

工序流引擎的设计及其在航空制造企业 MES 中的应用

金航数码科技有限责任公司 程思遥 张伦彦

针对目前航空制造企业 MES 系统现场执行过程管理存在的问题，从可配置性、控制逻辑和操作便捷性等方面对工序引擎的需求进行了分析，给出了工序流引擎的基于组件化的体系结构设计、工序流转的控制逻辑及管理工具设计方案，并对工序流引擎在某数字化 COE 项目中的应用情况进行了介绍。

航空制造企业车间生产管理本身的复杂性和多变性，决定了开发与实施 MES 时，须解决数据量大、业务流程

多样化、用户差异性显著以及二次开发量大等问题。目前 MES 大都强调了生产现场的即时数据采集、监督及控制，

此飞机起落架缓冲装置的外筒与活塞杆通过密封装置紧套，在外筒与活塞杆交接处还分别设有反向活门和凸轮，外筒内设有芯杆，芯杆下部穿过活塞杆，充气嘴与芯杆上部连接。

其中，芯杆上部设有孔洞，通过充气嘴将气体充入芯杆，气体和液体通过孔洞实现交换。芯杆底部与旋板上部分别设有螺纹和螺孔，并通过该螺纹和螺孔实现芯杆和旋板的连接。芯杆下部设有减摆阻尼孔，旋板上设有有一定数量的减摆阻尼孔，同时活塞杆内的旋板活动区域设有凸起。可配备多个旋板，每个旋板上设置的减摆阻尼孔空间的布置方式，孔径大小和数量等不同，形成了不同的阻尼系数，即每个旋板的减摆能力不同。凸起的尖端与旋板的中部设有较小缝隙，旋板左右活动距离需大于起落架最大缓冲行程。

此飞机起落架缓冲装置加工制作和装配使用方便快捷，无需单独设置减摆器及其配件，无需考虑起落架收起时，减摆器及摆振传递配件的收藏空间，同时将不再出现因摆振传递构件的疲劳损伤等原因导致的故障，降低了飞机故障率。

数周期后出现大幅度地向某一方向上偏移；旋板的速度整体上在零速度左右摆动，但其后期摆动幅度明显大于前期；因摆振周期和振幅以及液体粘度等因素的影响，同一减摆器旋板的位移出现大幅度偏摆的时刻以及其速度摆动的幅度不同。此液压减摆器的减摆特性对指导设计师设计减摆器具有重要作用。

(2) 本次测试试验表明在减摆器摆振过程中，可能也会出现类似于共振的现象，此现象对液压减摆器的减摆效果影响较大，故而设计师在设计液压减摆器时应考虑通过改变减摆器结构等手段，尽量降低共振可能带来的不利影响。

(3) 在液压传感器工作过程中，液压减摆器降低旋板的摆振幅度能力高于其减小旋板的摆振速度能力；同时增大液体粘度可一定程度上增强减摆器的减摆效果。

(4) 设计了一种具有减摆功能的飞机起落架缓冲装置，能够实现飞机起落架的降低冲击和减缓摆动的功能，无需单独设置减摆器及其配件，节约了成本，提高了减摆效率，实用性强，便于推广使用。

五、结语

(1) 在液压减摆器工作过程中，旋板的位移会在摆振

但在进行系统开发时，程序控制逻辑和业务流程绑定过于紧密，这导致了系统不够灵活，难以及时响应需求变更，这一问题在大量涉及工序流转逻辑控制的计划执行主线部分尤为明显。

为了解决目前 MES 开发中存在的问题，本文提出了一种工序流引擎技术，为车间现场执行过程中的工序流转提供底层支撑，同时作为一个开放式可定制的平台提供给用户，使其具有通过配置工艺路线模型来灵活调整工序流转控制逻辑的能力，从而适应现代企业经营过程管理的持续改善对 MES 功能的需求。

一、工序流引擎的需求分析

1. MES 在现场执行管理中存在的问题

作为典型的离散型制造企业，航空制造企业的 MES 需要针对不同生产方式的车间和多种类零部件的生产过程进行管理，面向的是复杂的工艺路线和多变的现场执行业务控制逻辑。航空制造企业现场制造执行过程的复杂性和多变性表现在以下几方面。

(1) 生产零部件种类及其工艺路线众多，由于车间生产现场管理流程和工序工艺要求的差异，对各工序的派工、完工工数据采集和质量检验条件的控制需求也是不同的，主要包括：①工作下达和分派策略不同，例如按工序顺序派工、按流卡多工序批量派工等；②工序间约束关系不同，例如有些简单的、加工周期短的工序，在进行完工工采集时可批量采集，以减轻工人录入采集数据的工作量，而对一些关键工序，则要求其完工检验之后，才允许后续工序进行完工工；③质量控制要求不同，例如一些普通工序不需检验，而有些关键工序可能需要工长、检验员甚至驻厂军代表等的多次检验和确认等。

(2) 执行过程中的变化和例外情况多，其生产状态受到外部需求的影响更为明显，主要包括零组件在管理方式不同的分厂、车间和工段间的周转、外协；一些工序对原材料进行切分（截料）导致的加工数量变化；以及质量问题导致的报废、降级使用，停工待处理以及返工返修；计划变更导致的流水卡片拆分合并、零件号变更等，这些都有可能导致执行控制逻辑的变化、工艺路线的调整和加工数量的变化。

以传统模式开发的 MES 在不同车间、不同企业进行实施时，以上差异和变化带来了大量的二次定制化开发工作，

降低了对需求变更的响应速度，造成了 MES 产品难以推广，系统模块复用率低，实施周期长、成本高等弊端。

为了解决这些问题，离散制造业的制造执行系统应具有更高的适应性和可配置性。然而，目前相关研究尚未受到足够重视。近年来，国内外研究者在如何提高 MES 组件化、可配置方面开展了部分研究，包括采用基于 CORBA 的分布式技术来提高 MES 与其他系统的互操作性，提出基于组件的可重构执行系统、可适应制造执行系统、基于构件复用技术的 MES 体系结构，以及结合组件技术、SOA 和 BPM 技术的开放式 MES 集成体系等。这些研究大部分着眼于通过信息技术改善 MES 的可集成性和模块级的可重构性，而缺少对 MES 现场执行的核心业务过程的提炼，难以对工艺路线乃至工序层次的业务需求差异和变化提供帮助。欧洲著名技术服务企业 Logica CMG 对 MES 的调查报告也显示，尽管国内外众多 MES 厂商针对不同的行业和生产模式提出了各自的解决方案，但当前主流 MES 系统的可适应性绝大多数体现在诸如报表定制等非核心业务环节，对生产需求和工艺流程变化的支持非常有限。

目前在航空制造企业实施的 MES 中，一般通过在数据库中定义大量的现场控制策略（进度采集影响策略、派工影响策略等）对工序流转进行控制，通过策略的组合应对不同的现场执行控制需求，这种方式虽然初步实现了控制逻辑的可配置性，但其逻辑十分繁琐，配置项互相依赖，理解和开发维护较为困难，一旦新增配置项，原有控制逻辑均需要进行修改；且在面向大量工序时，配置工作量大，在实施时为了减少维护工作量，最终往往简化为整个车间用统一的控制策略进行控制，不能满足精细化管理的需要。

2. 工序流引擎的设计目标

针对上述现状，本文对具有能够应对客户需求和工艺路线变化的可配置的工序流引擎技术进行了研究。为弥补现有制造执行系统的不足，工序流引擎主要是为车间现场执行过程提供支持，其设计应满足以下要求。

(1) 在可配置性方面，通过对工序内部工作流程及工序和工序之间的关系分析，抽取其中的内在逻辑，建立完备的工艺路线模型，力求以较少的独立的配置项组合出较多的适用场景。可根据现场实际业务，通过灵活便捷配置控制工序流过程。当现场的控制逻辑改变，或者应对不同实施单位对控制逻辑的不同要求时，只需通过改变工序的配置即可

满足新的需求，避免直接对系统代码进行修改，这是工序流引擎设计的根本目标。

(2) 在操作便捷性方面，在工序流数据准备阶段，为了避免工艺路线多、工序数量巨大导致配置工作量大的情况，引擎需提供通过预置的常用工序模板对工序进行一次性批量配置的功能，快速建立工序流。在引擎与 MES 结合运行的过程中，引擎用对相应角色推送待办的方式实现对工序派工、采集等各环节的控制，使各角色明确自己需要完成的工作，减少冗余信息，方便客户使用。

(3) 在功能设计方面，工序流引擎除了通过底层的逻辑控制工序流转外，还应与 MES 中的工艺路线管理模块相结合，为用户提供工序流维护功能，使用户的主工艺路线在引擎中均有对应的工序流描述信息并保证数据同步。为了方便引擎的开发维护和测试，还需提供设计器，可通过易于理解的、图形化的方式展示和调整工序流的配置。

总之，通过一系列丰富的控制功能，工序流引擎可用灵活直观的方式为不断变化的工序流转控制需求提供支持，当业务需求出现变化时，用户只需改变工序流配置即可满足新的需求，不需对数据库结构或系统代码进行任何新的修改及开发。

二、工序流引擎的体系结构

工序流引擎依托的 MES 产品采用组件化形式进行设计开发，每个业务模块为一个组件，组件运行时调用的后台控制逻辑封装为逻辑单元，展示页面封装为页面单元，工序流引擎为 MES 现场执行过程的控制提供支持，其体系结构如图 1 所示。为了便于与业务组件之间进行交互，工序流引擎提供了多个接口函数供逻辑单元进行访问。组件运行时，通过执行逻辑单元将发生的事件及产生的数据（参数）发送至工序流引擎，引擎将相应的执行新建和启动实例、基于业务

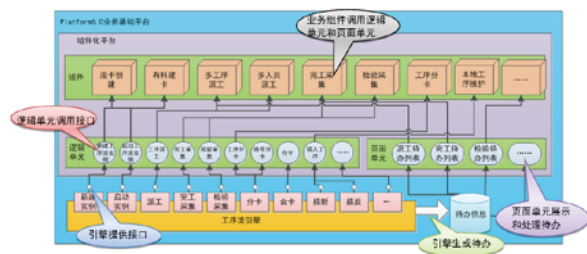


图 1 工序流引擎的体系结构

控制逻辑启动工序并向特定人员发送派工、采集或检验待办等操作。同时，页面单元将展示出工序流引擎产生的派工待办或采集待办等信息，在对待办进行操作后，调用逻辑单元与引擎进行交互，继续驱动工序流向下流转。

三、工序流转的控制逻辑

1. 工序信息的描述

工序流描述信息以配置项的形式展示给用户，信息以 xml 文件形式进行存储，包括基本信息、启动条件信息、群组信息、业务信息四部分。

(1) 基本信息定义了工序名称、投入产出比、工序内容等属性。

(2) 启动条件信息定义了工序启动的控制逻辑。工序的启动通过监听其前道工序的开工事件和自定义的其他监听条件（如生产准备检查等其他业务需求）来完成。当满足用户配置的事件触发条件时，引擎将启动工序并向指定用户发送派工或采集待办。因此在启动条件中，包含了以下 5 部分信息。①是否监听：定义工序是否对其紧前工序进行监听，对于不进行监听的工序，将无条件启动，这种情况在工艺路线较短，用时较少，允许工人不按顺序采集的零件加工中较为常见。②监听事件：定义工序启动的触发事件，包括其紧前工序的开始事件或紧前工序的完成事件。在实际业务中，开始事件可能是派工或开工，完成事件可能是完工或检验等，这取决于工序的群组信息配置。③监听数量：定义在监听事件上的发生数量，包括全部数量、任意数量和指定数量。选择“全部数量”则上一道工序开始或完成全部数量时，工序启动；选择“任意数量”则上一道工序开始或完成任意数量时，工序启动；选择“指定数量”则可根据具体业务来设置特定的数量，当上一道工序开始或完成该指定数量时，工序启动。④人工监听：设置工序启动监听人。设置人工监听后，工序启动前需要给设定的监听人发出待办，监听人同意后 will 启动工序，否则不启动工序。在实际业务中，这类工序可能是关键工序、中转工序等，需要在启动前进行人工审批确认等操作。⑤自定义其他监听：通过编写脚本的形式，定义其他非通用的监听事件。

(3) 群组信息用于控制每道工序本身派工、采集、检验的工作流程，主要定义了待办的接收人，包括是否派工、派工人、执行人、是否检验以及检验人信息。此外，定义了

工作项分派模式，有协同和竞争两种。协同即该工序由多人共同协作完成。竞争即该工序由多人互相竞争派工的数量。

(4) 业务信息则包含工序的设备、工时、工装目录等信息。

可以看出，工序流的流转方式主要由启动条件和群组信息进行定义，是工序流实例运行时决定控制逻辑的关键配置项。启动条件定义了工序间的流转逻辑，即工序节点的监听事件及工序启动时需要达到的数据状态，而群组则定义了工序内部的工作流程，包括是否派工，是否检验及接收待办的人员及人员之间的关系。其对工序流转的控制逻辑流程如图2所示。

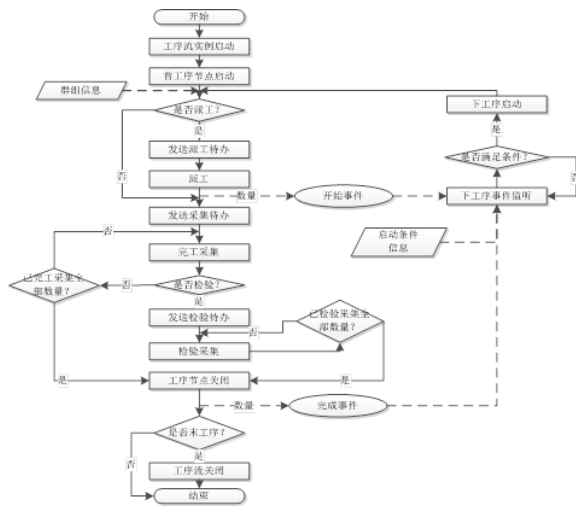


图2 工序流转的控制逻辑流程

2. 工序间流转数量控制逻辑

工序的启动条件信息定义了工序启动的监听事件和监听数量，而工序启动时的可开始数量受到其监听的前一道工序的事件上发生的数量的限制。当监听“开始”事件时，如果达到了工序的启动条件，工序的可开始数量为其前一道工序的开始数量；当监听“完成”事件时，工序的可开始数量为其前一道工序完成（完工或检验）的合格数量，工序流中的合格数量可能包括实际业务中工人自检合格数量、检验合格数量及让步使用、降级使用等可视作合格并继续向下流转的零件数量，可根据需求将相应数量作为参数传入工序流引擎。

例如，某工序监听其紧前工序的“开始”事件，数量限制为“任意数量”，则当上一道工序的开始（派工或开工）数量为5时，达到当前工序的启动条件，启动数量（可派工或可完工采集数量）为5，在用户进行派工或完工采集时，

数量不能大于5，如图3所示。

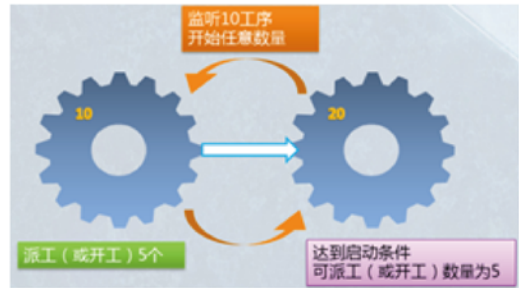


图3 工序流转数量控制逻辑

3. 示例

基于以上分析，以图4所示的某工艺路线为例，说明工序流的控制逻辑配置与实际业务中对现场执行过程的控制需求之间的关系。



图4 工艺路线图例

假设该工艺路线中20工序为关键工序，40工序为工艺段首工序。

对20的要求为：不派工，在前工序开始任意数量后即可生成采集待办，完工采集后需要进行检验，检验人为检验员A。

对30工序的要求为：不派工，20工序（因为是关键工序）完成全部数量后才能开工，不需检验。由于20工序需要检验，因此20工序的完成事件在检验采集时触发。

对40的要求为：派工，且必须前一工序完成全部数量后，40工序才能派工。派工人为工长B。派工前，需工段计划员C确认，不需检验。

对其余工序的要求均为不派工，前工序开始任意数量后即可开工。不派工的工序采集人均均为工人D，不需检验。

对应配置为如表1所示。

表1 工序节点配置示例

工序号	监听事件	数量限制	是否派工	派工人	执行人	是否人工监听	监听人	是否检验	检验人
20	开始	任意	否		工人D	否		是	检验员A
30	完成	全部	否		工人D	否		否	
40	完成	全部	是	工长B		是	计划员C	否	
其他	开始	任意	否		工人D	否		否	

四、工序流引擎的管理工具

为了便于对工序流进行管理和维护，并以图形化的方式编辑工序模板并对工序流及其实例运行情况进行直观的展示，产品中为工序流引擎开发了设计工具，包括工序流模板设计器、工序流设计器及工序流实例设计器，用户可通过这些工具方便的完成配置工作。

1. 工序流模板设计器

在工序流模板设计器中，用户可根据实施单位的具体业务预设多种类型的常用工序模板，如不派工任意采集工序、派工顺序采集工序、中转工序、检验工序和工艺段首工序等。预设好的工序模板可在维护工艺路线时批量生成工序流的各工序节点，以节省大量工序时的配置工作量，新生成的工序节点将完全继承模板的属性信息。设计器界面如图 5 所示。



图 5 工序流模板设计器

2. 工序流设计器

在生成工艺路线后，即可在工序流设计器中以图形化的方式查看各工艺路线对应的工序流信息，并对工序流属性进行编辑，可查看和调整工序流及各工序节点的属性。

3. 工序流实例设计器

MES 系统建立流卡时，将基于工艺路线版本对应的工序流生成工序流实例，工序流实例及其各工序节点完全继承来源工序流的属性，可在工序流实例设计器中查看和调整工序流实例及工序节点的属性，并进行分卡、合卡等单独针对实例的操作。

除设计器外，还对 MES 中的工艺路线管理模块进行了针对工序流的适应性修改，除了原有的对工艺路线及其各工

序信息的维护功能外，增加了为各工序设定工序模板和生成工艺路线对应工序流的功能，同时该模块可链接到各设计器，是工艺路线和工序流引擎的综合管理入口。

五、工序流引擎的工程验证

我公司在某航空发动机制造企业实施的数字化 COE（Center of Excellence，优良加工中心）项目是 MES6.0 产品的首个应用项目。作为平台级支撑，工序流引擎在 COE 项目中进行了验证。与工序流结合的业务组件包括投料建卡、多种类型的分合卡、本地工序维护、在制品派工、完工工采集等，通过对工序流的灵活配置，满足了不同车间多种零组件的现场执行控制需求，取得了良好效果。

六、结语

目前在航空制造企业 MES 的实施过程中，由于车间现场执行管理的复杂性以及不同产品、不同车间、不同企业对工序流转控制逻辑需求的差异性带来了大量二次开发测试工作，而传统的通过数据库定义控制逻辑的方式缺乏灵活性，且理解和维护难度较大，这些问题的存在降低了项目开发和实施效率。工序流引擎作为 MES 系统执行主线部分的重要支撑，通过图形化、可配置的方式，较好地解决这一问题，实现了对需求变更的快速响应，并在某航空制造企业中的应用中进行了验证。