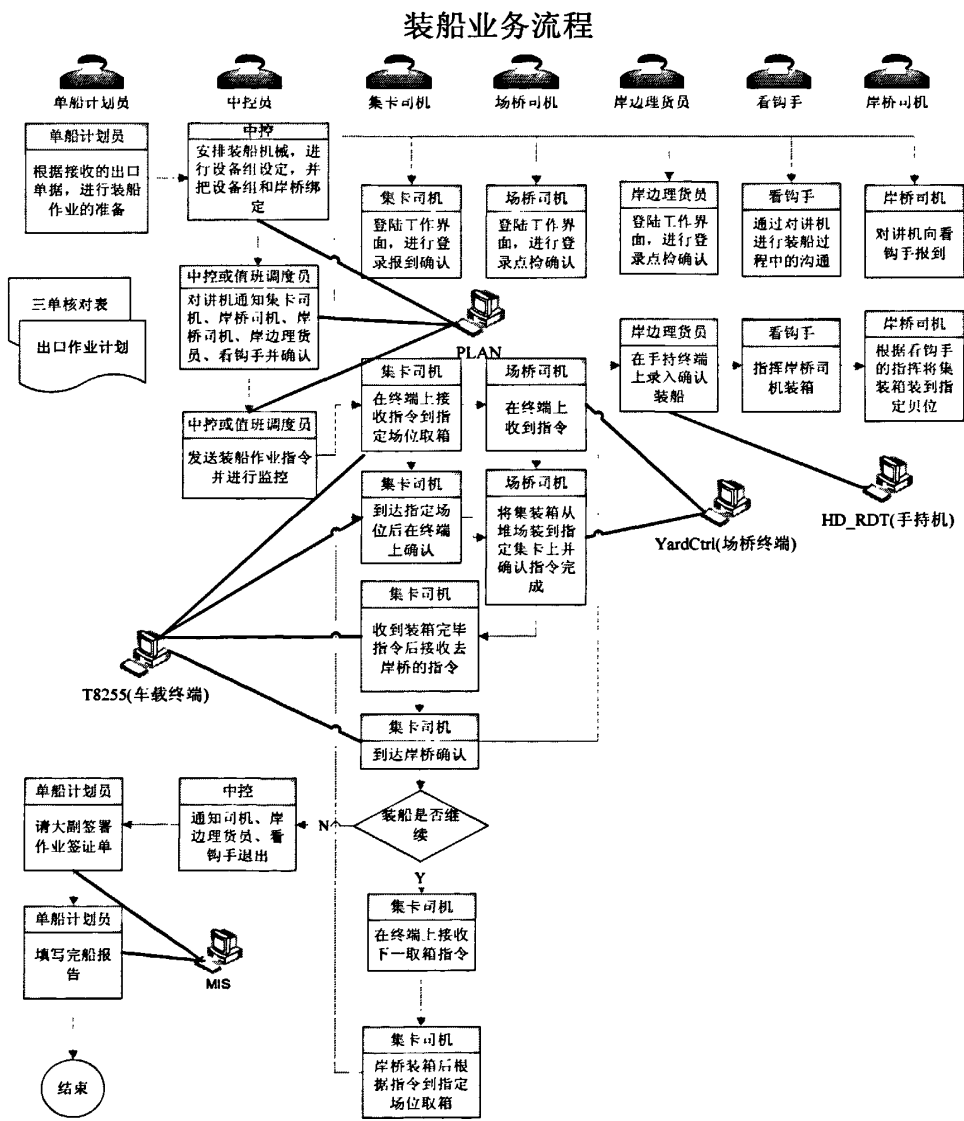


图 3.5 出口装船详细业务流程

Figure 3.5 Details of the export shipment business process

3.3 系统数据流例图

本节主要介绍 CTOS 数据用例。主要包括进口船图数据维护用例、出口配载用例图例。

3.3.1 进口船图数据维护用例

什么是船图？船图又叫“船舶积载图”，是对一个航次集装箱船舶的集装箱装载情况的说明文件。按照某一标准格式重点对集装箱的箱号、尺寸、装载位置等进行描述说明的一份电子数据文件，或者以图形方式展现的电子文件或打印出来的纸制文件。

单证员依据船公司所发进口船图资料，绘制船舶结构图，并进行 EDI 报文接收（或手工录入）、验证、导入，为卸船作业的提供依据；船图舱单导入完毕后，需要进行舱单与船图的核对，发现存在差异时，打出错误列表，及时联系船公司，进行更正，避免卸船开始后发现错误再进行不必要的更改。做好船图的处理工作以后，中控调度员才能进行卸船作业准备工作和之后的正常卸船作业。如图 3.6 所示，此用例满足单证员对进口船图数据进行增加、删除、修改、查询等操作的需求。如图 3.6 所示。

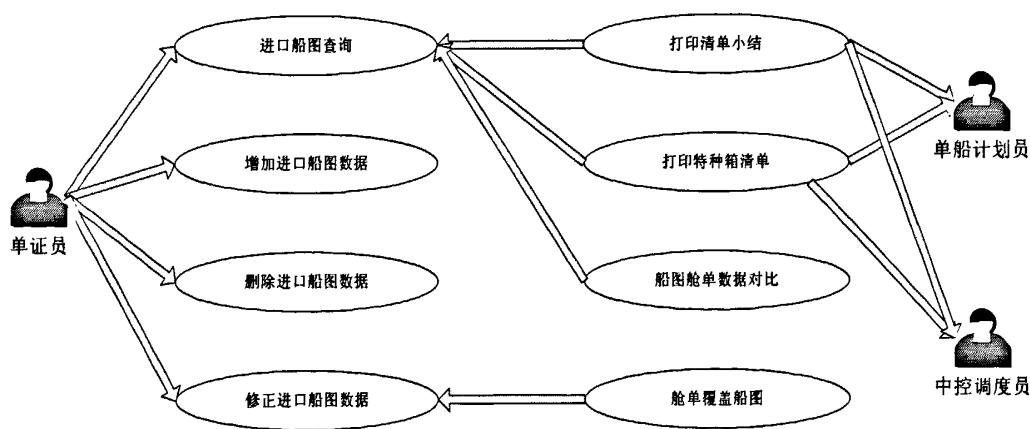


图 3.6 进口船图数据维护用例图
Figure 3.6 Import boats diagram data maintenance use case diagram

3.3.2 出口配载用例

单证员将船公司发来的订舱报文 (BOOKING) 或者出口清单 (NCL), 接收到系统中以后，单船计划员按照订舱报文和船舶配载指导图进行船舶预配，然后根据出口清

单中已经三单核对过的放行的箱子进行实配,然后将配载结果打印并生成 EDI 格式文件,与船方进行确认,如果有出入,要进行必要调整后,最终确定配载结果。

单船计划员根据集港船舶动态和收集、整理的单证资料,在装船前合理科学地编制船舶积载图和装船作业队列。如图 3.7 所示,对配载以及多人配载,不同岗位之间的协调交进行了情景再现。

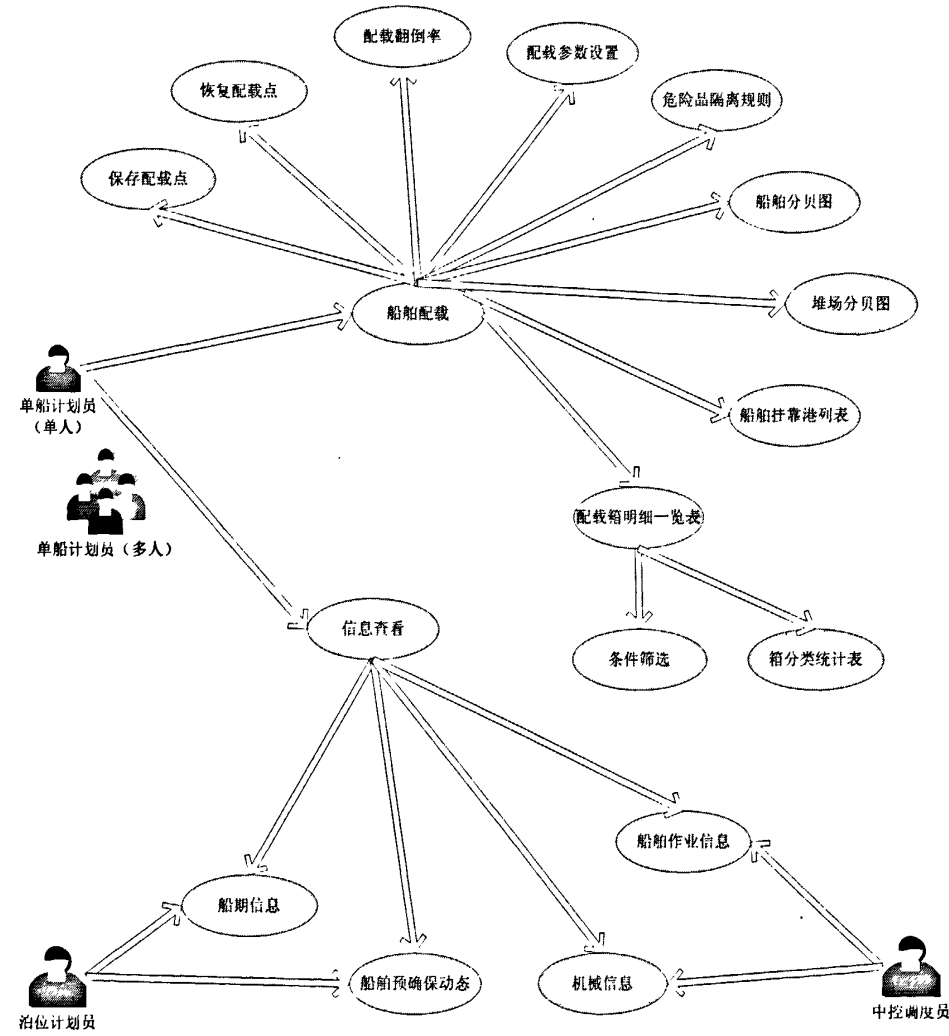


图 3.7 出口配载用例图
Figure 3.7 Use Case Diagram export loading

出口配载过程中有诸多因素要考虑,以下是几点注意:(1)保证船舶稳性,场地箱子的堆码规则一般是重压轻,配载根据取箱原则,船上按照重压轻原则进行配载。防止扭矩和横倾,船舶左右的重量注意要基本保证平衡。

- (2) 减少吃水差和纵向强度，前后重量应均匀合理分布。
- (3) 防止局部强度和绑扎出现问题，在船舶结构绘制的时候，对每列重量进行设置，限制堆积负荷。
- (4) 实际配载要以预配图为依据，特种箱应按照预配图中的位置进行配载。
- (5) 配载时，应考虑机械作业时候移动的合理性和有序性，尽可能避免作业冲突，从而降低了装船作业效率。
- (6) 作业线合理分配,减少场地翻倒作业。

3.4 系统数据流图

需求分析阶段，信息流是一个关键问题，通常用数据流程图描绘信息在系统中加工和流动的情况^[15]。

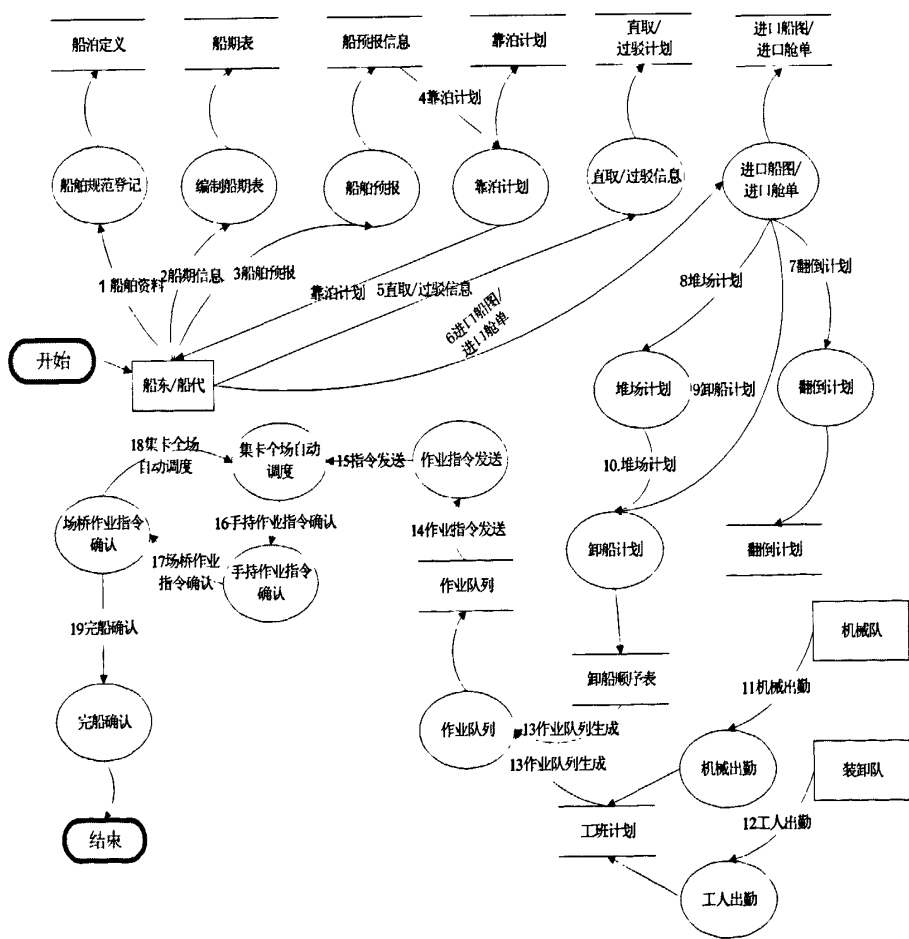


图 3.8 进口卸船作业数据流图

Figure 3.8 Unloading operations import data flow diagram

如图 3.8 所示,在需求初期的分析阶段,也可以成为系统的建模阶段,针对客户的需求,从满足用户从功能使用角度上出发,如何设计研发出能够操作简洁,贴近实际集装箱业务形态,这是问题的关键所在,也是系统设计成败的根基。数据流程图是以直观的图形清晰的描述系统中数据流动和数据变化的建模工具。如图 3.9 所示,为分析研讨后,汇总出的进口卸船作业数据流图及出口装船作业数据流图。

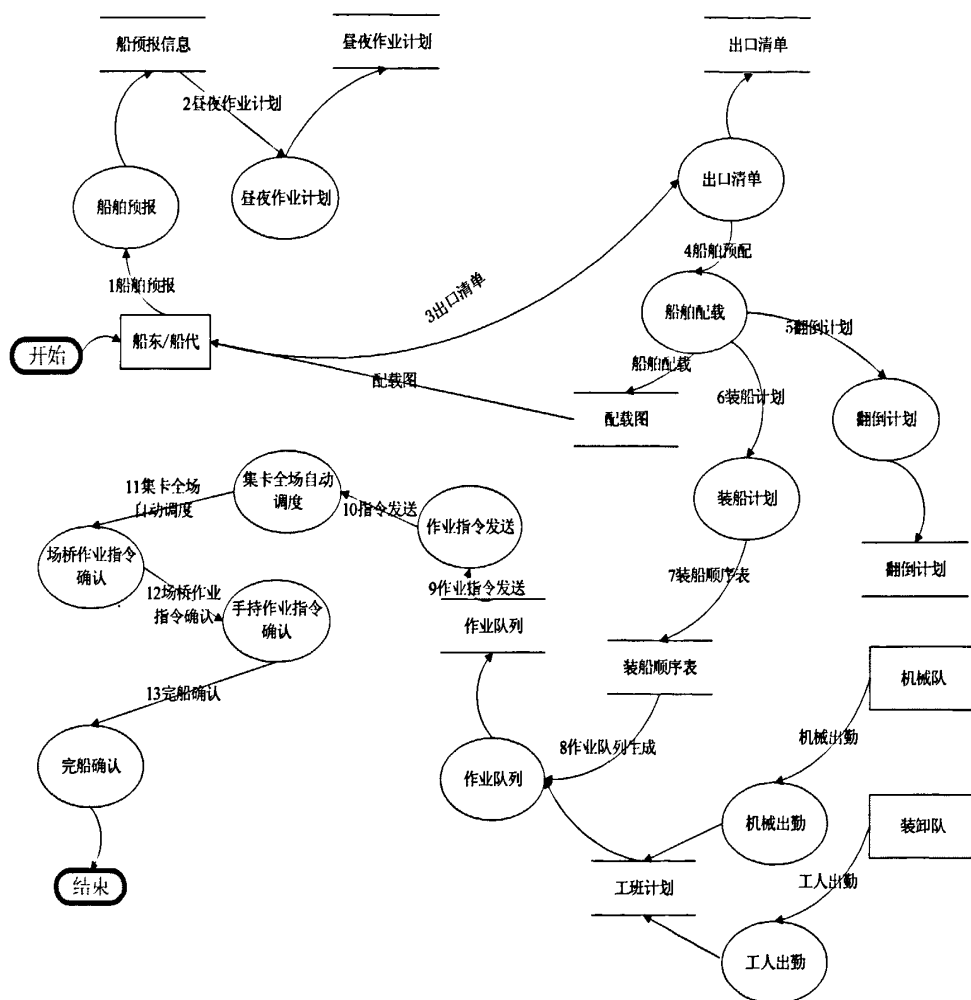


图 3.9 出口装船作业数据流图

Figure 3.9 Operating data flow diagram export shipment

第4章 系统设计

4.1 系统的设计目标

信息化带动传统产业的发展是大势所趋。近年来国内港口也纷纷通过合资的方式引进了国际上先进的集装箱计算机管理系统,并对此进行了不少改进和完善。天津港作为世界集装箱吞吐量海港,理应实现跨越式的发展。通过自动化的手段,力求在工艺创新、技术创新和管理创新上走在前列。研发适应集装箱物流发展的要求,功能和质量达到先进水平集装箱的生产作业管理的人性化操作系统,成为必然。

借着滨海新区开发开放的大好形势,为了推动天津港更好更快地发展,结合天津港各集装箱码头公司的业务需求及信息化整合的优势,本系统的目标务必建立在使集装箱码头作业更高质量、更高效、更稳定,稳步提高集装箱吞吐量业绩。本系统总体设计目标如下:全面提升集装箱业务人员的工作效率;有效减少集装箱码头信息化的人力成本与运行维护成本;建立健全完善的集装箱信息日志跟踪查询功能;使得集装箱码头公司的生产管理系统大致一体化,更好地推动天津港集团信息化整合的整体稳步前进。

4.2 系统设计原则

系统的研发需要遵循一定的开发设计的原则,以满足用户的客户化需要出发点,进行系统设计时,应该主要遵循以下设计原则:

(1) 客观易用性原则:系统设计建立在现有的客户化需求基础之上,从而明确系统应有规模,界定设备负载的能力范围。

(2) 安保性原则:CTOS 系统的设计采用 C/S 结构,通过网络进行承载,实现用户与数据库服务器之间的数据交换,因此安全是系统成败的关键。系统应该进行全方位的安全保护(网络层至应用层,切实跟进到多层),对于外部的入侵实时监测,通过加密机制防止越级操作,保证系统的正常运行。

(3) 可靠性原则:系统平台应能有效地避免由于单点故障引起的服务不可用,在设备、链路、数据层面上,应充分提供冗余备份,一方面最大限度的减少故障

的可能性，其次要保证系统平台能在最短的时间内恢复，最后要求在灾害发生的情况下的数据安全。

(4) 完备性原则：系统结构设计、系统硬件配置、系统管理方式等方面采用完备且普遍应用、实践验证过的成熟技术，而且不休具备行业的技术先进性。在技术的先进与成熟之间选取相对完善的结合点，才能做到系统的完备。

(5) 规范性原则：充分考虑到国际和国内外业界标准，研发设计采用的技术有理有据有节，设备定量且超过了设计所需标准，为与其它系统之间链接和日后系统升级做好的良好奠基。

(6) 扩展性原则：港口码头业务形态随着码头的发展，势必会有所变化，系统实际使用的硬件设备已经做出了预留，并在改良扩充模块后满足可预见的未来需求，保证现行系统设计完毕且正式运行后系统大规模升级时，用户的投资能得到最大限度的保护，达到设备应用的合理性和高效性。

(7) 可管理性原则：CTOS 系统的设备应便于管理，维护直观，操作简单，很容易上手，在设备、安全性、数据流量、性能等方面具备监视和控制，因此可以进行远程管理和故障诊断，保证系统良性循环与不间断运行^[16]。

4.3 系统设计逻辑结构与系统机构图

(1) 系统整体体系架构

系统按照多层标准，组件式应用的方式进行全面设计，因目前系统以企业内部应用为主的 C/S 结构系统，且以内部局域网业务应用为主，目前应用上按照逻辑上分为三层结构，分别为：表示层（presentation），领域层（do）和基础架构（infrastructure）层。因为基于组件式设计，我们在设计此架构时预留未来集中式或网络服务式应用的多层扩展基础。

表示层即用户应用程序接口，表示层实现向用户展现结果和处理用户和软件交互。主要职责就是为用户提供信息以及把用户指令翻译传送给领域层（业务处理）。负责基本的接收用户输入，数据合法性验证，操作提示及系统处理结果的展现。领域层以业务逻辑处理为主，表示层传入数据的业务逻辑及各类计算，其他组件实现系统的各种应用服务，在此与数据库持久层之间包含数据影射及数据

库实体层, 实现与持久层的交互。基础架构(infrastructure)层提供数据存取接口及复杂的高密度的数据处理及计算的复杂的业务逻辑的实现^[17]。

(2)图形化引擎组件

图形化应用是 CTOS 的核心技术, 对稳定性、运行效率要求及高, 另外要考虑维护难度、扩展能力和操作的方便性等等, 因此需要我们设计一套坚固的、能够灵敏的结合业务而又不受业务发展限制的图形组件, 故在领域层加入图形化引擎组件来提供未来更多图形化应用的需求。

(3)数据库适配器

CTOS 数据控制使用大量存储过程、函数及 SQL 语句, 我们需要一套独立的、可控的、统一的数据处理机制来保证稳定有序的控制数据流转。故在领域层加入独立的数据库适配器处理终端用户与数据库的交互及数据影射, 实现 SQL 及后台过程调用的集中层, 与表示层的分离各负其责, 层次分明。

(4)核心 UI 框架组件

集装箱系统涉及多个模块、多个部门, 因此系统需要一套完整的、稳定的、灵活的 UI 框架来支持各种用户的人机界面的个性化、美观度和使用方便性的要求。统一操作流程及方式, 形成良好系统 UI 体系, 故单独开发核心 UI 框架层, 支撑用户界面, 让表示层专注于用户交互处理。

(5)业务逻辑处理

引入业务逻辑处理组件, 一是在合理情况下充分发挥面向对象语言清晰组织和实现业务逻辑的优势(调试、维护修改方便), 同时减轻服务器压力, 逐渐形成标准的业务逻辑处理类库, 对未来系统的扩展、移植, 业务逻辑重用搭建框架。

(6)报表组件

鉴于港口业务各类单据复杂多样, 我们采用统一的方式处理提供各类报表服务, 统一报表库实现, 独立于界面, 层次清晰, 便于维护及扩展, 移植。如图 4.1 所示。

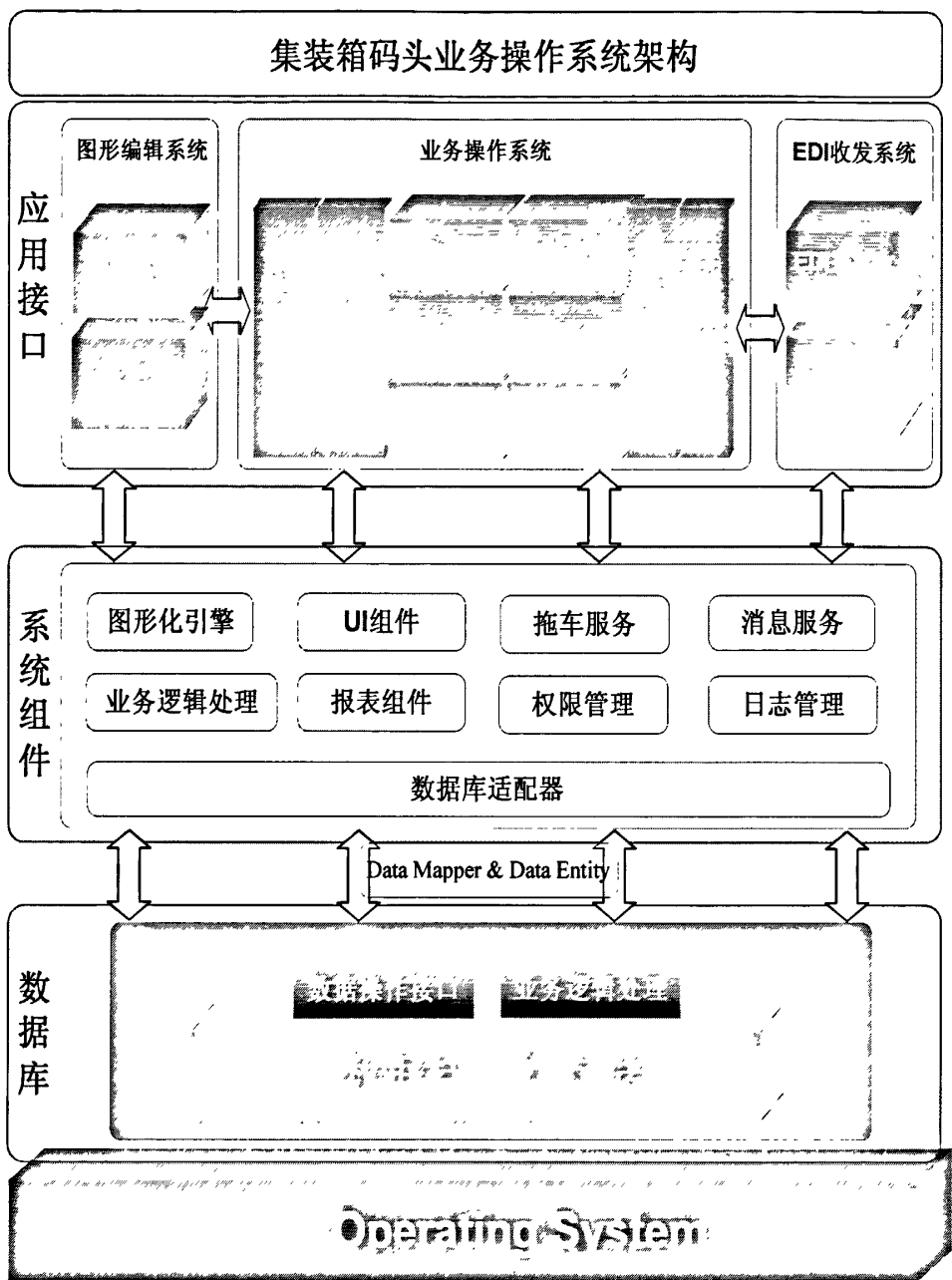


图 4.1 集装箱码头业务操作系统架构图

Figure 4.1 Container Terminal Operating System Chart

4.4 系统功能模块划分

系统功能模块划分的目的是要将复杂的，难以研究的大系统划分成若干个简单的易处理的子系统，降低软件系统的复杂性，提高可理解性、可维护性^[18]。系统功能模块遵循如下几点原则进行划分：子系统具有相对独立性；子系统之间的依赖性小；子系统之间的数据流向简明清晰；子系统的设置应更加灵活；子系统的应具有很强的可扩展性。CTOS 系统基于实时架构，以图形化操作方式为主，通过先进的控制平台和各种优化算法，实现了通过消息服务器动态实时跟踪传递、自动计划和监控、自定义报表随机组合、集卡调度等目标。系统设计根据模块划分为计划实时控制子系统（PLAN）、闸口子系统（GATECTRL）、基本管理子系统（MIS）和缓冲区管理子系统（GATEBUFFER），权限管理子系统（PRIVILEGE）、EDI（报文电子数据交换）子系统（EDI）、商务计费子系统（BIS）、统计子系统（SAS）和场桥作业子系统（Yardctrl），模块化的设计方法使 CTOS 更具扩展性与上升性。通过对泊位、堆场容量和码头机械设备等资源的控制，码头实践工作的形态与特性，适用于各种规模的集装箱码头。另外，采用消息服务，实现子系统之间数据与信息的传递，大大提高系统的实时性，实现了模块之间通过数据流保持紧密的联系，也使得子系统具有相对的独立性。开发一个软件产品是一项艰巨的任务，需要持续几个月甚至几年的开发周期，将这些工作划分为较小的部分或袖珍项目是切实可行的，每个袖珍项目都是一次能够产生一个增量的迭代过程^[19]。天津港集装箱码头作业管理系统涉及的业务形态和岗位功能划分很多，倘若偌大的一个系统一起实施，实现起来困难势必很多，我们将其按照一定原则划分为若干个子系统后，经由后台处理，使得表与表的之间的环环相扣，从而再把这些子系统捏合在一起，组成整个集装箱码头的生产管理系统，实现并行开发、统筹兼顾、相互关联，便于系统日后开发、运行和升级。如图 4.2 所示。

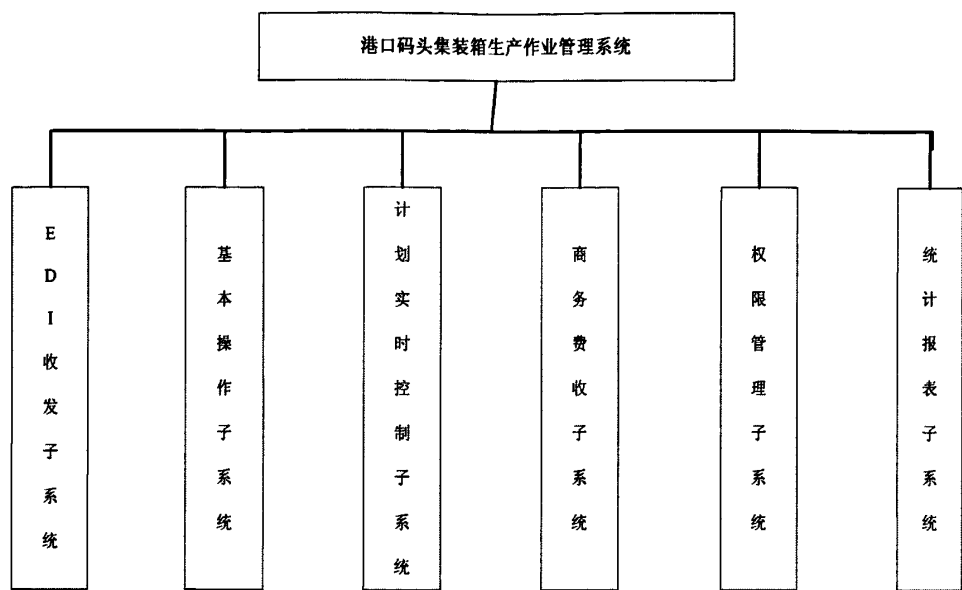


图 4.2 CTOS 系统功能模块图
Figure 4.2 CTOS system function block diagram

其中系统设计根据模块划分为计划实时控制子系统（PLAN）包括：闸口子系统（GATECTRL）、缓冲区管理子系统(GATEBUFFER)，场桥作业子系统(Yardctrl)。

4.5 系统功能模块设计

天津港是我国北方重要的国际港口和首都北京的海上门户，因此天津港的技术进步一直受到全国各港口同行的关注。港口改革和发展步伐的加快，集装箱码头的建设进入快速增长期, 仅天津港“十一五”期间将新建 21 个泊位，迫切需要先进的系统支撑。先进的信息技术正不断的应用到港口信息化建设上来，大幅度提高港口的生产经营和信息服务的水平。管理现代化、信息化已成为现代港口建设与发展的重要目标之一。目前，天津港员工对计算机的使用能力和操作能力良莠不齐，因此在功能模块设计阶段，做到以人为本，在满足业务需求的基础上，从利于业务人员操作、便于码头人员管理的角度出发，进行人性化的系统设计，建立统一、完备、灵活、简洁的人机交互界面，提高系统的稳定性和安全性，从而为码头生产作业起到保驾护航，进而使得天津港服务质量更上一层楼，企业服务形象全面提升。本系统主要就计划实时控制子系统、基本操作系统作为核心模

块进行讨论，其他子系统进行串联，码头生产操作是一个整体，应高照顾到方方面面。

4.5.1 EDI 收发子系统功能模块设计

报文是集装箱码头和船公司进行交流和沟通的桥梁，同时也是系统可靠性数据的来源。EDI 子系统功能主要是接收、验证并导入进口船图、舱单、出口清单等报文，EDI 子系统是系统的排头兵，是码头系统运行的基础，由此看来 EDI 子系统务必做到严谨、规范、统一、稳定。EDI 子系统可分为数据格式、数据处理、数据验证、数据应用及数据字典五大功能。其中数据格式为定义报文类型与格式，数据处理用来进行报文的接收与发送，数据验证进行报文导入到临时表后进行修改和验证、导入过程，数据应用时用来查询目前导入的结果，数据字典是定义报文发送的标准。

4.5.2 基本操作子系统功能模块设计

报文导入系统后，基本操作系统进行基础业务数据采集。该子系统可以划分为船舶管理、船舶单证管理、客户服务、堆场管理、系统管理五个功能模块。各个模块功能如下：

(1) 船舶管理:船期信息与船舶预确报信息采集与维护，制定各种业务计划，包括直取计划、直装计划、转栈计划、翻倒计划以及收货截单计划。

(2) 船舶单证管理:进口舱单、进口船图、出口清单数据查询，修改，增加删除。生成短溢报告、退关报告、单船小结、场站收据维护和三单验证维护。

(3) 客户服务:实现港内/港外查验办理，进口集装箱提箱委托办理，出口集装箱集港委托办理，集团内中转等集装箱陆运功能。

(4) 堆场管理:在场箱信息查询，折叠箱管理，危险品箱管理，强制修改箱信息，日办单滚动计划等功能。

(5) 系统管理:系统中船舶航线代码、港口代码集装箱代码、货类代码、统计代码，以及系统中一些功能性代码的维护。

4.5.3 计划实时控制子系统功能模块设计

作为 CTOS 系统的核心模块,改子系统能够完成集装箱码头作业的实时船舶作业监控、船舶配载、堆场计划和区域计划设定,拖车调度、多点情场等功能。子系统可以分为四个部分:

(1)船舶管理:选定船舶进行船图查看,分配桥吊及其作业任务,进行船舶作业测算,同时可以直观的进行船舶作业监控,图形化的查看船舶装卸情况等。

(2)堆场:绘制码头布局图编辑,在场机械查看,设定堆场计划,分航线设定重量等级,堆码规则设定,堆场找位失败跟踪查询等。

(3)集装箱:查询在场箱信息查询,查询移箱信息,修改场位,移箱申请等功能。

(4)配载:对出口集港箱进行船舶配载。

(5)资源:制定机械区域计划,E-RTG 转场计划,以及机械参数初始化设定。

(6)控制:实现装卸船作业任务的发送,收提箱作业任务的查看,堆场作业指令查看,设备组管理,拖车调度,拖车监控等功能实现。

(7)管理:设定堆场的作业规则,以及计划实时控制子系统权限控制。

4.5.4 商务计费子系统功能模块设计

集装箱码头作业中计费关系得到码头的收支运营情况,是码头正常运转的资金来源。商务计费子系统的功能是完成码头正常的进出口航运计费,进出口陆运计费、杂作业计费等码头日常收费管理。商务计费子系统主要模块包括客户关系,费用计算,发票管理,收费管理,实收统计查询五个功能模块。各个模块功能如下:

(1)客户关系:创建客户资料,建立客户航线协议,计费项目管理,服务项目维护等功能。

(2)费用计算:实现进出口船舶航运计费,出口陆运计费,进口提箱计费,退关查验计费,大件计费,发票手工录入等功能。

(3)发票管理:对发票进行规范管理,实现了发票的入库,发票领用,发票查询及作废功能。

(4)收费管理:主要功能包括航运大厅现场收费确认,发票收入确认,发票入账确认,退费管理等功能。

(5) 实收统计查询: 设计符合客户要求的报表, 为客户进行各种条件和各种形式的费用统计。

4.5.5 权限管理子系统功能模块设计

CTOS 系统涵盖了集装箱生产作业管理所有业务流程, 如此庞大的系统需要多人进行共同协作和管理, 因此, 必须通过权限对使用系统人员的岗位职责进行控制和划分, 权限管理子系统由此而生。该子系统包括系统管理、系统维护两个功能, 能够实现用户管理子系统用来进行部门、用户及岗位的建立, 登录密码的设置, 操作用户的权限分配, 岗位用户对应关系的对照及系统代码的维护。

4.5.6 统计报表查询子系统功能模块设计

进行 CTOS 系统数据综合查询和统计, 满足码头局报表和公司报表的客户化需求。该子系统可以分吞吐量统计、集疏运情况统计、船舶及泊位统计、库场统计、天津统计及系统管理等功能。

4.6 系统数据库设计

4.6.1 数据库设计

数据库设计是指对于一个给定的应用环境, 构造最优的数据库模式, 建立数据库及其应用系统, 使之能够有效地存储数据, 满足各种用户的应用需求(信息要求和处理要求)^[20]。在数据库领域内, 常常把使用数据库的各类系统统称为数据库应用系统。

数据库设计是指根据用户的需求, 在某一具体的数据库管理系统上, 设计数据库的结构和建立数据库的过程。数据库设计是建立数据库及其应用系统的技术, 是信息系统开发和建议中的核心技术。由于数据库应用系统的复杂性, 为了支持相关程序运行, 数据库设计就变得异常复杂, 因此最佳设计不可能一蹴而就, 而只能是一种“反复探寻, 逐步求精”的过程, 也就是规划和结构化数据库中的数据对象以及这些数据对象之间关系的过程。

4.6.2 数据库设计特点与要求

(1) 数据库设计特点: ①数据库建设是硬件、软件和干件的结合; ②三分技术, 七分管理, 十二分基础数据; ③技术与管理的界面称之为“干件”; ④数据库设计

应该与应用系统设计相结合;⑤结构(数据)设计:设计数据库框架或数据库结构;
⑥行为(处理)设计:设计应用程序、事务处理等;⑦结构和行为分离的设计

传统的软件工程忽视对应用中数据语义的分析和抽象,只要有可能就尽量推迟数据结构设计的决策早期的数据库设计致力于数据模型和建模方法研究,忽视了对行为的设计

(2) 数据库设计要求:

①分析用户的业务需求和数据的使用状况,掌握所用数据的种类、范围、数量以及在业务活动中交流的情况,确定用户对数据库系统的使用要求和各种约束条件等,形成用户需求规约。

② 形成独立于机器和各数据库管理系统产品的概念模式,用 E-R 图来描述。

③ 将 E-R 图转换成具体的数据库产品支持的数据模型,如关系模型,形成数据库逻辑模式。然后根据用户处理的要求,安全性的考虑,在基本表的基础上再建立必要的视图形成数据的外模式^[21]。

4.6.3 数据结构

CTOS 系统是一个完备的生产系统,有很多子系统组成,因此其数据结构比较庞大。其结构由基本代码表、常用数据表、大量触发器、存储过程、函数、过程包、JOB 事件组成。由几十个基本代码表、数百个数据表以及大量的触发器、存储过程、函数和过程包构成。代码表是系统统一使用的代码表,随着业务要求不断更新并保持稳定,使得各个子系统连贯统一,处理事务保持一致性。数据表则是生产操作而产生,例如报文导入、装卸船、计费、办单、提箱等,间接的反映了码头生产能力。基于系统的表和后台处理很庞大,以下简要的介绍几个常用表,这些表保存的数据,都是码头装卸船作业中实际产生的动态数据。如表 4.1 至 4.6 所示。

表 4.1 记录了集装箱来港以后的基本信息。

表 4.1 在港箱信息表 (PORT_CNTR)

Table 4.1 Box information table in Hong Kong

表名称	PORT_CNTR 在港箱信息表					
列名	类型	长度	小数位	可空	默认值	说明
CNTR_NO	System.Int32	22	0	N		箱编号
CNTR	VARCHAR2	12	0	N		箱号
CURRENT_STAT	CHAR	1	0	N	'1'	当前状态
CURRENT_STAT_COD	VARCHAR2	2	0	Y		当前动态码
CNTR_PLAC	VARCHAR2	12	0	Y		箱位

表 4.2 记录了船图报文导入后的信息。

表 4.2 船图数据 (SHIP_CNTR)

Table 4.2 Boat Map Data

表名称	SHIP_CNTR 船图数据					
列名	类型	长度	小数位	可空	默认值	说明
SHIP_NO	VARCHAR2	14	0	N		船编号
CNTR_NO	System.Int32	22	0	N		箱序号
CNTR	VARCHAR2	12	0	N		箱号
CNTR_CLASS	VARCHAR2	2	0	N		流向类别
CNTR_SIZ_TYP	VARCHAR2	4	0	Y		箱尺寸类型

表 4.3 记录了进口舱单报文导入后的信息。

表 4.3 提单箱 (BILL_CNTR)

Table 4.3 Bill of lading Box

表名称	BILL_CNTR 提单箱					
列名	类型	长度	小数位	可空	默认值	说明
SHIP_NO	VARCHAR2	14	0	N		船编号
BILL_NO	VARCHAR2	24	0	N		提单号
CNTR_NO	System.Int32	22	0	N		箱序号
CNTR_CLASS	VARCHAR2	2	0	Y		流向类别
CNTR	VARCHAR2	12	0	Y		箱号

表 4.4 记录船舶作业指令的动态信息。

表 4.4 作业指令表(SHIP_COMMAND)
Table 4.4 Operating Instruction List

表名称	SHIP_COMMAND 作业指令表					
列名	类型	长度	小数位	可空	默认值	说明
VOYAGE_NO	VARCHAR2	14	0	N		艘次号
UNLOAD_ID	CHAR	1	0	N		装卸标志
CNTR_NO	System.Int32	22	0	N		箱根
WORK_QUEUE_NO	VARCHAR2	30	0	N		队列组号
SHIP_MACH_NO	VARCHAR2	10	0	Y		岸桥号
TRUCK_NO	VARCHAR2	10	0	Y		拖车号

表 4.5 记录船舶作业队列信息。

表 4.5 作业队列表 (WORK_QUEUE)
Table 4.5 Operations team list

表名称	WORK_QUEUE 作业队列表					
列名	类型	长度	小数位	可空	默认值	说明
WORK_QUEUE_NO	VARCHAR2	30	0	N		队列组号
QUEUE_TYP	VARCHAR2	4	0	N		队列类型
CNTR_NO	System.Int32	22	0	N		箱根
SHIP_NO	VARCHAR2	14	0	Y		航次编号
TRAIN_NO	VARCHAR2	14	0	Y		火车编号
FUTURE_STAT	CHAR	1	0	N		

表 4.6 记录堆场作业指令的信息。

表 4.6 堆场作业指令表(CY_COMMAND)

Table 4.6 Yard work instruction sheet

表名称	CY_COMMAND 堆场作业指令表					
列名	类型	长度	小数位	可空	默认值	说明
CNTR_NO	System.Int32	22	0	N		箱根
WORK_TYP	VARCHAR2	2	0	N		作业类型
WORK_QUEUE_NO	VARCHAR2	30	0	N		队列号
VOYAGE_NO	VARCHAR2	14	0	Y		艘次编号
TRAIN_NO	VARCHAR2	14	0	Y		火车编号
MACH_NO	VARCHAR2	10	0	Y		机械号
TRUCK_NO	VARCHAR2	10	0	Y		拖车号
SHIP_MACH_NO	VARCHAR2	10	0	Y		岸桥号
USE_CNTR_ID	CHAR	1	0	Y		换箱标志
NEW_CNTR_NO	System.Int32	22	0	Y		新箱根
NEW_CNTR_PLAC	VARCHAR2	12	0	Y		新箱位

4.6.4 表与表之间关联

存储操作数据的数据表之间一般都存在关联，通常为了适应伸长操作的需要和满足使用系统的便捷性，在处理数据的时候通过触发器，存储过程和过程包等方式对前操作表的关联表进行操作，例如插入、删除或修改数据。船图报文导入系统后，数据存储在 SHIP_CNTR 表中，SHIP_CNTR 本身建立了触发器，通过 SHIP_CNTR 触发器向 PORT_CNTR 插入箱信息数据；在产生卸船队列时，系统会根据对应的箱量和船图位置，按照一定的原则，通过触发器队列信息数据插入 WORK_QUEUE。如图 4.3 所示。

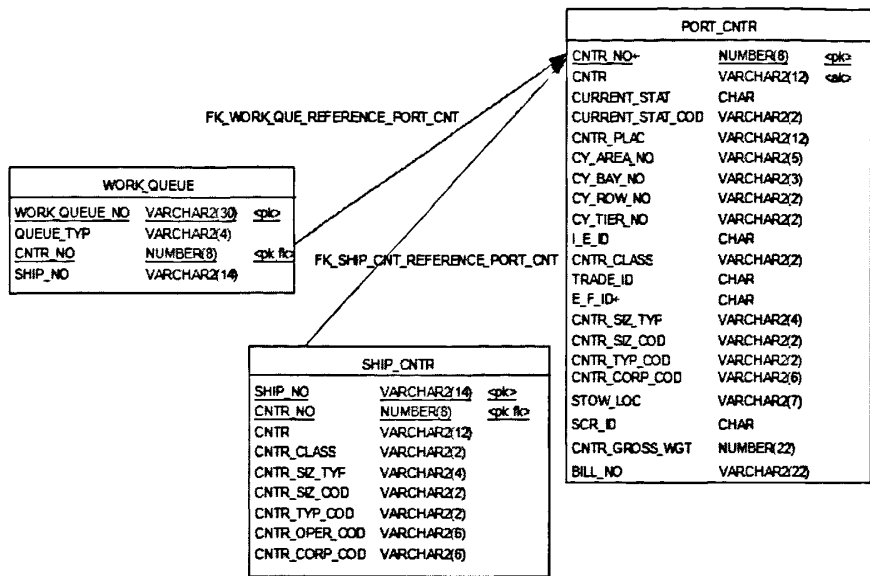


图 4.3 进口卸船数据表之间的关系

Figure 4.3 Container Terminal Operating System Chart

进口卸船发送指令后，WORK_QUEUE 表数据进行更新变为发送状态，SHIP_COMMAND 数据表通过后台处理产生数据，即为卸船指令，TRUCK_COMMAND 也会产生数据，即为拖车指令，通过设备组和岸桥的绑定，利用消息机制，拖车会收到到岸桥卸船的指，岸边手持确认后，CY_COMMAND 数据表产生数据，即场桥确认指令。场桥作业任务表及岸桥作业任务表收到指令数据，场桥作业完毕后通过触发器改变 PORT_CNTR 中箱状态标记为在场，完成卸船数据流程。如图 4.4 所示。

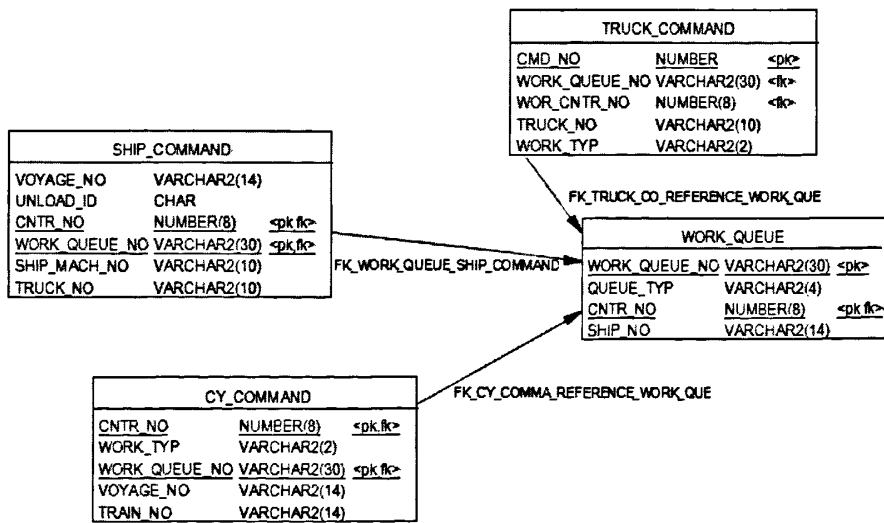


图 4.4 进口卸船指令数据表关系图

Figure 4.4 Import the relationship between discharge data table

各个子系统都是在后台经过处理产生了关联，其中：

基本操作子系统的主要表有：SHIP_CNTR（船图信息表）、SHIP_BILL（舱单基本信息表）、BILL_CARGO（舱单货物信息表）、BILL_CNTR（舱单箱明细信息表）、PORT_CNTR（堆场箱基本信息表）、M_WORK_PLAM（记录集装箱委托类型信息表）、SHIP_SHIFT_PLAN（翻倒计划表）、OUT_CNTR_PLAN（转栈计划表）、GATE_IO_PLAN（收货截单计划表）、EXP_CNTR_LIST（出口清单信息表）、EXP_CNTR_BILL（出口清单拼箱信息表）。

EDI收发子系统主要数据库表有：T_EDl_SHIP_CNTR（EDI接收船图信息表）、T_EDl_SHIP_BILL（EDI接收舱单头信息表）、T_EDl_BILL_CARGO（EDI接收舱单货物信息表）、T_EDl_BILL_CNTR（EDI接收舱单箱信息表）、T_EDl_EXP_CNTR_LIST（EDI接收出口清单信息表）、T_EDl_EXP_CNTR_BILL（EDI接收出口清单拼箱信息表）、T_EDl_EXP_BOOKING（EDI接收订舱信息表）

智能计划子系统主要数据库表：WORK_QUEUE（作业队列生成信息表）、PLAN_SHIP_CNTR（配载基础数据表）、SHIP_COMMAND（岸桥作业情况记录表）、SHIP_WORK_RECORD（岸桥作业情况历史信息记录表）、CY_COMMAND（场桥作业表）、CY_WORK_RECORD（场桥作业历史情况记录表）、TURCK_PLAC（拖车位置及状态表）、MACH_PLAC（场桥位置及状态表）、PLAN_GROUP（堆场计划表）、PLAN_RANGE（区域计划表）、PLAN_PLAC（作业机械当前位置信息表）、TRUCK_COMMAND（拖车指令表）、TRUCK_CNTR（大门拖车进出情况表）、BUFFER_TRUCK（缓冲区表）、BUFFER_TRUCK_CNTR（缓冲区箱信息表）、ERTG_MOVE_PLAN（转场计划表）。

以上三个子系统是本系统设计研发的核心模块。系统功能实现会计划实时控制系统和基本操作系统中的核心内容进行阐述。

第 5 章 系统主要功能实现

本文前面四章就 CTOS 系统的对集装箱码头作业管理系统的研发背景、开发系统用到的理论知识、系统的总体设计，功能模块划、数据库设计等进行了比较详细的介绍，第五章将对本章将介绍系统主要功能的实现。

5.1 系统卸船作业功能的实现

5.1.1 船舶动态维护

卸船功能涉及到大量的数据排序，有效的排序算法对于经济性而言相当重要^[22]。首先，码头泊位计划员通过集团业务处，参照集团调度系统，对局业务处船舶动态的划拨进行查询，经与集团确认后，按照时间的顺序，进行船舶预确报信息的维护。这是码头船舶作业计划的重要依据和前提。如图 5.1 所示，维护北欧亚超库，进口航次 201N, 出口航次 202S 的船期预确报信息。

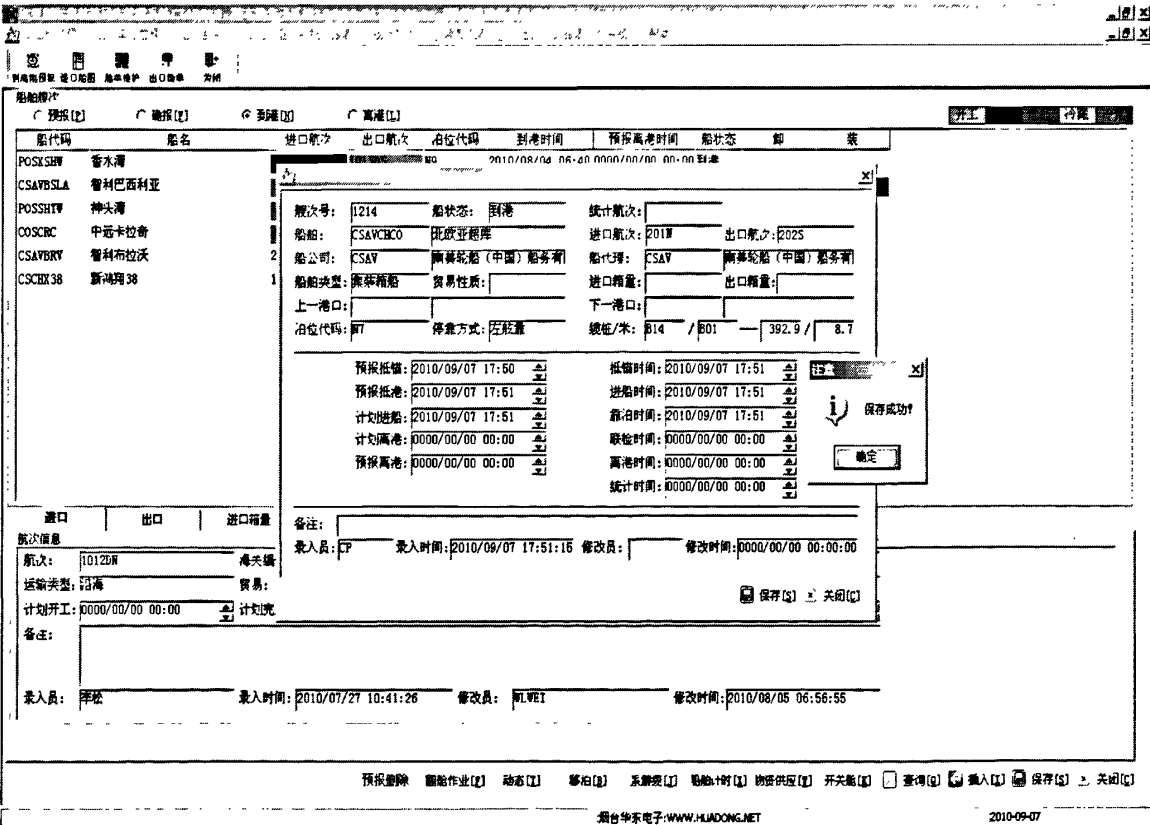


图 5.1 船舶动态维护界面

Figure 5.1 Ship dynamic maintenance interface

需要注意的是每条船舶都必须维护船舶的贸易性质以及航线，这是船舶完船离港口船舶计费的重要依据，另外对于船舶出口航次的挂靠港也要进行维护，因为无论是出口收箱集港，还是船舶配载，都要以港口为出发点。

5.1.2 EDI 报文导入——进口船图

船图一般可以准确说明集装箱在船上的具体位置，即贝位，对于船舶的卸船作业起指导作用，也是码头安排作业机械的主要依据。

5.1.3 EDI 报文导入——进口舱单

舱单办理船舶出(进)口报关手续的单证。经船长签字的载货清单送海关，作为办理船舶出(进)口报关手续的依据。海关凭此验货放行。船舶离港时，还需随带若干份清单，以备船舶中途挂港或驶抵卸货港时办理进口报关手续之用；船舶载运所列货物的证明。载货清单所列货物必须与船舶实际载运货物一致。如果船舶未装货出口，也需填报无货出口的载货清单。舱单是业务联系的单证，载货清单的留底，常用作承运人在装货港的代理人拍发开航货载电报的依据；也是向船长和船公司或卸货港的代理发出更正通知的依据。当承运人在卸货港的代理人尚未收到邮寄的货运资料时，也可复制随船携带的载货清单，用以作为安排泊位、卸货和货物进出库场的依据。

5.1.4 进口船图数据维护

EDI 进行报文导入以后，进入基本操作子系统可以进行导入的船图数据具体查询，如果出现问题，可以及时与船公司或者船代联系，进行报文的更正。

5.1.5 舱单数据维护

与进口船图一样，EDI 导入舱单后，也可以进行数据的查询修改，别且也可以和船图进行校验，在正式卸船作业开始以前，保证集装箱信息的准确性，避免后续提箱业务因为舱单的问题而出现差错，影响码头正常的计费提箱作业。

5.1.6 船舶作业线分配

基础数据准备完毕后，可以进行船舶作业线的分配，按照箱量的大小和船图的分布顺序，进行合理的岸桥作业数据的分配，然后可以生成卸船的作业队列，为接下来的卸船作业做最后的准备，如图 5.2 和 5.3 所示。

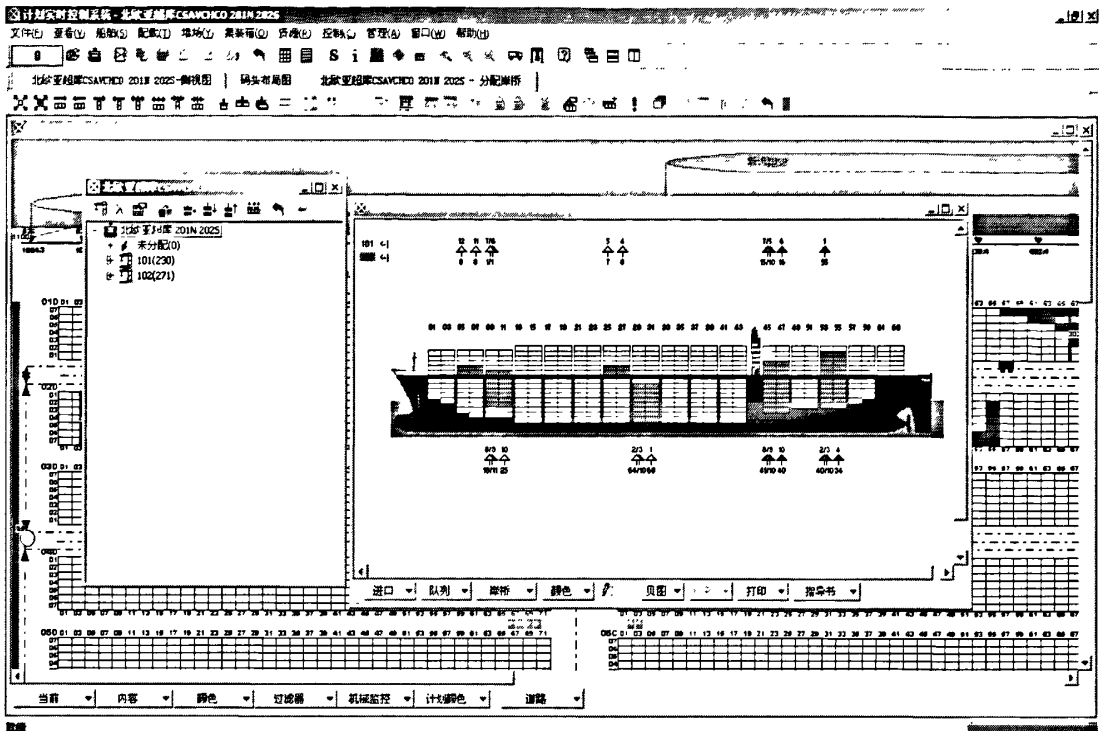


图 5.2 分配作业线及作业量操作界面

Figure 5.2 The amount of homework, route planning and operation interface

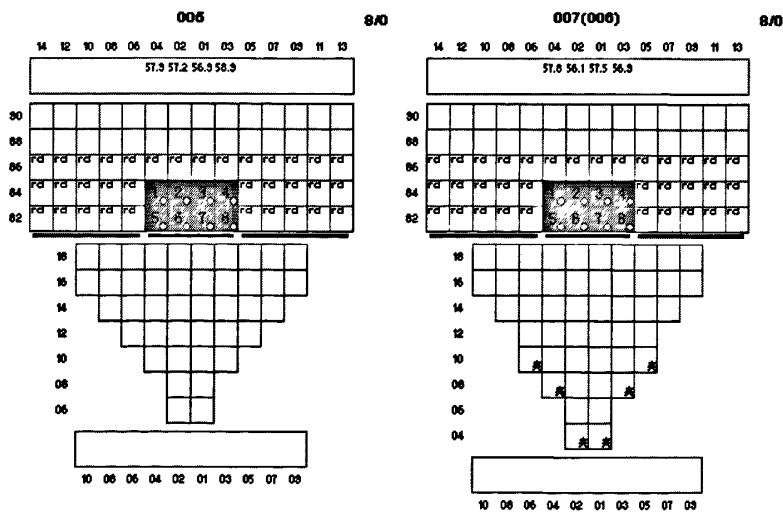


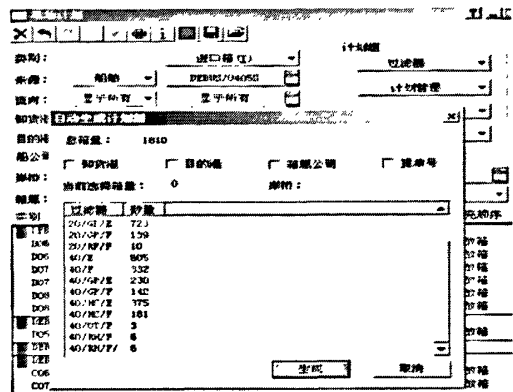
图 5.3 生成作业队列的操作界面

Figure 5.3 Generate job queue interface

5.1.7 堆场计划的制定

卸船工作之前,场地宏观计划员要根据 EDI 导入的进口箱量做宏观堆场计划,场地实时计划员根据类别、流向、卸货港、箱型、重量范围、场桥类型(单钩或双钩)和作业效率制作堆场计划。堆场计划设定的过滤器可以通过导入的船图和舱单进行设定。如图 5.4 所示。

自动生成堆场计划



当系统通过EDI接入进口船图或出口清单(订舱报文)后,运用堆场策划的自动生成计划组功能,可快速生成场地计划。此功能支持内外贸自动分关、大票单独堆码、分作业线的堆场策划、分箱公司的堆场策划

图 5.4 堆场计划自动生成
Figure 5.4 Yard plan automatically

如图 5.5 所示,场地宏观计划员做完宏观堆场计划之后,场地实时计划员在此基础上细化制作堆场计划。

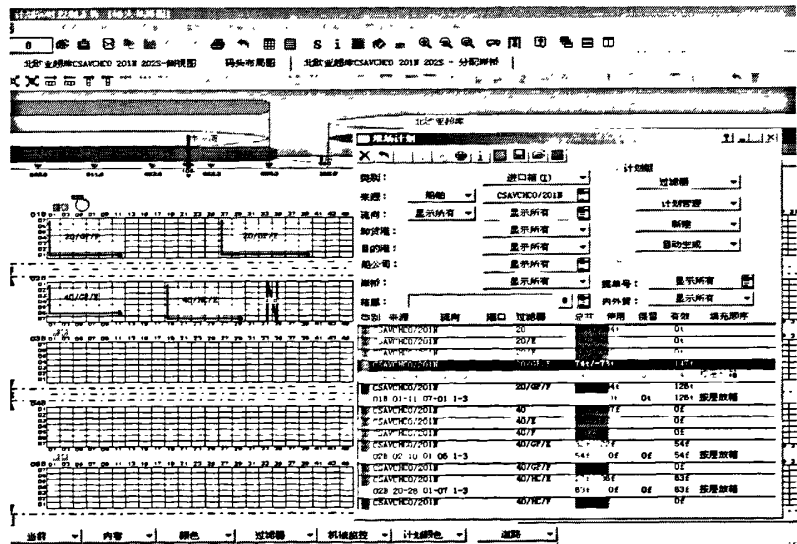


图 5.5 堆场计划设定界面
Figure 5.5 Yard plans to set interface

5.1.8 机械区域计划的设定

卸船堆场计划设定好以后，我们需要对场桥作业的区域进行设定，岸边理货员进行手持确认 P 箱后，需要通过场桥进行落场确认，而每一台场桥作业的区域，可以通过区域计划做灵活的控制，这样不仅可也使得作业有序，避免了同一条作业指令两台机械同时看到而争抢确认的问题，提高了作业效率，使得后续的场桥司机计件统计一目了然。如图 5.6 所示，区域计划覆盖在堆场计划的上层，而且是透明的，可以同时查看区域计划和堆场计划，也可以看到箱子的码放位置。如图 5.6 所示。

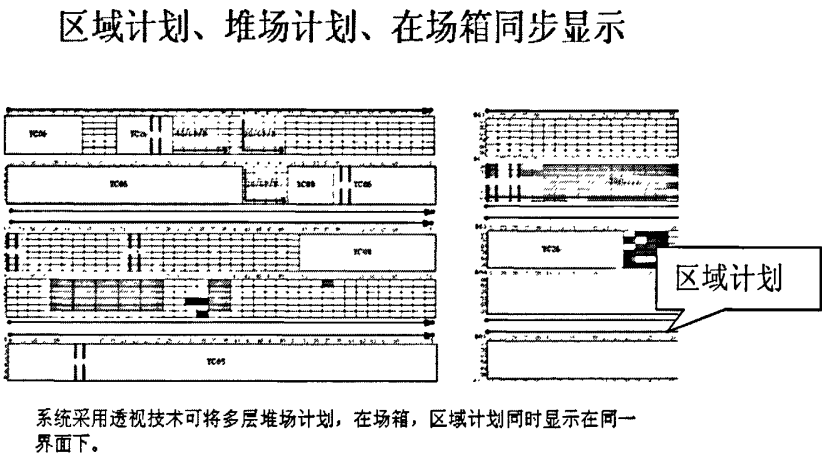


图 5.6 区域计划设定
Figure 5.6 Regional plans set

5.1.9 卸船作业指令发送

中控员确认做好了卸船的准备工作以后就可以按照一定顺序，进行卸船指令的发送。如图 5.7 所示。

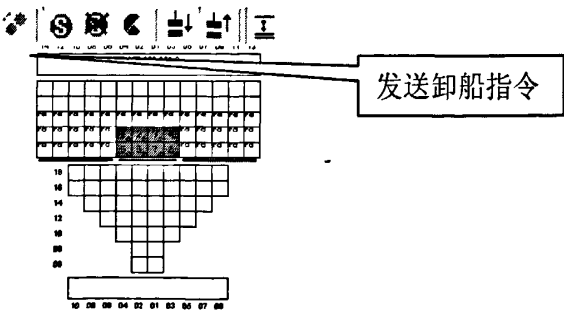


图 5.7 发送卸船指令
Figure 5.7 Send unloading instruct

5.1.10 岸边理货员手持卸船操作

在接到卸船通知后，岸桥司机按照船上看勾手的指挥，进行卸船作业，岸边理货员检查箱信息，如果发现残损，进入手持残损录入界面，维护残损信息；若是打捆箱，岸边理货员则执行打捆箱处理流程最后执行箱号确认操作；若是溢短箱，岸边理货员及时联系中控，核对确认后执行溢短箱处理流程最后执行箱号确认操作；正常情况下，理货员可以输入箱号，进行正常的手持确认，确认到拖车。同时手持也可以支持双箱工艺的作业，卸船时同时进行双箱确认，如图 5.8 所示。

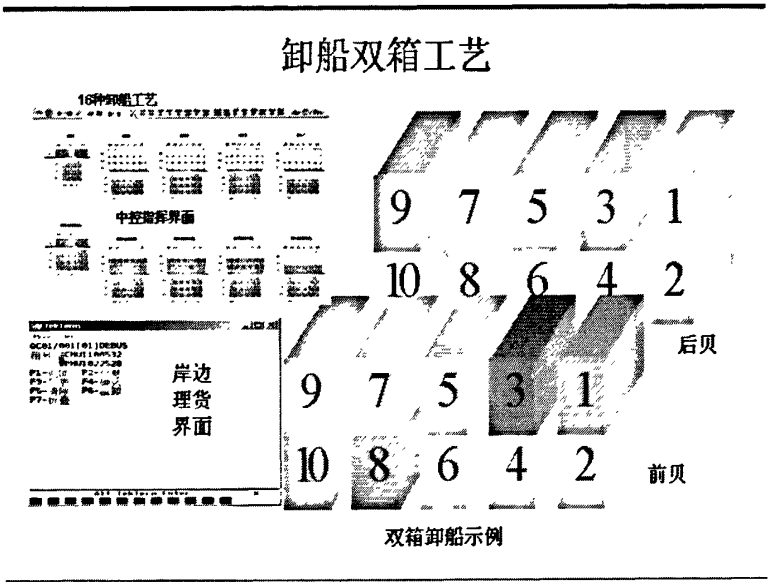


图 5.8 卸船双箱工艺
Figure 5.8 Dual Box Technology unloading

5.1.11 集卡全场调度

卸船作业需要各个岗位工作协作，因此在整个卸船过程中，对集卡拖车的跑到观测和督导必不可少。集卡调度提高了码头拖车的利用率，方法是估计每辆可用的拖车的最优的场地工作分派，然后自动将拖车分派到相应的码头起重机装卸货柜。完成同样的工作只需要较少的拖车，从而降低了您的基本建设投资和用于人员、燃料和维护的可变成本。更高的利用率和更低的成本使您能够以更少的花费完成更多的任务，从而获得丰厚的 IT 投资回报。目前集卡调度在装卸船过程

中正发挥着重要作用，指示着集卡司机准确及时跑位，减少了中控指挥强度，秩序了码头生产，提高了码头效率。如图 5.9 所示。

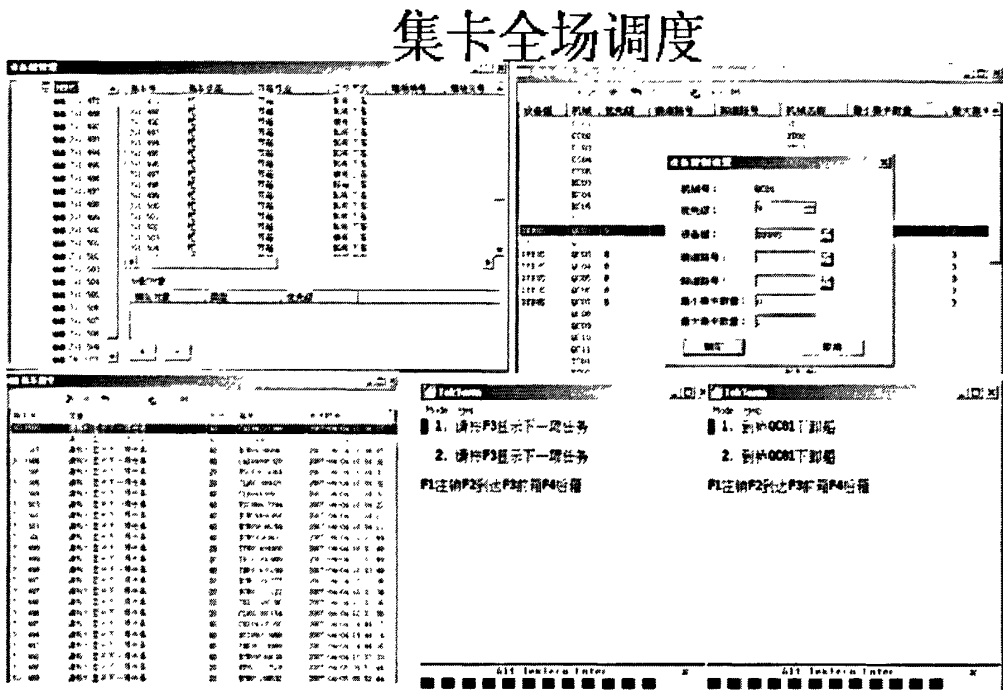


图 5.9 集卡调度图示
Figure 5.9 A scheduling icon

5.1.12 场桥落场确认

岸边理货员手持确认后，集卡拖车根据车载终端指令，拖箱到指定场地，由场桥司机进行箱子的落场确认，落场确认后，如果该集卡拖车绑定的设备组依然能够收到卸船指令，拖车会继续收到岸桥下继续卸船的消息，从而循环往复，一直到卸船结束。

5.1.13 系统卸船回退功能的实现

卸船作业中由于岸边理货员的误操作，或者确实需要实现卸船作业的逆过程，就必须进行进口箱的退卸。退卸可以是集装箱回到最初的状态，也就是回到指令发送未作业状态，删除数据库中曾经进行的卸船作业的所有操作记录，退卸完毕后，生成一条装船队列，可以根据实际情况进行装船作业，其核心代码如下：

```
IF NVL (wl_work_order, 'N') = 'N' THEN
```


--不产生指令

p_port_cntr_reset (wl_cntr_no, '1');--修改来港箱状态

p_handset_reset (wl_cntr_no, wl_current_stat);

delete from ship_command where cntr_no=wl_cntr_no and unload_id='.';

--清空手持确认操作信息

p_stat_reset (wl_ship_no, wl_cntr_no);--清动态

ELSE

IF wl_current_stat = '4' THEN --箱子在车

--更新PLAN_SHIP_CNTR

UPDATE PLAN_SHIP_CNTR SET I_SHIP_NO=NULL, E_SHIP_NO=WL_SHIP_NO

WHERE I_SHIP_NO = WL_SHIP_NO AND (cntr_no = wl_cntr_no or

cntr_no in(select cntr_no from port_cntr where fold_cntr_no=wl_fold_cntr_no));

--更新队列，将卸船队列更新成装船队列

DELETE work_queue WHERE (cntr_no = wl_cntr_no or cntr_no in(select cntr_no from
port_cntr where fold_cntr_no=wl_fold_cntr_no)) AND

queue_typ = 'SO';

UPDATE work_queue SET comm_status = '2', future_stat = '5',

future_plac = wl_cntr_plac, queue_typ = 'SO', ship_mach_no = wl_ship_mach_no, truck_no =

wl_truck_no, rec_nam = wl_oper_cod

WHERE (cntr_no = wl_cntr_no or cntr_no in(select cntr_no from port_cntr where
fold_cntr_no=wl_fold_cntr_no)) AND queue_typ = 'SI';

--发送装船指令

BEGIN

SELECT work_queue_no INTO wl_work_queue_no FROM work_queue

WHERE cntr_no = wl_cntr_no AND queue_typ = 'SO';

SELECT voyage_no, ship_cod INTO wl_voyage_no, wl_ship_cod

FROM ship_voyage WHERE i_ship_no = wl_ship_no;

SELECT COUNT (0) INTO wl_count FROM ship_plan

WHERE voyage_no = wl_voyage_no AND unload_id = '+'

AND ship_mach_no = wl_ship_mach_no AND work_queue_no = wl_work_queue_no;

IF wl_count = 0 THEN

INSERT INTO ship_plan (deck_id, bay_no, voyage_no, unload_id,

ship_mach_no, ship_work_seq, work_scheme, work_queue_no)

(SELECT deck_id, bay_no, voyage_no, '+', wl_ship_mach_no, ship_work_seq, work_scheme,
wl_work_queue_no FROM ship_plan

WHERE voyage_no = wl_voyage_no AND unload_id = '-' AND work_queue_no

```

        = wl_work_queue_no);
    END IF;
DELETE ship_command WHERE work_queue_no = wl_work_queue_no
AND (cntr_no = wl_cntr_no or cntr_no in(select cntr_no from port_cntr
where fold_cntr_no=wl_fold_cntr_no));
INSERT INTO ship_command (work_queue_no, voyage_no, unload_id, cntr_no, ship_mach_no,
truck_no, send_tim, send_nam, work_tim, work_nam, emp_no) VALUES (wl_work_queue_no,
wl_voyage_no, '+', wl_cntr_no,
wl_ship_mach_no, wl_truck_no, SYSDATE, wl_oper_cod, NULL, NULL, NULL);
EXCEPTION
    WHEN OTHERS
    THEN
        wl_return := SQLERRM;
        ROLLBACK;
        RETURN;
    END;
END IF;
END IF;

```

5.1.14 系统卸船找位算法的实现

卸船找位是本系统的核心算法之一，遵循节约堆场利用率，方便作业的原则。

下面对卸船找位算法的重点内容进行说明：

1) 卸船找位后台存储过程核心代码如下：

```

    IF VR_CNTR_PTY(1).MSG_COD<>'99' AND WG_X1='1' AND VS_FIND_ID='3' THEN
--找位扩展1 (大门滑块找位)
        IF VR_CNTR_PTY(1).MSG_COD='20' AND      --有双吊计划,但找位失败
            WG_DOUBLE_ID='1' AND                --双箱标志
            VR_CNTR_PTY(1).PLAC_F IS NULL AND    --找位失败 前箱
            VR_CNTR_PTY(1).PLAC_L IS NULL THEN  --找位失败 后箱
            --分拆调用(按单箱)前箱
            VR_CNTR_PTY(1).CNTR_F:=VS_CNTR_F;
            VR_CNTR_PTY(1).CNTR_L:=NULL;
            VR_CNTR_PTY(1).MSG_COD:=NULL;
            VR_CNTR_PTY(1).MSG_TXT:=NULL;
            --前箱找位调用,获得位置及错误号

```

```

        P_GET_PLAC(VR_CNTR_PTY,          VR_PLAN_GRP,          VR_WORK_PTY,
VR_SPEC_PTY);
        WL_PLAC_F:=VR_CNTR_PTY(1).PLAC_F;
        WL_MSG_COD:=VR_CNTR_PTY(1).MSG_COD;
        --分拆调用(按单箱)后箱
        VR_CNTR_PTY(1).CNTR_F:=VS_CNTR_L;
        VR_CNTR_PTY(1).CNTR_L:=NULL;
        VR_CNTR_PTY(1).MSG_COD:=NULL;
        VR_CNTR_PTY(1).MSG_TXT:=NULL;
        --后箱找位调用,获得位置及错误号
        P_GET_PLAC(VR_CNTR_PTY,          VR_PLAN_GRP,          VR_WORK_PTY,
VR_SPEC_PTY);
        WL_PLAC_L:=VR_CNTR_PTY(1).PLAC_F;
        IF NVL(WL_MSG_COD,'00')<NVL(VR_CNTR_PTY(1).MSG_COD,'00') THEN
            VR_CNTR_PTY(1).MSG_COD:=WL_MSG_COD;
        END IF;
        VR_CNTR_PTY(1).PLAC_F:=WL_PLAC_F;
        VR_CNTR_PTY(1).PLAC_L:=WL_PLAC_L;
        END IF;
        if VS_CNTR_F  is not null and VS_CNTR_L is not null then
            if VR_CNTR_PTY(1).PLAC_L IS NULL then
                VS_MSG_COD:='02';
                VS_MSG_TXT:=VS_CNTR_L||'没有堆场计划';
                return;
            elsif VR_CNTR_PTY(1).PLAC_F IS NULL then
                VS_MSG_COD:='02';
                VS_MSG_TXT:=VS_CNTR_F||'没有堆场计划';
                return;
            end if;
        end if;
        IF
        F_CHECK_DUP_PLAC_QUEUE(VR_CNTR_PTY(1).PLAC_F,VR_CNTR_PTY(1).PLAC_L)='
0' THEN --无效箱位
            VS_MSG_COD:='11';
            VS_MSG_TXT:=返回箱位在队列中已存在, 请重试;
            --插入队列箱位重复, 重新找位日志

```

```

INSERT INTO T_FIND_LOG (CNTR, MSG_COD, MSG_TXT, REC_TIM)
VALUES (NULL,
        '11',
        'P_GET_PLAC- 返回箱位在队列中已存在，请重试
        '||'/'||VS_CNTR_F||'+'||VR_CNTR_PTY(1).PLAC_F||'--'||VS_CNTR_L||'+'||VR_CNTR_PTY(1).PL
        AC_L,SYSDATE);

ELSIF VR_CNTR_PTY(1).PLAC_F=VR_CNTR_PTY(1).PLAC_L AND --前后箱位重复
sVR_CNTR_PTY(1).PLAC_F IS NOT NULL AND
        VR_CNTR_PTY(1).PLAC_L IS NOT NULL THEN
        VS_MSG_COD:='12';
        VS_MSG_TXT:='双箱找位返回相同箱位，请按单箱方式找位';
        --插入队列箱位重复，重新找位日志
        INSERT INTO T_FIND_LOG (CNTR, MSG_COD, MSG_TXT, REC_TIM)
        VALUES (NULL,'12',
        'P_GET_PLAC- 双箱找位返回相同箱位，请按单箱方式找位
        '||'/'||VS_CNTR_F||'+'||VR_CNTR_PTY(1).PLAC_F||'--'||VS_CNTR_L||'+'||VR_CNTR_PTY(1).PL
        AC_L,SYSDATE);

ELSE
        --队列中无重复箱位，找位结果返回
        IF VR_CNTR_PTY(1).MSG_COD='99' AND      --有双吊计划,但找位失败
        WG_DOUBLE_ID='1' AND
        WL_double1='1' AND
        WL_double2='1' AND      --双箱标志
        VR_CNTR_PTY(1).PLAC_F IS not NULL AND  --找位失败 前箱
        VR_CNTR_PTY(1).PLAC_L IS not NULL
        and VR_CNTR_PTY(1).PLAC_F < VR_CNTR_PTY(1).PLAC_L THEN  --找位失败
        后箱。卸船找位算法分析如图 5.10 所示。

```

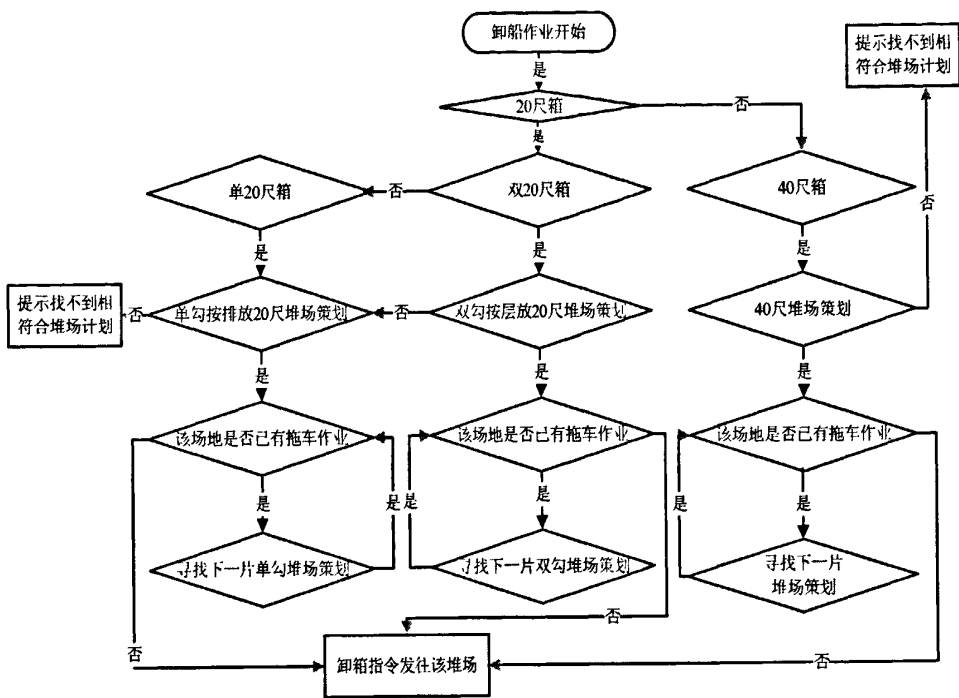


图 5.10 卸船找位算法分析

Figure 5.10 Algorithm to find bits of unloading

5.2 系统装船作业功能的实现

5.2.1 出口装载清单的导入

码头的集装箱生产系统接收完装载清单，会自动生成报文内所有集装箱的集港计划，从而为闸口收箱人员提供集装箱的船名航次、提单号、箱号、箱型、尺寸、空重、箱重、卸货港、特种箱描述等详细信息。但是一般来讲，出口装载清单(NCL)有一定的时间延迟，通常也可以进行订舱报文的导入，订舱报文中箱号为空，其主要意义就是为了跟上码头作业的进行，集港收箱的时候一般使用的是订舱报文

5.2.2 出口装载清单维护

报文接收完毕后，可以在基本操作系统里进行出口装载清单查询修改。

5.2.3 出口堆场计划设定

集港收箱之前，场地计划员要根据 EDI 导入的出口箱量做制定堆场计划，场并根据类别、流向、卸货港、箱型、重量范围、以及码放规则合理的设定收箱策划范围。如图 5.11 所示。

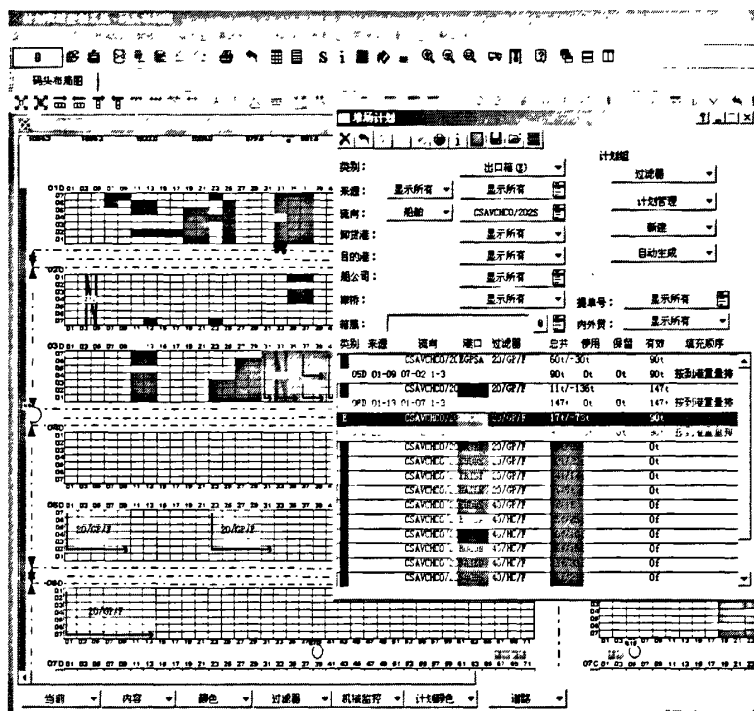


图 5.11 出口堆场计划

Figure 5.11 Export yard plan

出口堆场计划的设定要考虑的因素比较多了，比如重量等级，本系统对重量等级按照航线进行了比较详细的划分。设定好航线的重量等级后，对堆场计划进行条件的约束，进行重量等级的控制。同样，堆场计划还可以就码放规则，双吊具等等进行设定。按照设定好的堆场计划进行收箱，能够保证不同港口的箱子进行分港口分重量码放，为以后的问配载工作做好前提准备。

5.2.4 闸口收箱

闸口操作员会审核出口集港箱的装箱单和设备交接单等单据，核对之后将相关信息输入系统。然后打印入场小票，指导集港车辆进场作业。同时闸口还可以进行打捆箱的处理，以及修改来港箱的属性。如图 5.12 所示。

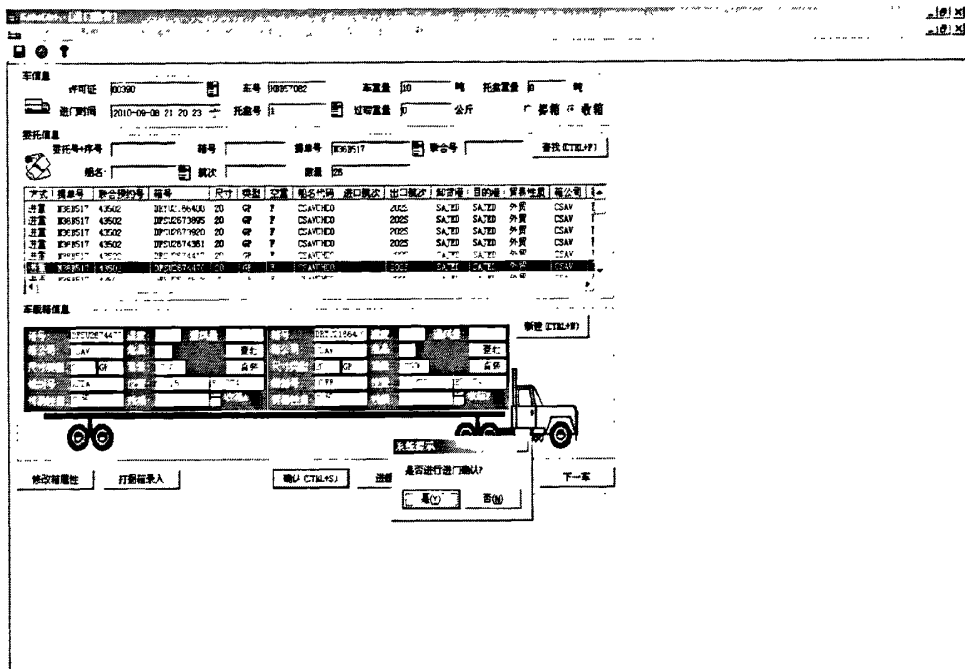


图 5.12 闸口收箱操作界面

Figure 5.12 Gate collection box interface

5.2.5 出口箱放行操作

单船计划员会在配载前进行出口清单、在场箱及场站收据的验证，当三单齐全时系统允许该箱配载并放行。出口箱的放行要求严格，必须有海关的放行信息才可以进行装船操作，单船计划员，在集港收箱中就要进行三单核对，而且三单核对要多次进行操作，因为箱子的放心信息是一直在变更的，没有放行的箱子是不会出现在配载列别中的。三单核对系统主要是对出口清单、在场箱信息，场站收据（海关放心信息）三个环节进行详细的核对，三单齐全的系统会自动发行，将数据打入配载的数据准备表中。由于核对的次数比较多，因此系统进行了控制，不允许多人同时进行三单核对。出现问题的箱子，会在错误报告与结果清单里显示，单船计划员可以对问题箱子进行处理，与船公司核实情况，做到操作游刃有余，核对结果如图 5.13 所示。

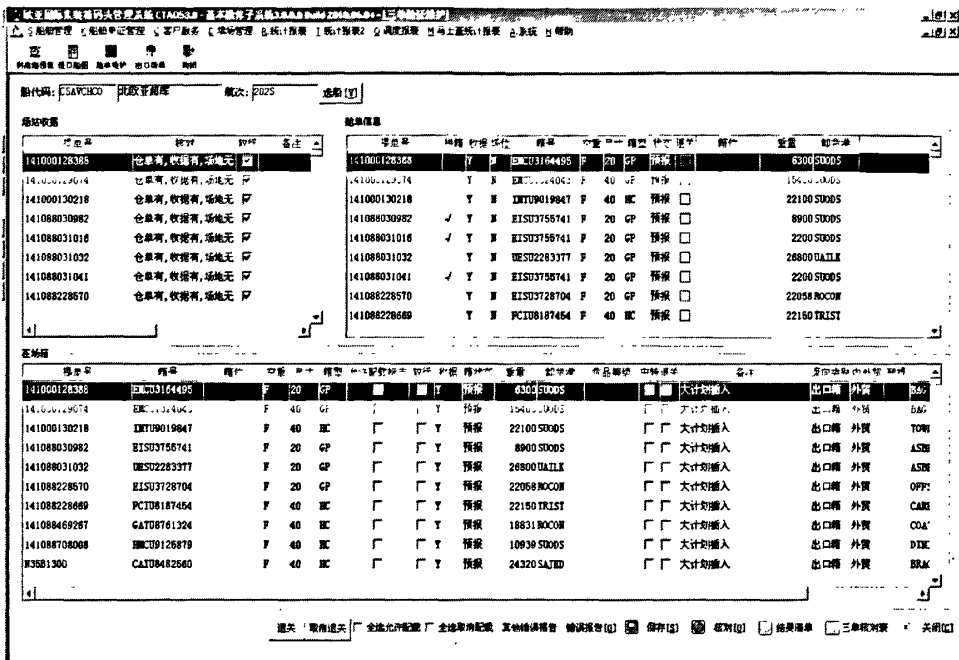


图 5.13 三单核对放行出口箱界面

Figure 5.13 Three single check box interface export clearance

1) 该窗口界面中“核对”按钮对应前台 PB 代码如下:

```
declare proc_check procedure for
p_three_check(:is_ship_no,:uapp.is_oper_nam);
--第一步调用存储过程 p_three_check
execute proc_check;
fetch proc_check into :ls_out;
close proc_check;
if (SQLCA.sqlcode = 0 or SQLCA.sqlcode = 100) AND ls_out = '99' then
else
messagebox('p_three_check 核对失败!', ls_out+'/' + sqlca.sqlerrmtext)
close(w_message)
end if
setnull(ls_out)
declare proc_check_bill procedure for
p_three_check_bill(:is_ship_no,:uapp.is_oper_nam);
--第二步调用存储过程 p_three_check_bill
execute proc_check_bill;
fetch proc_check_bill into :ls_out;
```



```

close proc_check_bill;
if (SQLCA.sqlcode = 0 or SQLCA.sqlcode = 100) AND ls_out='99' then
    messagebox('提示','核对完毕!')
else
    messagebox('p_three_check_bill 核对失败!',
ls_out+'/' + sqlca.sqlerrtext)
    close(w_message)

```

2) 三单验证后台算法代码如下:

```

SELECT COUNT(*) INTO wl_count FROM T_PORT_CNTR_TMP WHERE
SHIP_NO=V_SHIP_NO;
IF wl_count > 0 THEN
    vs_return:= '不能多人同时进行核对!';

```

此段代码用来判断是否有多人核对，防止数据锁的产生。

```

FOR I IN (SELECT DISTINCT BILL_NO FROM THREE_BILL_CHECK_BILL WHERE
SHIP_NO=V_SHIP_NO ORDER BY bill_no ASC) LOOP
    BEGIN
        BEGIN
            SELECT CNTR INTO WL_CNTR
            FROM T_PORT_CNTR_TMP
            WHERE SHIP_NO=V_SHIP_NO AND
                BILL_NO=I.BILL_NO AND
                CURRENT_STAT NOT IN ('2','4','5') AND
                NVL(exit_custom_id,'0') <> '1';
            I_IN_PORT_CNTR := 1;
        EXCEPTION

```

此段代码先根据场站收据表验证在场箱表数据以及出口清单数据

```

IF (I_IN_PORT_CNTR = 0 AND I_PORT_CNTR_NUM > 0 ) AND I_IN_EXP_CNTR
> 0 THEN
    对于三单齐全的处理
        UPDATE THREE_BILL_CHECK_BILL
        SET CHECK_ID='1',CUSTOM_ID='1'
        WHERE SHIP_NO=V_SHIP_NO AND
            BILL_NO=I.BILL_NO;
    ELSIF (I_IN_PORT_CNTR > 0 OR I_PORT_CNTR_NUM = 0) AND I_IN_EXP_CNTR > 0
THEN

```

对于有舱单不在场箱的处理

```
UPDATE THREE_BILL_CHECK_BILL
SET CHECK_ID='2',CUSTOM_ID='1'
WHERE SHIP_NO=V_SHIP_NO AND
      BILL_NO=I.BILL_NO;
ELSIF (I_IN_PORT_CNTR = 0 AND I_PORT_CNTR_NUM > 0 ) AND
I_IN_EXP_CNTR=0 THEN
```

对于无仓单在场箱的处理

```
UPDATE THREE_BILL_CHECK_BILL
SET CHECK_ID='6'
WHERE SHIP_NO=V_SHIP_NO AND
      BILL_NO=I.BILL_NO;
ELSIF (I_IN_PORT_CNTR > 0 OR I_PORT_CNTR_NUM = 0) AND
I_IN_EXP_CNTR=0 THEN
```

对于无仓单不在场箱的处理

```
UPDATE THREE_BILL_CHECK_BILL
SET CHECK_ID='7'
WHERE SHIP_NO=V_SHIP_NO AND
      BILL_NO=I.BILL_NO;
END IF;
```

5.2.6 出口船图配载

码头单船计划员根据预配图和码头集港堆场情况,综合考虑船舶要求和码头作业要求,编制码头集装箱实际配载图。实际配载图得到船方确认后,码头按照该实配图进行集装箱装船作业。配载可以分为两种方式:

1) 手动配载 手动配载,是由单船计划员灵活选定配载条件、范围,按照船舶结构图,逐一进行配载,通过配载锁功能,实现多人同时配载。在配载过程也采取了人性化的提示。

2) 自动配载

系统支持,自动配载,只要设定好配载的条件,可以实现自动配载。

5.2.7 集装箱三态显示与作业队列四层模型

1) 箱三态显示:在场箱进行配载后,可以有三种不同的显示,分别是当前,现在和复合。如果是当前状态,箱子尚未装船,则显示为实心状态,选择将来状态,

该场位变为空白，复合状态则显示为重影状态。中控员可以清晰的分辨集装箱的当前状态，更好的对现场作业给予监控与指导。

2) 配载完毕后，可以查看可以作业的作业队列，即装船的作业队列，这里我们为了方便查询操作，设计为四层机构：作业任务、作业队列、队列组、组明细。如图 5.14 所示。

作业任务四层模型

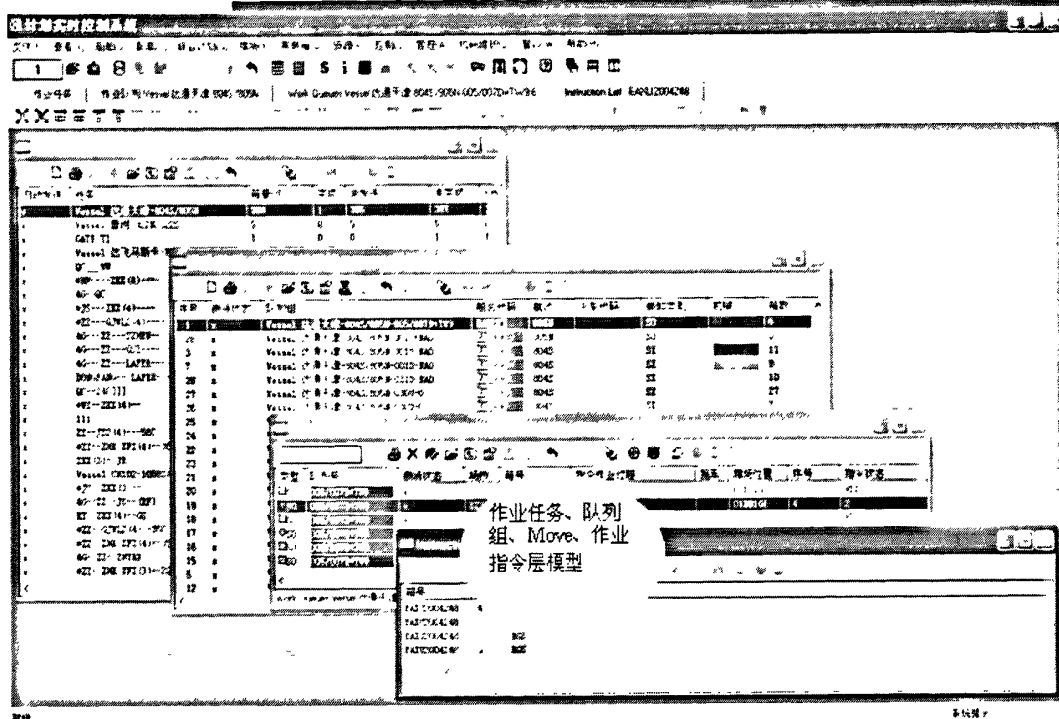


图 5.14 作业任务四层模型图例

Figure 5.14 Tasks in the four model Legend

而且作业队列组可以跳转到分贝图，如图 5.15 所示。

作业队列组可跳转到船舶分页图

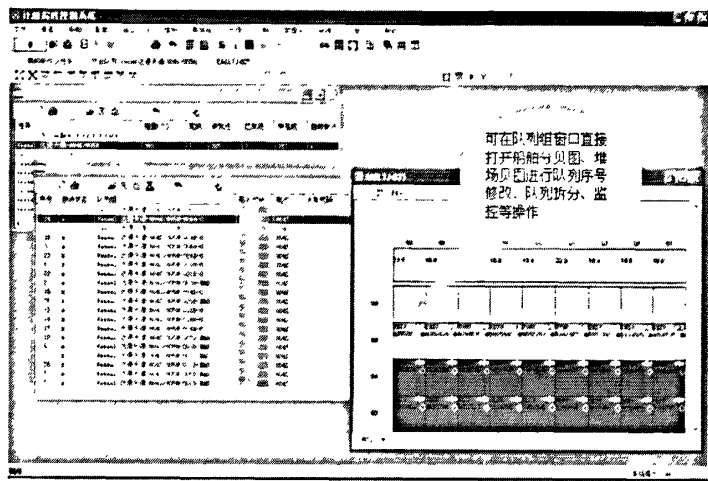


图 5.15 作业队列跳转分贝图

Figure 5.15 Job queue jump decibel chart

配载之后，调度可对配载的箱子发送装船指令，经岸边理货员手持 P 箱上船，至此整个装船作业流程结束，这个过程简单的说是卸船工艺的反向操作，操作图例就不在做详细介绍。

5.3 消息服务器

消息服务器保证了客户端显示与数据库数据的实时同步。服务器和客户端都与消息服务器相连，当任意一台客户端（PC、场桥终端、拖车终端、手持终端）导致数据变化时，系统将通过消息机制将该变化迅速发送到其它所有客户端，使得各数据端得到的数据同步一致，不需操作人员手工刷新。实时消息传递拓扑图如图 5.16 所示，图中 HDM Server 为消息服务器，通过客户端程序配置文件使客户端与消息服务器相连接。以卸船为例，场桥终端确认箱落场后，该箱状态变为在场，消息服务器迅速将这种变化广播至各客户端。

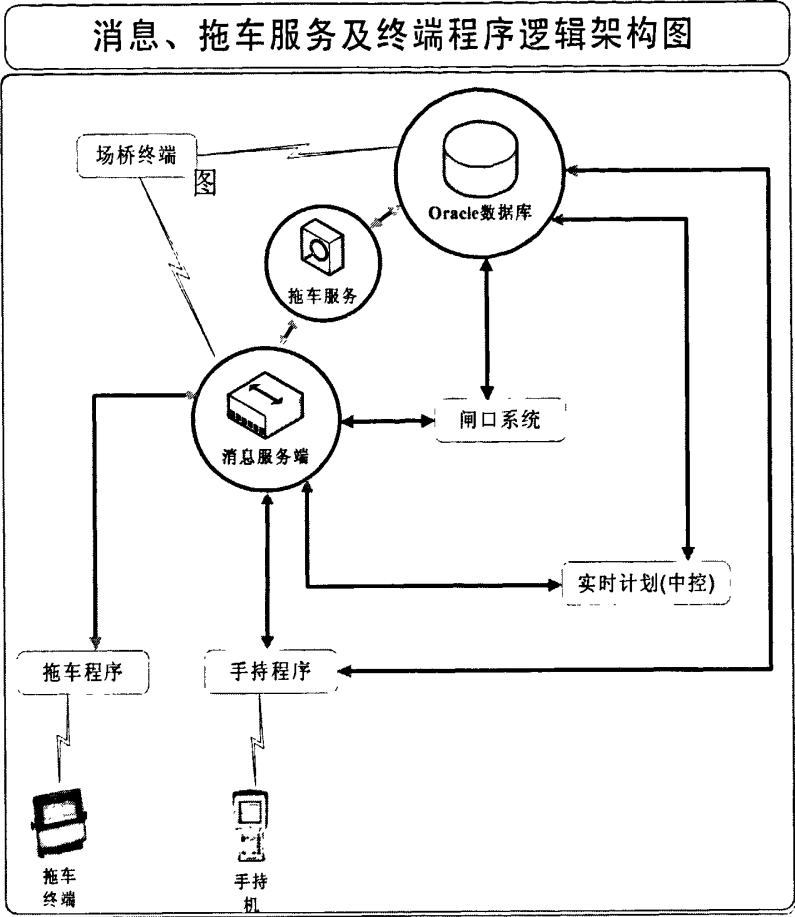


图 5.16 消息服务器

Figure 5.16 Message server

第6章 总结与展望

6.1 总结

CTOS 基于实时架构,采用图形化操作方式,通过先进的控制平台和各种优化算法,实现了动态实时、自动计划和监控、智能预演和回放、用户报表自定义、自动配载、智能集卡调度等目标。模块化的设计方法使 CTOS 更具扩展性。通过对泊位、堆场空间和机械设备等资源进行智能化控制,充分考虑了岸桥等各种设备及操作工艺的特点,适用于各种规模的集装箱码头。智能计划与智能操作系统以稳定、实时、高效、智能、灵活为目标,采用面向对象的方法设计,设计上应用了多种设计模式,提高了代码的高效性、易用性、可读性,并采用消息服务提高系统的实时性。

CTOS 目前已经在天津港联盟国际,太平洋国际,欧亚国际投入正式使用,自投入运行以来,快速、准确、及时、自动地将码头生产的各种信息反应到相关用户手中,大大提高了港口生产效率,树立了一流大港形象,保障了天津港集装箱运输的长期稳定增长,提高天津港在市场竞争中的地位。今年 CTOS 正在天津港集装箱码头、东方海陆集装箱码头、五洲国际集装箱码头三家码头公司推广应用,基于 CTOS 系统在天津港的广泛应用,未来将逐步开展航运中心一站式集中服务的研发整合工作,将逐步完成全港集卡调度资源整合,这些工作的逐步开展必将大大节约资源,最大程度利用现有资源,实现信息共享,做到互联互通。

6.2 展望

系统设计研发完成后,将为天津港集团提供港口集装箱装卸运输完整解决方案,天津港集团将成为国内港口集装箱装卸运输信息化领域最具竞争力的企业,系统将具备综合的行业优势、先进的功能优势、领先的技术水平,将具备替代进口集装箱码头软件产品的能力。

CTOS 系统为建设北方国际航运中心和国际物流中心提供强有力的支撑。系统设计研发使用后可每年为社会创造 GDP 增加值 3 亿多元,同时可节省大量的港口建设投资。带动 RFID 等相关信息技术产业的快速发展。对企业经济效益和社会效益的提高产生良好的推动作用,为天津港跨越式发展奠定了良好的基础。

参考文献

- [1] 雷晓明, 王建章, 李江, 新世纪初信息产业导向, 北京: 清华大学出版社, 2000, 1~10
- [2] 陈戌源, 集装箱码头业务管理, 大连: 大连海事大学出版社, 1998, 32~33
- [3] 游五洋, 陶青, 信息化与未来中国, 北京: 中国社会科学出版社, 2003, 218~237
- [4] 于汝民, 现代集装箱码头经营管理, 北京: 人民交通出版社, 2007, 135~138
- [5] 侯俊杰, 深入浅出 MFC, 华中科技大学出版社; 第 2 版, 2001, 235~237
- [6] 赵书良, 王艳君, 邱志宇, 基于 PowerBuilder 的可视化管理信息系统的研究与实现, 计算机应用研究, 2003, 153~154
- [7] 杨昭, Powerbuilder 9.0 数据窗口技术, 北京: 中国水利水电出版社, 2003, 1~10
- [8] 郑阿奇, PowerBuilder 实用教程, 北京: 电子工业出版社, 2004, 1~15
- [9] 飞思科技研发中心, Oracle9i 基础与提高, 北京: 电子工业出版社, 2003, 1~10
- [10] 张青, 范颖, 王继成, Oracle9i 中文版基础教程, 北京: 清华大学出版社, 2002, 10~30
- [11] Kevin Loney 著, 李晓军译, Oracle 8 数据库管理员手册, 北京: 机械工业出版社, 2000, 96~98
- [12] 凯特, 苏金国, Oracle9i&10g 编程艺术: 深入数据库体系结构, 北京: 人民邮电出版社, 2006, 67~72
- [13] 蒋年德, 李 英, Oracle 环境下管理信息系统的优化设计, 计算机工程与设计, 第 12 期, 131
- [14] (美) Benjamin L. kovitz, Practical Software Requirements 著, 胡辉良译, 北京: 机械工业出版社, 2005, 16~21
- [15] 张海藩, 软件工程导论, 北京: 清华大学出版社, 2003, 45~62
- [16] 赵磊, 集装箱码头作业管理系统的设计与实现: [硕士学位论文], 天津, 天津大学, 2009
- [17] [英] C. J. Date, Hugh Darwen, 数据库、类型和关系模型, 北京: 机械工业出版社, 2007
- [18] 邓良松, 刘海岩, 陆丽娜, 软件工程, 西安: 西安电子科技大学出版社, 2000, 36~40
- [19] (美) Ivar Jacobson, Grady Booch, James Rumbaugh, The Unified Software Development Process 著, 周伯生译, 北京: 机械工业出版社, 2002, 6~11
- [20] 丁宝康, 数据库原理, 北京: 经济科学出版社, 2000, 110~120
- [21] 姚春龙, 数据库系统基础教程, 北京: 北京航空航天大学出版社, 2003, 58~75
- [22] 黄林鹏, 王德俊, 张仕, 计算机算法的设计与分析, 北京: 机械工业出版社, 2007, 46~60
- [23] 陈启安, 软件人机界面设计, 北京: 高等教育出版社, 2004

- [24] 刘嫚, 陈梦东, 甘雨, 管理信息系统中统计报表灵活定制的设计与实现, 计算机应用与软件, 2005, 22(9):79~85
- [25] John W. Satzinger, Robert B. Jackson 著, 朱群雄译, 系统分析与设计, 北京:机械工业出版社, 2002, 215~220
- [26] 程言清, 港口物流管理, 北京:电子工业出版社, 2007, 177~182
- [27] 陈世鸿, 彭容, 面向对象软件工程, 北京:电子工业出版社, 1999, 117~119
- [28] 赵刚, 航运企业生产管理, 大连:大连海运学院出版社, 2000, 19~25
- [29] 洪永清, 黄德才, 吕丽民, 面向对象建模与设计, 北京:人民邮电出版社, 1998, 103~106
- [30] 何旭洪, 傅立宏, PowerBuilder 9.0 信息管理系统开发实例导航, 北京:人民邮电出版社, 2005

攻读学位期间公开发表论文

- [1]陈鹏. PowerBuilder 数据库中并发控制的实现. 发表于《黑龙江科技信息》.
- [2]陈鹏. 太平洋国际集装箱码头有限公司生产调度交接班系统研究. 发表于《中国新技术新产品》.

致 谢

值此论文完成之际，我谨向在海大学习期间给予我无私帮助和热情支持的老师和同学们表示感谢！

本论文的工作是在我的导师赵广利副教授的悉心指导下完成的。导师丰富的学识和亲切的教诲给了我启迪和鼓励，在此，我向赵教授表示深深的感谢。

同时，我要感谢大连海事大学，感谢计算机科学与技术学院给了我学习、深造的机会，感谢在攻读工程硕士学位期间所有老师的辛勤培育和指导，你们传授给我的专业知识是我不断成长的源泉，也是我完成本论文的基础。在此，向你们表示诚挚的感谢！

另外也感谢我的单位领导，我的企业导师马全华，是他们的理解和支持使我能够全身心地投入到毕业设计中，并顺利完成学业。在此，我向他们表示衷心的感谢！

最后，非常感谢各位专家、评委们参加我的论文评审工作。

1

—