

基于三F装备可靠性设计与管理系统研究

曾慧娥 (重庆工业高等专科学校 400016)

周庆忠 (重庆后工学院 400016)

摘要:本文对基于三F装备可靠性设计与管理系统进行研究,论述系统模型、结构和功能设计方法,叙述系统实现关键技术,阐明该系统的研究对促进三F技术的推广,提高工程装备可靠性具有重要作用。

关键词:可靠性 装备 故障 FMECA FTA FRACAS

一、问题提出

随着科学技术的发展,工程装备逐步向标准化、单元化和集成化方向发展,对装备可靠性提出了更高的要求。可靠性是固有的质量属性,它表示装备在使用过程中能否保持正常工作状态或完成规定任务的能力。要保证装备达到高可靠性要求,必须运用可靠性设计和管理技术,从指标论证,到装备设计、研制,乃至使用管理的全过程,通过一系列可靠性工程活动,确保装备的可靠性。三F技术是行之有效的三项可靠性技术,对保证装备可靠性具有重要作用,在工程装备,尤其是在军用装备研制与管理领域普遍采用并取得良好的效益。所谓三F,即是故障模式影响及危害性分析 FMECA、故障树分析 FTA、故障分析及纠正措施系统 FRACAS。三F以故障作为工作对象,运用分析方法来获取决策信息,为分析设计、改进设计、完善管理体系,增强装备可靠性提供科学依据。

由于各类装备功能的不断提高,装备日趋复杂,所含部件越来越多,故障信息繁杂,若仍以手工方式对装备进行三F分析、设计和管理已力不从心。应用计算机进行辅助三F分析、设计和管理势在必行。因此,对基于三F技术的装备可靠性设计与管理系统的研究很有必要。

二、系统模型与结构设计

用功能/层次矩阵来描述支持三F各项功能,系统结构如图1所示。

图中每一列代表支持三F可靠性活动的子系统;每一行代表一个管理层次;每一个纵横交汇的方块代表为某一管理层次服务的管理功能所对应的模块或文件。如 FMECA 子系统是由故障模式及影响分析(FMEA)模块和危害性分析(CA)等模块所构成,同时设有自己专用的数据文件。被整个系统所共享的数据和功能,包括公用数据文件、公用应用模块、故障模型库及可靠性数据库管理等模块。每一模块又可用一个树结构进一步分解成下级更小的模块。

运用“责任驱动”方法建立系统概念模型,模型被划分为五个层次:对象层、结构层、主题层、属性层和服务层。它强调以对象在问题中所承担的责任为依据来获取层次结构。系统实现模型是对得到的概念模型进行修改、完善,在系统设计策略指导下,将概念模型转换为系统模型,并根据系统所具备的环境条件、运行要求及实现语言对系统模型作进一步修改,以得到能与面向对象语言直接相对应的实现模型。主要工作有对人机交互部分、问题域部分、任务管理部分及数据管理部分等四个方面的扩充与完善。

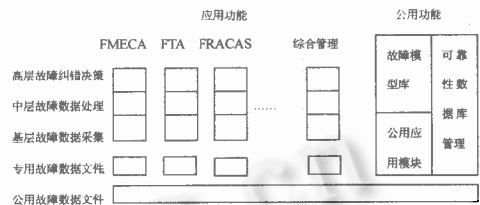


图1 系统结构图

系统采纳视图—模型—数据分离的分层结构。把模型对象封装在复合视图模块中。所有的故障数据处理逻辑和验证规则都是在模型类中实现的。模型对象包括:装备、故障、故障发生时域。数据操作对象有:故障数据采集,故障分析和故障纠正处理。模型对象和数据操作对象被实现为非可视化部件,有助于使设计出来的可视化部件连接更少。分层结构系统的特点是用户界面与应用逻辑位于不同的平台上,并且应用逻辑被所有用户共享。其主要优点有:(1)可伸缩性好。应用服务层通过一个数据库连接池与数据库连接,根据用户请求的多少来动态调整池中的连接数,使系统消耗较少的资源来完成用户请求。(2)可管理性强。整个系统被分为不同的逻辑块,层次清晰,管理和维护变得相对简单。(3)安全性高。应用服务层上的安全服务作为一个公用服务被所有应用调用,不必为每一个应用编写安全服务,用户不

能直接访问数据库。(4)可重用性好。整个系统由许多服务组成,每种服务可被不同应用重用。构建系统时采用面向对象的组件模式,每种服务又由许多可重用的组件构成。

三、系统设计

对应于三F技术,系统设置FMECA、FTA和FRACAS三大子系统,三者相互关联,又各具特色。

1. FMECA子系统

FMECA子系统为运用归纳方法系统地分析产品设计可能存在的每一种故障模式及其产生的后果和危害程度,通过全面分析找出设计薄弱环节,实施重点改进和控制。该子系统主要模块有:

(1)故障模式及影响分析(FMEA):定义产品,构建方框图;确定产品进行FMEA的最低约定层次;选择与填写FMAA表格;

(2)危害性分析CA:填写危害性分析表格,绘制危害性矩阵

分析对象包括研制任务书或合同内规定的项目,也包括由转承制方承担供应的外协装备、外协件。对象除了电子、电气、机械、热、机电、液压、气动、光学、结构、动力、点火器等火工品等硬件及产品的组成功能外,还包括试验装备、试验方法、工艺技术软件及人(研究人员的操作差错、操作序列差错等的影响)。

输入信息主要有:

(1)设计任务书:如设计产品的技术指标要求,执行功能,装备工作的任务剖面、寿命剖面以及环境条件、试验要求、使用要求、故障准则、其他约束条件等。

(2)设计方案论证报告;

(3)被分析对象在所处的系统内的作用与要求的信息:如所处系统诸组成单元的功能、性能的要求及容许限,诸组成单元间的接口关系及要求,被分析对象在所处系统内的作用、地位。

(4)有关的设计图样:如在研制初期的工作原理图和功能方框图。

(5)被分析对象及所处系统、分系统在启动、运行、操作、维修中的功能、性能、可靠性信息:如不同任务的任务时间;测试、监控的时间周期;预防维修的规定,修复性维修的资源;对可能出现严酷度高的等级的后果,特别是属于安全性事故时,能采取应急补救措施的时间。

(6)可靠性数据及故障案例 如典型电子元件、机械零件故障模式库及频数表。

FMECA以综合报告形式输出分析信息。报告数据有:所分析的对象,约定层次,引用的故障数据源,分析方法,分

析表格,分析得到的严酷度为I、II类故障模式,建议的补偿措施,关、重件及单点故障模式清单等。

2. FTA子系统

FTA子系统功能为运用演绎法找出导致装备故障的各种潜在危险因素及其相互关系,构造故障树,经定性和定量分析,识别系统故障模式,评价故障因素的影响程度。主要模块包括:

(1)数据接口管理:以一体化管理方式,包括对故障树按用户分组管理,并可对故障树进行创建、选择、删除、复制和打印等基本操作。此外,提供自动的内部数据转换,并控制用户的使用流程。分为两级,第一级是“用户”,第二级是“故障树”。“用户”可以是个人项目组。每个“用户”或以对属于他的一个或多个“故障树”进行管理、编辑和分析。

(2)故障树编辑:是与用户交互的主要界面,是数据输入的通道,所谓数据主要是指用户对故障树和相关失效数据的定义。有两种编辑方式:一是用传统的故障树文本格式进行编辑,二是用新颖的图形方式进行编辑。对较为复杂的故障树进行自动排版,生成精美故障树插图,供用户编写报告使用。在排版之前,系统内部先对故障树建树或输入时产生的逻辑或语法进行错误检查。

(3)故障树定性分析:为FTA核心模块。提供下行法、上行法和混合算法三种算法。按割集概率或割集阶数对割集进行截断,用户可根据需要设置。

(4)故障树不确定性分析:使用Monte-Carlo方法,求出顶事件所代表的系统不可用度在不同置信度下的置信限、频率直方图、分布均值和方差。

(5)故障树定量分析:求解底事件概率重要度、相对概率重要度、结构重要度。

3. FRACAS子系统

FRACAS子系统功能为对故障实施有计划、有组织、按程序地调查、证实、分析和纠正工作,保证故障原因分析的准确性和纠正措施的有效性,对故障实行闭环控制,彻底消除故障产生的原因,真正实现问题“归零”。

FRACAS是一个故障信息子系统,它将极其分散的可靠性数据及时、准确完整地收集起来,并按可靠性工程的要求加以科学分类、整理,编制产品可靠性电子数据档案,为修订可靠性设计准则、改进产品可靠性管理提供科学依据。FRACAS主要包括数据采集、加工、储存和传递等功能模块,其输入的是故障报告信息,输出的是纠正措施信息。运行FRACAS的关键是对信息流—故障信息流以及物流—故障产品的控制。为此,按规定统一故障信息报表,统一处理故障产品。故障信息包括:故障对象的识别数据:装备种类、编号、生产厂家、出厂日期、使用经历等;故障识别数据:故障类别、发生时间、发现时状况等;故障鉴定数据:故障现象、故障

原因、寿命、测试值等;故障装备历史资料。故障信息是研制的极其重要资源,应妥善处理及保存。为了将故障数据管理从功能程序中分离出来,采用图2所示故障数据管理模型,这种模型便于适应不同的数据库,可根据不同应用情况下的响应时间特性而采取不同的方式。

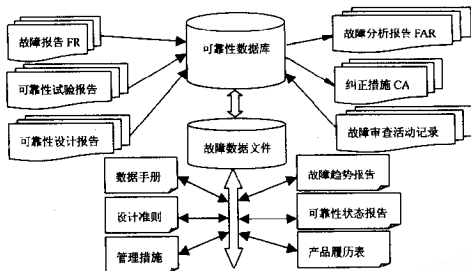


图2 FRACAS故障数据模型

四、系统关键技术

1. 从部件进行构造技术

在系统设计中,从部件进行构造原则为:

(1)用容器和表的图元作为用户界面的主体形式。在各子系统中使用一致的选择风格,用上下文菜单或按钮对一个对象选择一个操作。

(2)设计中尽量进行复用,降低连接的复杂度。提供一个在可视化编程中使用起来方便自然的公共界面,并确保一个部件的公共界面尽可能简单。

从部件进行构造的工作是从创建一个可视化类入手的,创建一个可视化类,进行以下工作:

(1)用快速表格显示对象或手工编写出用户界面,如故障树的构建。寻找集成最终用户界面所需的额外可视化类。把非可视化类作为一个变量子部件增加进来,使之成为此可视化类公共界面的一个属性。

(2)编辑基类的已有对象,这个可视化类可把基本显示视图作为子部件。为了提供编辑功能,用延迟更新部件作为该类的一个子部件。执行动作将由验证动作产生的正确事件触发。输入数据应被验证,同时也应为系统定义必需的外部事件。

(3)为此基类创建新对象。在某些情况下这个类可能与基本编辑视图相同。该类可用基本编辑视图作为子部件或实现为基本编辑视图的子类。

(4)显示对象的列表,定义所需的外部事件。

2. 数据库技术

建立数据库来存储和管理故障数据,降低数据存储的冗

余度,实现公共数据的充分共享,使应用功能与数据尽可能地相互独立,使得应用功能不但较少地依赖于数据的存储结构和介质种类,而且当数据结构改变时,不要求功能模块作较大的修改,同时,对数据的完整性、安全性和保密性提供统一的控制手段。故障数据库属于工程数据库,具有以下特点:

(1)数据结构和数据类型复杂 既有常规的结构化数据,如表示零件信息的零件图号、零件名称、毛坯材料等,又有特殊的非结构化数据,如零件的几何图形、装备典型故障率曲线、装备寿命周期特征曲线等。

(2)数据语义丰富 数据之间的联系多种多样,语义十分丰富。除了一般实体间的一对多、多对一、多对多关系外,还有其他一些特殊的关联:实体间具有继承性,实体间具有限制关系,数据具有动态特性。

(3)由于故障有一定随机性,这使得对故障数据采集和管理实时性要求较高 对数据动态特性,绝大多数体现在装备状态数据中。一个大型装备状态不计其数,且变化频率相当快,它们随着装备运行由状态采集系统源源不断地收集来,并在数据库中得到及时反映。

3. 对象建模技术

对象建模技术是用一组面向对象概念和与程序语言无关的图形符号来分析问题的需求,设计解决问题的方案,然后用程序语言或数据库实现。对象模型描述对象、类及其之间相互关系的静态数据结构。一个对象模型包含一个或多个模块。一个模块有一个或多个页面。

根据调查所得资料和需求文档,确定实体故障对象类:系统、装备和部件、零件等,功能对象类:FMECA、FTA和FRACAS等,它们都提供增加、删除、修改和查询等服务,还要有用户、用户单位、管理部门、管理者等对象与这些对象合作。围绕系统问题域的责任,找出承担这些责任的最小对象空间类。结构是问题域复杂关系的表示。若合作者之间的联系反映了组装、包含、容纳这样的整体一部分观念,则提炼为聚合关系;若多个类对象具有相同或相似的责任及联系,则将它们表示为继承关系。定义主题是从更高角度考察整个系统,是对问题空间的高度概括和总结。主题之间通过“故障报告”建立消息连接。由于对象是封装了属性及其上的服务的机制,应对属性进行定义,每项对象要完成自己的责任、实现服务,必须确定它所保存的信息,以反映其状态变化。如用户完成其输出及查询服务,须保存名称、单位属性;而故障则有名称、分类、失效率、有效度等属性。因故障处理应用的纷繁复杂,中间件产品只能提供一些通用的服务,大量的开发工作仍需开发人员根据实际情况完成,故在使用面向对象技术时,特别要注意使用对象组件技术。

(来稿时间:1999年9月)