

基于 ECA 规则的动态数据交换技术

周 伟¹, 陈小安¹, 罗天洪²

(1. 重庆大学机械传动国家重点实验室, 重庆 400044; 2. 重庆交通大学机电学院, 重庆 400074)

摘 要: 动态数据交换是实现网络协同设计的关键技术, 在协同设计中每一步的设计信息都进行实时交换必然导致在动态数据交换中包含不必要的数据。为了解决这个问题, 该文提出基于 ECA 规则的动态数据交换技术。建立基于 ECA 规则的动态数据交换系统结构, 并定义动态数据交换中的 ECA 规则和动态数据更新中事件间的基本关系。应用该技术实现了跨平台异构 CAD 系统之间动态数据的实时交互和一致性。

关键词: 网络协同设计; 动态数据交换; ECA 规则

Dynamic Data Exchange Technique Based on ECA

ZHOU Wei¹, CHEN Xiao-an¹, LUO Tian-hong²

(1. State Key Lab of Mechanical Transmissions, Chongqing University, Chongqing 400044;

2. Mechanical and Electrical Engineering College, Chongqing Jiaotong University, Chongqing 400074)

【Abstract】 The dynamic data exchange is the key technique to realize the Internet supported collaborative design. But the information of each step designing in collaborative design needs to exchange real timely, so many unnecessary data must be contained in dynamic data exchange format. To resolve the problem, the dynamic data exchange technique based on ECA is proposed. The system structure of dynamic data exchange based on ECA is constructed. The ECA rule in dynamic data exchange and the basic relation of events are defined. The dynamic data exchange technique based on ECA is applied to realize the real-time exchange and consistency of the dynamic data between the heterogeneous platforms and CAD systems.

【Key words】 network-based collaborative design; dynamic data exchange; Event-Condition-Action (ECA)

网络协同设计中数据交换的特点主要有: 产品数据复杂多样, 产品数据交换频繁、流量大, 产品数据的一致性要求高, 产品数据的并发性访问频繁。这些特点使得数据在网络协同设计中难以得到实时交互^[1]。而动态数据交换可以克服这些特点带来的问题。ECA 规则是主动数据库的研究和应用过程中普遍使用的一种控制和规范主动数据库行为的规则, 由 3 个部分构成: 事件, 条件和动作^[2]。在动态数据交换中包含了不必要数据, 因此需要提供一种方法来监控和仲裁在异地实时协同设计中的动态数据交换。本文提出基于 ECA 规则的动态数据交换技术, 从而在网络协同设计过程中减少数据信息的网络传输量以及在更新数据中降低不必要数据产生的可能性。

1 基于 ECA 规则的动态数据交换系统结构

基于 ECA 规则的动态数据交换系统结构如图 1 所示。

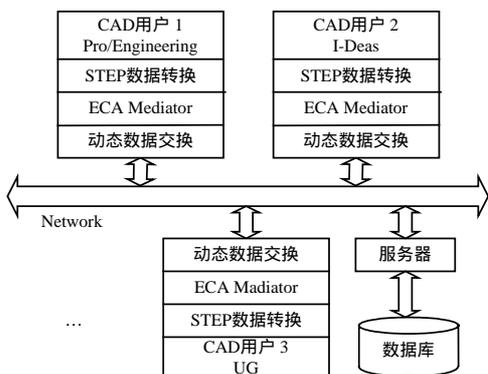


图 1 基于 ECA 规则的动态数据交换系统结构

本文在网络实时协同设计系统中应用 ECA 规则来监控和仲裁动态数据交换。更新的数据在本地被识别然后通过 ECA 规则自动实时地传输给其他设计者。

系统采用 C/S 体系结构, 由服务器、数据库和处于不同地理位置的协同用户组成。在系统结构中, 服务器对用户访问权限、数据存储、数据查询、版本控制等进行管理。每个用户端由 CAD 软件、STEP 数据转换、ECA Mediator 和动态数据交换 4 个模块组成。

1.1 STEP 数据转换

协同环境中的 CAD 用户, 各自有自己的数据格式。为了在不同的系统间进行准确无误的数据交换, 需要对模型数据进行 STEP 数据转换。现今大多数的 CAD 系统都能够输出、输入零件或产品的 STEP 模型。可以集成 Java Servlet 服务构件对模型数据自动进行 STEP 数据转换。

网络协同设计系统对 STEP 数据转换的方式为: STEP 前处理器把某一 CAD 系统(A)内的数据转换成符合 STEP 交换结构语法的文件(交换文件), STEP 后处理器读入 STEP 文件, 把交换结构描述的数据转换成接收系统(B)内的数据^[3]。

1.2 ECA Mediator

在协同设计过程中, ECA Mediator 监控实体模型 STEP 数据并根据 ECA 规则识别更新数据。当数据更新被监测到

基金项目: 国家自然科学基金资助项目(50175113); 高等学校博士点专项科研基金资助项目(20040611006)

作者简介: 周 伟(1978 -), 男, 博士研究生, 主研方向: 网络协同设计; 陈小安, 教授; 罗天洪, 副教授

收稿日期: 2007-09-17 **E-mail:** cqzw2001@126.com

时,ECA Mediator 根据 ECA 规则自动在实体模型 STEP 数据上识别更新,并只传送更新的数据给动态数据交换。另一方面,当 ECA Mediator 接收更新数据时也根据 ECA 规则把数据传送给 STEP 数据转换。图 2 显示了实体数据在从 STEP 数据转换传送给动态数据交换时 ECA Mediator 识别实体数据更新的程序流程图。STEP 数据转换从动态数据交换接收更新数据时,ECA Mediator 识别实体数据更新的程序流程图与此类似。

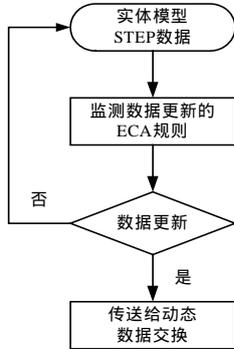


图 2 ECA Mediator 识别实体数据更新的程序流程

ECA Mediator 中的关键是更新数据的识别,也即是监测数据更新的 ECA 规则定义,它是整个系统的核心技术之一。

1.3 动态数据交换

在课题组前期研究成果的基础上^[4],本文提出了STEP动态数据交换格式来进行动态数据交换。STEP动态数据交换格式由数据段和控制段组成,如图 3 所示。控制段包含 User ID,实体序列号和编辑命令。User ID和实体序列号可以保证网络协同设计系统在各协同设计用户之间唯一标识一个实体。编辑命令一般包含insert, delete和modify这 3 种命令。数据段中的数据格式是STEP数据格式,数据只是ECA Mediator识别的更新数据。STEP动态数据交换只对更新的数据对象进行操作,减小了网络传输的负担,提高了数据交互效率。

控制段	User ID
	实体序列号
	编辑命令	insert, delete, modify
数据段	STEP 数据	

图 3 STEP 动态数据交换格式

2 动态数据交换中的 ECA 规则定义

如果在网络协同设计中每一步的设计信息都进行实时交换,那必然在动态数据交换中包含了不必要的信息。因此需要准确定义 ECA 规则来监测和识别实体数据来减少数据信息的网络传输量以及在更新数据中降低不必要数据产生的可能性。

ECA可定义为 $R = (E, C, A)$ 。其中, E 为激活该规则的事件; C 为条件集,用以反映系统和环境中的不同情况; A 为操作集(或动作集合)^[5]。

定义 1 在动态数据交换的 ECA 规则中, E 具体来说就是数据更新,它是一个函数,将时间映射为布尔值,可表示为 $E: T \rightarrow \{True, False\}$

$$E(t) = \begin{cases} True & \text{if 数据更新在 } t \text{ 时发生} \\ False & \text{其余情况} \end{cases}$$

式中, t 为数据更新 E 的发生时间。

数据更新可以分成原子数据更新和复合数据更新。原子数据更新是那些可以直接被探测到的数据更新,包括几何图

形、图像、文档、计算参数等数据的更新。为了表达动态数据交换执行中的复杂逻辑,原子数据更新可以复合,可以用一些基本的复合算子,例如 AND, OR, PRE, NOT, REP, ANY 等来表示原子数据更新复合。

为了表示方便,还可以在上述基本算子的基础上,定义其他算子。例如:

(1)ALL(多元并)。ALL(E_1, E_2, \dots, E_n)= E_1 AND E_2 AND ... AND E_n ;

(2)SEQ(多元序列)。SEQ(E_1, E_2, \dots, E_n)= E_1 PRE E_2 PRE ... PRE E_n ;

(3)EXC(互斥)。EXC(E_1, E_2, \dots, E_n)= $(\text{NOT } E_1 \text{ AND } E_2 \text{ AND } \dots \text{ AND } E_n) \text{ OR } (E_1 \text{ OR } E_2 \text{ AND } \dots \text{ AND } E_n) \text{ OR } \dots \text{ OR } (E_1 \text{ AND } E_2 \text{ AND } \dots \text{ OR } E_n)$ 。

将通过原子数据更新和复合算子形成的表达式代表的更新称为复合数据更新,表达式称为复合数据更新表达式。

性质 1 假定 E_i 是复合数据更新表达式 E 中的一个元素,如果 $E_i(t) = \text{true}$, 且 $\forall t_1, t_1 < t, E(t_1) = \text{False}, E(t) = \text{True}$, 则复合数据更新的发生时间记为 t_i 。

定义 2 复合数据更新表达式中包含的原子数据更新的个数,记为 $\text{dim}(E)$ 。

定义 3 数据对象变量是动态数据交换中涉及的数据对象的属性或参数,简称变量。变量以 x_i 来表示,变量集合记为 $X = \{x_1, x_2, \dots, x_k\}$ 。

定义 4 变量 x_i 的取值范围为域,用符号 D_i 来表示。

定义 5 条件 c_i 是 $X_i = \{x_{i1}, x_{i2}, \dots, x_{ik}\} \subseteq X$ 上的 k 元关系,是笛卡尔积 $D_{i1} \times D_{i2} \times \dots \times D_{ik}$ 上的子集。条件 c_i 是对变量之间关系的限制。条件集合记为 $C = \{c_1, c_2, \dots, c_k\}$ 。条件 c_i 可以通过条件表达式来描述。

定义 6 操作可表示为 $A = (E, T_a, O_b, T_o)$, E 为触发操作的事件, T_a 为操作的类型, O_b 为操作的对象, T_o 为操作的工具。

定义 7 数据更新的缺省状态有等待(Waiting, W),开始(Beginning, B),挂起(Hanging, H),执行(Executing, Ec),完成(Completed, Co),超时(Time-out, T),放弃(Abandon, A),其状态集 $S = \{W, B, H, Ec, Co, T, A\}$ 。图 4 所示为数据更新活动状态图。

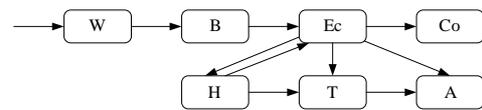


图 4 数据更新活动状态图

定义 8 数据更新事件包括原子数据更新和复合数据更新,因此在数据更新活动过程中,活动可以被分解,即 $p = (P, R_c, R_m)$, P 为子活动集合, R_c 为更新活动控制规则, R_m 为监控规则。被分解的活动称为复合活动。 R_c, R_m 为ECA规则。活动进行层次分解后,需要一定的机制来保证模型之间的一致性:

规则 1 父活动的状态由其所包含的子活动状态来决定。

规则 2 父活动的资源与人员是所有子活动的资源和人员的总和,子活动的时间属性需要满足父活动的时间属性的约束。

在动态数据更新过程中,存在一个数据更新事件的发生会导致另一个数据更新事件的产生这种情况。对于这种类似情况,引入触发的概念来说明事件间的关系。

定义 9 事件 E_i 的发生引起另一事件 E_j 的发生,称 E_j 被 E_i 触发,而 E_i 被称为 E_j 的触发条件,记为 $E_i \rightarrow E_j$ 。

这样,通过事件的复合运算和触发关系,就可以表达动态数据更新中定义的事件间的基本关系。

在数据更新过程中,ECA Mediator 中的事件检测器首先直接探测各种原子数据更新事件,然后根据 ECA 规则判断某些条件表达式值是否为真。如果为真,则执行相应的动作,系统执行后,又将引发新的事件。

在某些数据更新原子事件执行后,数据更新复合事件的表达式可以简化。这样,下一次相关事件发生后,只需检查简化后的表达式,提高了处理效率。根据这一思想,基于事件表达式重写方法的事件判断方法被提出。

定义 10 假定 $E_1 \in E$,则 E_1 发生后, E 变化为 E' ,称 E' 为 E 的余,记为 $E' = E/E_1$ 。假设求余运算的事件表达式中包含的事件在 E_1 发生前均没有发生,则有:

- (1) $E/E_1 = E$, if $E_1 \notin E$;
- (2) $(E_1 \text{ AND } E_2)/E_1 = E_2$;
- (3) $(E_1 \text{ OR } E_2)/E_1 = \text{True}$;
- (4) $(E_1 \text{ PRE } E_2)/E_1 = E_2$;
- (5) $(E_1 \text{ PRE } E_1)/E_1 = \text{False}$;
- (6) $\text{REP } E_1 n / E_1 = \text{REP } E_1 (n-1)$;
- (7) $\text{ANY } (m, E_1, E_2, \dots, E_n) / E_1 = \text{ANY } (m-1, E_2, \dots, E_n)$;
- (8) $\text{NOT } E_1 / E_1 = \text{False}$ 。

性质 2 $\dim(E/E_1) < \dim(E)$ 。

根据上述求余运算的法则,该性质是显然的,说明通过求余运算,事件表达式得到简化。假如事件表达式为 $E = (E_1 \text{ AND } E_2) \text{ PRE } (E_3 \text{ OR } E_4)$,假定 E_1, E_2, E_3, E_4 依次发生,则有:

E_1 发生后, $E' = E_2 \text{ PRE } (E_3 \text{ OR } E_4)$;

E_2 发生后, $E'' = E_3 \text{ OR } E_4$;

E_3 发生后, $E''' = \text{True}$ 。

通过上述动态数据交换过程中的动态数据交换事件、条件和动作的定义,有关规则和性质的描述,就可以准确定义 ECA Mediator 中的 ECA 规则:

```
WHEN ( $E(t_1) = \text{True}$ )
IF ( $E(t_2) = E(t_1)$ )
  THEN (Set_Executing S) and (传送更新数据至动态数据交换)
ELSE (Set_Abandon S)
END IF
```

其含义为:数据更新事件 E 在时间 t_1 发生,经过一定时间(本文定义这里时间为 60 s)在时间 t_2 数据更新事件 E 没有发生变化,这时就可认为设计人员完成对该数据的修改,执行该数据更新活动 S 并把更新的数据传送至动态数据交换,如果不满足该条件则放弃该数据更新活动。

3 网络协同设计动态数据交换的实现

应用本文所提出的基于 ECA 规则的动态数据交换技术可以成功实现网络协同设计模型数据的实时共享和交互,其过程如下:

(1)设计人员通过网络异地对同一件产品进行设计,通过会议对产品的修改意见达成一致。

(2)设计人员对产品进行修改,该客户端的 STEP 数据转换完成产品的 STEP 数据转换。

(3)该客户端的 ECA Mediator 监控产品模型 STEP 数据并根据 ECA 规则识别更新数据,并只把更新数据传送至动态数据交换。

(4)动态数据交换生成 STEP 动态数据交换格式,该数据格式数据段只包含了产品更新的数据,然后向服务器和其他协同设计人员传送该数据。

(5)服务器和其余客户端根据 ECA 规则接收更新数据,以 STEP 格式保存在服务器数据库和客户端的子数据库上。

(6)其余客户端 STEP 数据转换把 STEP 更新数据转换成各自 CAD 系统内的数据。

以 2 个设计者利用 Pro/E 和 UG 作为建模工具对一部手机进行实时协同设计为例来说明上述过程。设计者 A 和 B 分别用 Pro/E 和 UG 对手机进行协同设计,2 个设计者通过协同会议讨论决定将手机天线的长度由 15 mm 修改为 20 mm。设计者 A 对手机天线进行修改后,其客户端的 ECA Mediator 对手机模型进行监测和识别,只把手机天线的更新数据传输给 STEP 动态数据交换进行数据转换。接着更新数据的 STEP 动态数据交换格式传送给设计者 B,设计者 B 端接收更新数据后,手机天线长度已改为 20 mm。

4 结束语

本文根据网络协同设计中数据交换的特点,提出了基于 ECA 规则的动态数据交换技术。首先建立基于 ECA 规则的动态数据交换系统结构,通过 STEP 数据转换,ECA Mediator 和动态数据交换 3 个模块来实现协同设计中的动态数据交换,并提出了 STEP 动态数据交换格式。通过定义动态数据交换过程中的动态数据交换事件、条件和动作以及描述有关规则和性质来准确定义动态数据交换中的 ECA 规则,并通过事件的复合运算和触发关系表达动态数据更新中定义的事件间的基本关系。应用基于 ECA 规则的动态数据交换技术能够在网络协同设计中减少数据信息的网络传输量以及在更新数据中降低不必要数据产生的可能性。

参考文献

- [1] 罗天洪, 陈小安, 林利红, 等. 网络协同设计动态数据交换系统[J]. 重庆大学学报, 2003, 26(12): 1-5.
- [2] 樊晓平, 罗 熊. 基于 ECA 规则实现 PVR 系统中任务演绎和动作规划的新方法[J]. 铁道科学与工程学报, 2005, 2(4): 82-87.
- [3] Yeh S C, You C F. Combining Express and UML to Implement a Step-based System[J]. International Journal of Computer Applications in Technology, 2002, 15(1/3): 98-108.
- [4] Luo Tianhong, Chen Xiaolan, Lin Lihong, et al. Dynamic Data Exchange in the Collaborative Design Based on Internet Environment[C]//Proc. of Europe-Asia Symposium on Advanced Engineering Design and Manufacture for Globalization. Xi'an, China: [s. n.], 2004: 327-336.
- [5] 高秀峰, 胡昌振. 基于 ECA 规则的入侵检测研究[J]. 计算机工程. 2005, 31(16): 132-134.