

知识融合 程序设计与应用

■ 朱文华 杜素芬 主编



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

知识融合程序设计与应用

朱文华 杜素芬 主编

上海交通大学出版社

内 容 提 要

知识融合(Knowledge Fusion)技术是通过获得和操纵工程规则、设计意图,利用工程知识库和工程演算规则建立强大的产品开发应用系统的技术。一个单一的产品数据模型就能够用来管理设计参数和知识信息,设计工程师就能够构造完全可重复使用的知识库,从而实现设计的自动化。

本书内容系统全面,主要侧重于知识融合程序设计与应用,结合作者多年的应用和项目开发经验编著而成。本书属于 NX 软件的高级培训教材,涉及知识融合程序设计、程序扩展与应用开发三大部分,无论是对初学者还是对有一定基础或一定经验的 CAD 人员,均会有相当的帮助和启发。本书采用实例进行讲解,使读者能在较短的时间内对知识融合程序设计有所了解,并能进行应用。

本书既可作为高等院校机械制造、自动化、计算机和多媒体等专业的研究生和高年级本科生教材,也可作为虚拟现实技术应用人员、虚拟现实技术爱好者的参考用书。

图书在版编目(CIP)数据

知识融合程序设计与应用/朱文华,杜素芬主编.
—上海:上海交通大学出版社,2011
ISBN 978-7-313-07792-9
I. ①知… II. ①朱… ②杜… III. ①计算机辅助设计—应用软件, UG NX 6 IV. ①TP391.72
中国版本图书馆 CIP 数据核字(2011)第 204617 号

知识融合程序设计与应用

朱文华 杜素芬 主编

上海交通大学出版社出版发行

(上海市番禺路 951 号 邮政编码 200030)

电话: 64071208 出版人: 韩建民

常熟市梅李印刷有限公司印刷 全国新华书店经销

开本: 787mm×1092mm 1/16 印张: 19.75 字数: 460 千字

2011 年 11 月第 1 版 2011 年 11 月第 1 次印刷

印数: 1~2 030

ISBN 978-7-313-07792-9/TP 定价: 36.00 元

版权所有 侵权必究

告读者: 如发现本书有印装质量问题请与印刷厂质量科联系

联系电话: 0512-52661481

前 言

知识工程(Knowledge Based Engineering, KBE)技术就是充分地利用工程专家的领域知识和企业在长期的设计开发中所积累的经验,通过知识驱动,对工程问题和任务提供最佳方案的计算机集成处理技术;是近年来十分活跃的研究开发与应用的热点技术之一。

NX 知识融合(Knowledge Fusion, KF)技术是通过获得和操纵工程规则、设计意图,利用工程知识库和工程演算规则建立强大的产品开发应用系统的技术。通过工程规则控制 UG NX 的对象、超越单纯的几何模型,这样一个单一的产品数据模型就能够用来管理设计参数和知识信息,设计工程师就能够构造完全可重复使用的知识库,从而实现设计的自动化。

知识融合语言是一种结构简单、易学易用的语言,它是声明性、解释性的语言,不是过程性的语言,你可以在规则编写时不考虑顺序;它是面向对象的语言,有类和对象,能多重继承,类是对象的共同特征的抽象。通过知识融合语言建立的对象模型与通过 UG NX 系统建立的完全相同,从而以一种简单的方法描述了熔接工程规则的三维设计构思,增强了系统通过工程规则驱动产品设计的能力。

本书的编写主要侧重于知识融合程序设计与应用。书中讲述了知识融合程序设计、知识融合程序扩展和知识融合程序应用的实例。

本书第一篇由鲁刚、陈鹏、周浩、黄夫理、叶振宇、张梦之编写,第二篇由朱文华、杜素芬、周浩、鲁刚编写。全书由朱文华、杜素芬统稿,陈鹏对全书进行了文字整理。苏玉鹏、王大斌、吴东琦、王海东、魏丕广、谭慧猛、崔海波、王天鹏、宋尧、白彬、邹鹏程、李宝瑞等同志参与了资料的收集和部分研究工作,在此表示衷心感谢。

特别感谢 Siemens PLM Software 公司资深专家洪如瑾先生、知识融合专家张先宏先生及他们的同事们的大力支持,衷心感谢上海联宏创能信息科技有限公司提供了部分素材和资助。

在编写本书的过程中,得到上海大学方明伦教授、李莉敏教授和 CIMS 中心的诸多同事的关心和支持,也得到了各方面的大力支持和帮助,在此深表感谢。

由于目前知识工程技术发展极为迅速,且知识融合相关的技术尚未完善,加之作者的水平有限,时间仓促,书中错漏和不当之处恳请读者批评指正。

本书中的程序代码可以在 <http://www.plmvs.shu.edu.cn> 下载,欢迎致信: toney.wh.zhu@gmail.com。

编 者

2011 年 6 月

目 录

第一篇 知识融合程序设计

第 1 章 知识融合语言的介绍	3
1.1 知识工程	3
1.2 知识融合语言概述	4
1.3 关键术语的定义	5
练习 1.1 创建的第一个 KF 程序	6
第 2 章 知识融合语言基本编程	10
2.1 类	10
2.2 属性	13
2.3 声明语句	15
2.4 表达式	16
2.5 代码编写	18
2.6 信息传递	19
2.7 需求顺序	24
2.8 需求值	26
2.9 参考框架	27
练习 2.1 创建一个门的装配类	29
练习 2.2 创建工字梁	31
练习 2.3 创建墙的装配类	38
练习 2.4 墙装配体上开门	40
第 3 章 知识融合语言编程进阶	42
3.1 条件语句	42
3.2 循环语句	46
3.3 方法	56
3.4 函数	61
3.5 表达式块	65

3.6	子列表	67
3.7	组件类	71
3.8	文件操作	71
3.9	装配	74
练习 3.1	传递变量、条件语句训练	76
练习 3.2	循环训练	82
练习 3.3	方法训练	83
练习 3.4	函数训练	84
练习 3.5	表达式块训练	85
练习 3.6	子列表训练	86
练习 3.7	组件类训练	87
练习 3.8	装配训练	88
第 4 章	用户界面样式编辑器	91
4.1	知识融合应用	91
4.2	与用户界面样式编辑器的交互	93
4.3	自定义综合用户界面	97
练习 4.1	创建用户自定义界面	99
练习 4.2	创建一个综合的用户自定义界面	103
第 5 章	知识融合交互类编辑器	109
5.1	交互类编辑器概述	109
5.2	交互类编辑器的安装	110
5.3	交互类编辑器进程	110
5.4	交互类编辑器生成复杂的 DFA 文件	121
5.5	结合几何采用使用类交互编辑器	124
练习 5.1	几何采用训练	129
第 6 章	知识融合中电子表格的访问	133
6.1	知识融合中电子表格的概念	133
6.2	电子表格的应用及电子表格类	134
6.3	电子表格驱动建模及外部电子表格函数	135
练习 6.1	运用 NX 内部电子表格控制圆柱体	136
练习 6.2	运用 NX 外部电子表格驱动空间管模型	139
第 7 章	外部数据库访问	148
7.1	ODBC 接口	148
7.2	UG_ODBC_DATABASE 类	148

7.3 UG_ODBC_RECORDSET 类	149
7.4 ODBC 驱动器	149
练习 7.1 读取外部数据库	152
练习 7.2 写入外部数据库	158
练习 7.3 利用访问外部数据库生成自行车车架	162
第二篇 知识融合程序应用实例	
第 8 章 知识融合在塑壳式断路器设计开发中的应用	179
8.1 塑壳式断路器介绍	179
8.2 塑壳式断路器设计方案	180
8.3 塑壳式断路器主菜单设计	181
8.4 塑壳设计	183
8.5 机构设计	193
8.6 整体设计	208
练习 8.1 脱扣器设计实例	212
第 9 章 知识融合在超临界锅炉设计开发中的应用	216
9.1 超临界锅炉介绍	216
9.2 超临界锅炉主菜单设计	218
9.3 集箱设计	219
9.4 省煤器设计	247
练习 9.1 集箱设计实例	278
练习 9.2 省煤器设计实例	291
参考文献	305

第一篇 知识融合程序设计

第 1 章 知识融合语言的介绍

本章将对知识融合语言进行概述,并对其基本概念和术语进行解释。完成本章学习后,读者就能了解知识融合语言中的一些基本概念和术语。

1.1 知识工程

知识工程(Knowledge Based Engineering, KBE)这个概念是 1977 年由美国斯坦福大学人工智能专家费根鲍姆教授(B. A. Feigenbaum)在第五届国际人工智能会议上提出的。他认为,“知识工程是人工智能的原理和方法,对那些需要专家知识才能解决的应用难题提供求解的手段。恰当运用专家知识的获取、表达和推理过程的构成与解释,是设计基于知识的系统的重要技术问题。”这类以知识为基础的系统,就是通过智能软件而建立的专家系统。由于在建立专家系统时所要处理的主要是专家的或书本上的知识,正像在数据处理中数据是处理对象一样,所以它又称知识处理学。其研究内容主要包括知识的获取、知识的表示以及知识的运用和处理等三大方面。

知识工程可以看成是人工智能在知识信息处理方面的发展,研究如何由计算机表示知识,进行问题的自动求解。知识工程的研究使人工智能的研究从理论转向应用,从基于推理的模型转向基于知识的模型,包括了整个知识信息处理的研究,知识工程已成为一门新兴的边缘学科。

21 世纪人类全面进入信息时代。信息科学技术促进了劳动资料信息属性的发展,从而促使科学技术与生产力比过去更加紧密地结合在一起,构成我们这个时代社会经济发展的新的特征,以计算机、网络和通信相结合的形式,体现在变革社会协作方式的推动力量中。信息化的必然趋势是智能化,它将使世界经济从工业化阶段进入知识经济阶段,即将物质生产和知识生产结合起来,充分利用知识和信息资源,提高产品的知识含量。知识和技术密集型产业将取代劳动密集型产业。知识产业的兴起是后工业化社会的特征。在后工业化社会中,社会的主要功能从生产制造转向了知识经济,理论知识、技术和信息成了商品的主要形式。当前许多人把知识产业与知识服务紧密相连。

随着三维软件的大量应用,企业已经进入了三维设计阶段,基于三维平台提出了更多的需求。由于每个行业、企业都有各自的专有知识,每个产品设计工程师都有自己的经验,并

且在一系列的设计过程中有许多工作都是重复的,于是 KEB 技术在工业领域的应用越来越广泛。美国的 GE、Timken,日本的三菱、日立、马自达、本田等公司普遍都在零部件的计算机辅助设计和制造中引入了 KBE 技术,取得了很好的效果。知识工程技术就是充分地利用工程专家的领域知识和企业在长期的设计开发中所积累的经验,通过知识驱动,对工程问题和任务提供最佳方案的计算机集成处理技术,是近年来十分活跃的研究开发与应用的热点技术之一。

NX 知识融合(Knowledge Fusion, KF)是 Siemens PLM Software 公司基于知识工程理念的 KBE 系统开发工具,主要用于建立基于知识的几何和非几何特征模型,使工程师在设计过程中,能随时获得领域知识的帮助,从而提高设计的创新能力。KF 内嵌在 NX 软件系统中,融合了传统的以计算机三维几何模型为核心的 CAD 技术和传统的知识工程技术,产品设计工程师可以利用知识融合技术将其行业专有的产品知识、工作经验和一些重复性的工作直接构建在产品模型中,使产品的数字化模型提高到一个新的水平,实现知识的再利用,减少重复性工作,从而提高工作效率,避免人为的设计错误,使复杂的设计工作智能化。

1.2 知识融合语言概述

知识融合语言是一种结构简单、易学易用的语言,它是声明性、解释性的语言,不是过程性的语言,可以在规则编写时不需要考虑顺序;它是面向对象的语言,有类和对象,能多重继承,类是对象的共同特征的抽象。通过知识融合语言建立的对象模型与通过 NX 系统建立的完全相同,从而以一种简单的方法描述了融合工程规则的三维设计构思,增强了系统通过工程规则驱动产品设计的能力。

NX 知识融合技术是通过获得和操纵工程规则、设计意图,利用工程知识库和工程演算规则建立强大的产品开发应用系统的技术。通过工程规则控制 NX 的对象、超越单纯的几何模型,这样一个单一的产品数据模型就能够用来管理设计参数和知识信息,设计工程师就能够构造完全可重复使用的知识库,从而实现设计的自动化。

知识融合技术可以从结构控制工程师那里得到设计规则和设计要求,并通过在 NX 中进行实例(instantiation)创建而将这些设计规则和要求传递到产品设计定义中。由于 KF 的存在使得 NX 系统在设计自动化方面的能力大大提高。因为 KF 可以在一个给定的部件中同时获得几何和非几何特性,这就让工程师可以在产品文件中获得涵盖在产品中的 Know How 信息。知识融合语言的主要特点如下:

- (1) 解释性的、面向对象的语言。
- (2) 允许通过创建规则增加工程知识。
- (3) 能够连接外部知识源,如数据库、电子表格等。
- (4) 其他应用程序的接口,例如分析包或优化包的接口。
- (5) 自动获取给出的零件或装配体的几何或非几何属性,并记录能够描述创建这个零件或装配体过程的规则,能够获得 CAD/CAM 文件中的信息和数据。
- (6) 提供丰富的 NX 类库,并允许用户创建自己的类。

1.2.1 声明性和需求驱动的语言

知识融合语言是一种声明性的、由求解需求驱动的、面向对象的解释性语言,而不是过程性的。这意味着程序编写不需要考虑顺序,就像在电子计算表格中那样,公式不必按照特定的顺序书写。如果某个公式被引用了,即需要求解,系统自动找到并求解这个公式,这和过程性语言如 C、Fortran 等不一样。知识融合语言按照规则间的依赖关系决定规则执行的先后顺序。

1.2.2 面向对象的语言

如同 C++ 等计算机语言,知识融合语言是面向对象的语言。有类和对象,以及多重继承性,类(class)是对象的共同属性的抽象。

例如,长方体(block)是一类具有长、宽、高属性的几何对象的抽象。一个对象是一个类的特定实例。当创建一个长方体对象时,只需要定义长方体类中的长度、宽度和高度。这样就可以根据需求创建 block 对象,如 block1、block2、block3 等。它们可以尺寸不同,但它们都属于 block 类。从一个类中创建一个对象的过程称为实例化。

1.2.3 层次化的语言

在知识融合语言中,工程师在设计产品时都采用层次化的思想。组件组成了子装配件,而子装配件组成了装配件,并在所有层次中都采用一致的处理方法。

1.2.4 知识融合语言与交互式系统

知识融合语言和交互式系统一样,可以创建 NX 对象。同一对象的语言规范和交互系统中所得到的有用信息是相同的,其差异仅仅在于对唯一对象的描述方法不同而已。

如果一个对象是交互式编辑的,那么与之相应的 KF 语言描述会随着对象的变化而自动地改变,反之亦然。根据相应的规则,可以控制由交互式编辑带来的影响。通过对基本规则属性的设定,可以拒绝任何的编辑修改。只有更深层次的策略才能使这些修改的行为成为可能。

知识融合系统如同 CAD 系统,用户可以用不同的方式与系统交互:

- 知识融合部件可以是“纯几何”部件。
- 知识融合语言可以在 NX CAD 系统中直接创建几何元素。
- 知识融合系统控制其部件间的关系,并自动将它传递到 CAD 系统中。
- 知识融合系统支持 NX 的几何体。

1.3 关键术语的定义

1.3.1 类

- 类是创建一个对象的逻辑演算。
- 类是一些规则的集合。

- 类中包含能够接受特定数值的参数。
- KF 中的类与 C++ 和 JAVA 中的类相似。
- 类通过实例化来创建 KF 对象(Instances)。
- 类可以通过创建其他类的实例而形成一个对象树。

1.3.2 对象

- 对象是类的实例。
- 对象也可被称为一个实例。
- 对象有着类中参数的特定数值。
- 类可以多次实例化得到多个不同的实例。

1.3.3 属性

- 属性是一个可以被引用的有名字的值。
- 属性类似于计算机程序语言中的变量。
- 属性可以是浮点数、整数、字符串、点、矢量、实例和列表等。
- KF 语言不支持数组,但它支持包含混合数据类型的列表。
- 与 NX 中的属性没有关系(KF 的属性通过规则可得到值,而 NX 的属性不会如此)。


1.3.4 规则

- 规则是对象或部件如何被创建的描述。
- 规则用于规定属性值和参数默认值。
- 规则描绘了对象或设计中的约束。
- 规则可以陈述尺寸、半径、长度、角度、起始点和终点等。
- 规则可以是一个算术表达式或公式,或者是可以表达类似设计强度、装配质量、位置、配对和方位这样的信息。
 - 规则可利用数值、名称、字符串或部件等来创建,也可使用函数、引用、方法和条件判断来生成。
 - 规则分为类规则和动态规则,类规则在类文件中,动态规则在实例化时生成在 NX part 文件中。


练习 1.1 创建的第一个 KF 程序

在这个练习中,将创建一个知识融合的程序并且执行它。

步骤 1: 在任意盘符下,创建文件夹层次:“exercise\chapter1\my_dfa_files”。

 | 例如: D:\exercise\chapter1\my_dfa_files。


步骤 2: 用文本编辑器在 my_dfa_files 文件夹下创建 mybox.dfa 文件。

 DFA 文件是 ASCII 格式的文件,字母 DFA 表示 DefClass/Defun File ASCII。以 .dfa 为后缀名的文件,在 KF 中只用于类和方法的定义。

步骤 3: 在 mybox.dfa 文件中输入以下代码。

```
#! NX/KF 6.0
### 类定义 ###
DefClass: MyBox (ug_base_part);
### 定义长度、宽度、高度参数 ###
(number parameter) my_height: 84;
(number parameter) my_width: 36;
(number parameter) my_length: my_width: /12;
### 创建 block1 子规则 ###
(child) block1:
{
class, ug_block;
length, my_length;;
width, my_width;;
height, my_height;;
};
```

步骤 4: 创建新文件。

- 在 NX 中,选择“新建”.
- 单位项选择“毫米”。
- 文件名输入“my_startbox”。
- 点击“确定”。

步骤 5: 显示知识融合导航器。

- 选择“开始”→“所有应用模块”,确认“知识融合”已勾选。
- 在资源条中,选择知识融合导航器(Knowledge Fusion Navigator, DFN),并可以按照用户需要调整窗口大小,如图 1-1 所示。

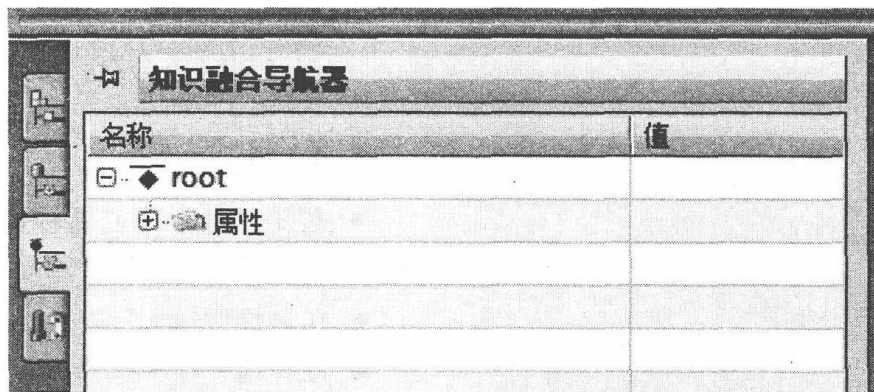



图 1-1 知识融合导航器

步骤 6: 指定知识融合搜索路径。

- 从知识融合工具条中选择“DFA 管理器”(DFA Manager) 。

 | 如果知识融合工具条未显示,可以右键单击工具栏空白区域,并在面板中选择“知识融合”。

- 在 DFA 管理器对话框中选择“浏览”来指定路径,如图 1-2 所示。

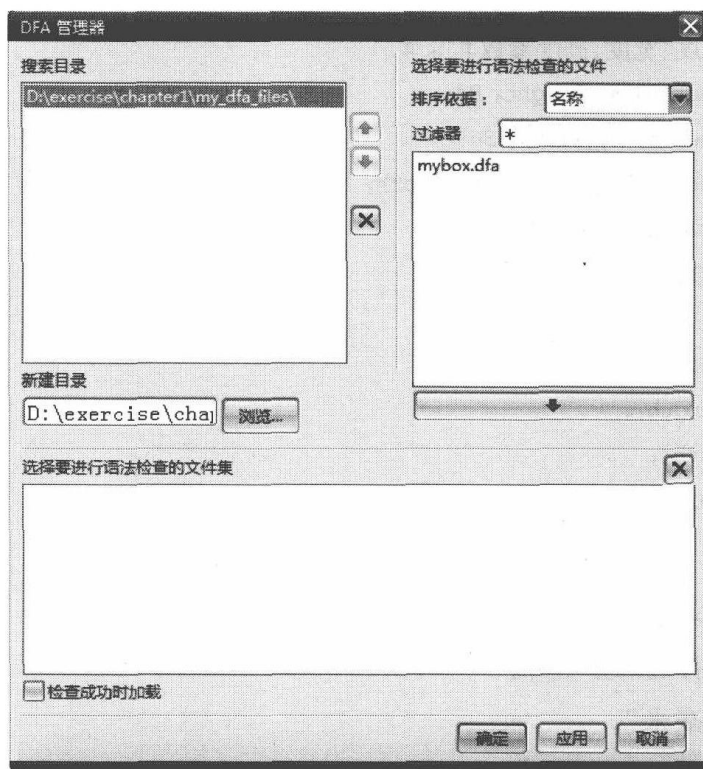


图 1-2 DFA 管理器

- 浏览路径: 文件夹“my_dfa_files”具体地址。

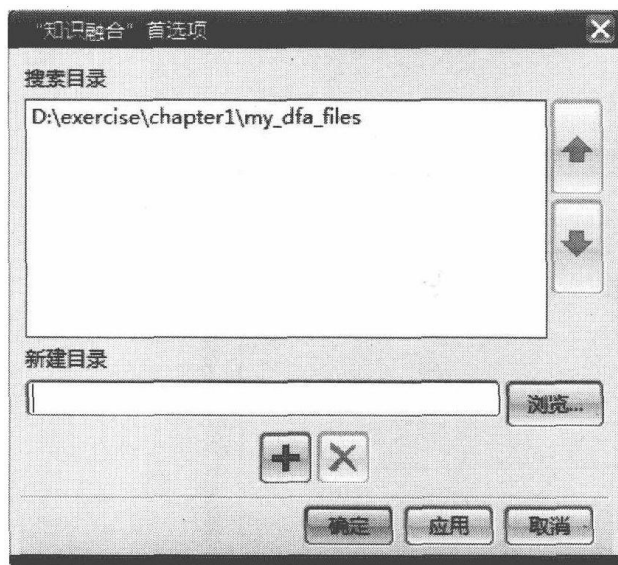



图 1-3 “知识融合”首选项

- 点击“确定”。
- 在“DFA 管理器”对话框中点击“确定”。

同样也可以通过菜单“首选项”→“知识融合”,打开“知识融合”首选项对话框来指定知识融合查询路径,如图 1-3 所示。

步骤 7: 执行程序。

- 在知识融合导航器中,将鼠标放置在“root”节点上并右键选择“添加子规则”。
- 在名称栏输入“my_block”。
- 在类列表框中选择“MyBox”。
- 点击“确定”。
- 在工具栏上,选择“适合窗口” 。

这样 mybox.dfa 程序就被执行了,并且创建了方块。

步骤 8: 在知识融合导航器中,核实信息。

在知识融合导航器中,展开 my_block,如图 1-4 所示。



图 1-4 知识融合导航器

步骤 9: 关闭文件。

第 2 章 知识融合语言基本编程

本章将对知识融合语言中的一些基本语法进行介绍。通过本章学习,能够使用以下知识融合语言的语法元素,如:类(Class),属性(Attribute),表达式(Expressions),信息传递(Information Passing),需求顺序(Demand Order),需求值(Demand Value)和参考框架(Reference Frames)。

2.1 类

类是已命名的能够为实体提供精确定义的规则的集合。类可以是诸如线或长方体等几何体,也可以是非几何体形态,例如:质量、约束等。类有获得参数值的特性。一般而言,类分为两种不同类型:用户自定义类和 NX 系统类。

1. 用户自定义类

在 NX 中,用户可以通过使用 KF 语言来创建用户自己的类。类保存在 DFA 文件中,而文件可以通过使用文本编辑器来创建编辑。DFA 文件能够包含多个类,但只有第一个类在知识融合导航器中可见。同时用户必须定义 KF 搜索路径,以使用户定义的类可以被 NX 发现并识别。

在有些情况下,将一个部件的所有几何描述压缩归为一个类来应用是恰当的。当用户在 NX 中将其实例化的时候,使用类生成的模型特征与这个类是密切关联的。

2. NX 系统类

NX 有着一个丰富的类函数库,可供用户使用、扩展和继承。多数系统类是以词头 nx_ 和 ug_ 来命名的,如: nx_arc、nx_axis、ug_block、ug_mass_properties 等。

用户可以使用系统类来创建线、基准平面、实体和装配体。类的使用也不仅仅局限于创建模型特征上。用户同样可以通过 KF 系统给予的功能融合自己的知识基础去创建更加自动化的程序。

2.1.1 DFA 文件

DFA 文件是让用户定义类和方法的源代码。

- 任何 DFA 文件必须以“#! NX/KF 6.0”语句开头。