

Simio 文档

所有步骤

所有步骤库 (A 到 Z) 包括 Simio 所有的过程步骤。这些步骤为仿真提供了运行时逻辑。包括:

分配(Allocate)步骤用于将一个父对象的可用资源容量 (如果有的话) 手工分配给其他等待使用这些容量的对象。

到达(Arrive)步骤由实体对象用来通知其接受的访问请求, 该实体已经到达了它的当前节点。

赋值(Assign)步骤用于为状态变量赋一个新的值。

合并(Batch)步骤用于连接批量逻辑元素来和多实体相匹配, 从而把那些实体组成批量, 然后把批量的成员附属在父实体中。

消耗(Consume)步骤消耗指定的物料数量——或者只是在顶部——或是在物料分解单。

新建(Creat)被用作创建新的对象来实现指定对象的种类。

判断(Decide)步骤被用于以过程逻辑决定一个符号的工作流程。

延迟(Delay)步骤延迟指定时间段到达的符号。

销毁(Destroy)步骤销毁父对象或执行符号的相关对象。

断开连接(Disengage)步骤用于解开父连接的一个实体的位置, 这样连接和实体就可以独立地进行移动。

放下(Dropoff)步骤被运输器对象用作放下运动中的实体, 并把它放在运输器目前的节点上。

结束活动(EndActivity)步骤用来结束操作中当前的活动。

结束操作(EndOperation)用来结束操作。

结束转移(EndTransfer)步骤说明和执行符号有关的实体对象已经完成转移到一个对象或站中。

连接(Engage)步骤用于封锁父连接对象的一个实体的位置。

执行(Execute)步骤用于执行一个指定的过程。

故障(Fail)步骤计算故障时间。

激发(Fire)步骤用于激发一个独享事件。

插入(Insert)步骤用于把一个对象插入指定的队列中。

中断(Interrupt)步骤可用于中断过程的延迟。

移动(Move)步骤可用于请求一个或多个可移动资源的移动, 而这些资源已被其父对象或与执行令牌相关的对象所捕获。执行令牌被保留在移动步骤内, 直到资源到达了请求的位置。

通知(Notify)步骤用来输出一个用户自定义的跟踪信息或警告信息。

停靠(Park)步骤用于把一个实体对象移动到节点的停驻位置。

拾取(Pickup)步骤用于运输器对象拾取一个目前处于等待队列节点中的实体。

计划访问(PlanVisit)步骤用于运输器对象搜索并接受系统中的拾取预留请求。

生产(Produce)步骤生产指定的物料数量, 或者只从顶级生产, 或者从物料的分解单生产。

释放(Release)步骤释放一个或多个对象的容量。这些对象代表父对象或和执行符号有关的对象。

修复(Repair)步骤是故障时间的终止。

移出(Remove)步骤用于从指定的队列中移出对象。

恢复(Resume)步骤用于继续一个过程或继续对象的移动。

搭载(Ride)步骤用于初始化一个运输器关于搭载某节点上某一实体的行驶请求。

路线(Route)步骤和路线集合元素一起使用，来为一个实体对象从多个候选节点中选择一个到达目的地的路线。

搜索(Search)步骤用于搜索对象的集合。

捕获(Seize)步骤用于捕获一个或多个对象的容量。这些对象代表父对象或和执行符号有关的对象。

选择放下(SelectDropoff)步骤用于为正在运行中的实体设置其父运输器对象的目的地节点。

选择访问(SelectVisit)步骤用于设置父运输器对象的目的地节点来拾取目的地。该目的地被计划访问步骤所接受。

设置网络(SetNetwork)步骤用于设置行驶于节点的实体所使用的连接网络。

设置节点(SetNode)步骤用作设置任何实体对象的目的地节点。

设置表格(SetTable)步骤用于为符号或对象设置数据表格或序列表格。

开始活动(StartActivity)步骤开始保留操作的下一个活动。

开始作业(StartOperation)开始一个操作。

预设事件(Subscribe)步骤用于为过程增加一个新的触发事件，即如果事件发生该过程将被执行。

暂停(Suspend)步骤用于种植过程或对象的移动。

计数(Tally)步骤记录每一个到达符号的观察值。

转移(Transfer)步骤用于转移对象和空白空间之间的，和执行符号有关的实体对象。

拆分处理(UnBatch)步骤用于将批处理成员从和执行符号有关的父实体对象中移开。

结束停靠(UnPark)步骤用于将在节点位置停驻的实体对象移开。

取消预设事件(UnSubscribe)步骤用于为过程取消一个触发事件。

访问节点(VisitNode)在节点逻辑的范围内，用于初始化和执行符号有关的实体的访问节点过程。

等待(Wait)步骤用于持有该步骤中到达的符号直到指定的事件发生。

请将关于该话题的评论发邮件给我们。

2006-2011版权所有, **Simio LLC**有限公司保留一切权利。

Simio 文档

用户自定义过程步骤

Simio基于微软最新的.NET4开发技术，提供了一种开放式的架构，允许用户在系统中增加新步骤和元素。步骤和元素可以用任何.NET语言（Visual Basic， C#， J#等等）编写。您只需添加您的动态链接库*.dll文件到Simio的UserExtensions文件夹。了解更多关于如何创建用户自定义的过程步骤，请参考 [定制化的Simio扩展](#)和Simio 应用程序接口说明书（**C:\ProgramFiles\Simio\Simio API Reference Guide.chm**）。参阅[Simio Visual Studio模板](#) 了解如何使用Simio提供的开发模板开始创建自定义步骤。Simio提供了一些用户自定义步骤的实例，包括[关门（CloseGate）](#)，[开门（OpenGate）](#)，[从门穿过（PassThruGate）](#)，[读取数据（Read）](#)和[写入数据（Write）](#)。

请将关于该话题的评论发邮件给我们。

2006-2011版权所有, **Simio LLC**有限公司保留一切权利。 迅合科技对译文保留版权。 **2011**

Simio 文档

关门

关门

关门这一用户定义步骤用于关闭一个指定的入口名称。

关门步骤需与开门和从门通过中的用户定义步骤以及二进制门中的用户定义元素配合使用。

下表是关门的属性：

属性	描述
门	要关的门的名称

Simio 文档

开门

开门

开门这一用户定义步骤用于打开一个指定的入口名称。

开门步骤需与关门和从门通过中的用户定义步骤以及二进制门中的用户定义元素配合使用。

下表是开门的属性：

属性	描述
门	要开的门的名称

Simio 文档

从门通过

从门通过

从门通过这一用户步骤允许符号在门打开的情况下通过指定的门。

从门通过步骤和开门，关门用户定义以及二进制门用户定义步骤元素联合使用。

下表是从门通过的属性：

属性	描述
门	要通过的门的名称。如果门目前是关的，符号将等待直到门打开

Simio 文档

读取

读取

读取这一用户定义步骤用于从文件中将数据读入模型。

读步骤和文件用户定义元素一起使用。如果分离器属性为空白，则文件中的整条线将被读入同一个状态。

为了能够读入个体值，可将合适的字符输入分离器属性。例如，.csv 或.xls 文件应有逗号隔开，并列入分离器属性，个体就可以被读入不同的状态。注意，如果您在状态中读入了值，值必须是数字的，因为状态里不能有非数字字符。

下面是读的属性：

属性	描述
文件	将要读入的文件名称
分离器	文件中将数字分离出来的字符
状态	将要被读入的状态值

Simio 文档

写入

写入

写入这一用户定义步骤用于从一个模型向文件写入数据。当向文本文件写入的时候，要从指定的项目属性中指明一个项目，并使用花括号{}。{0}可用于指明第一个项目，{1}指明第二个项目，以此类推。例如，下面的这个句子可以在属性中指定：在时间{2}有类型{1}的{0}部分。当写入.csv类型的文件时，格式部分可以留白，因为它默认使用常用分离格式。

写入步骤和文件用户定义元素一起使用。

下表是写入的属性：

属性	描述
文件	将要写入的文件名称
格式	提供将要写入的串的格式。如果没有提供，则将使用逗号分离格式
项目	指定的，与将写入值有关的项目

Simio 文档

过程步骤指南

步骤指南部分有 Simio 的全部过程步骤的详细信息。

Simio 文档

分配(Allocate)

分配

分配步骤用于将父对象的可用资源能力（如果有的话）手工分配给等待使用这些能力的对象。

Simio 文档

到达(Arrive)

到达

到达步骤被实体对象所使用，用来通知其已接收的访问请求，告知该实体已经‘到达’它当前位置。

注意:该步骤只能在实体内部被使用。

Simio 文档

赋值(Assign)

赋值

赋值步骤被用于为状态变量赋予新的值。

指定的状态可以被令牌，访问实体，父对象或成员元素所拥有。

关于如何使用赋值步骤，参考小模型 [LearningCurveWithFunction](#), [LeveledArrivals](#), [LogicBasedOnEntityState](#), [RecordDistanceTraveled](#) 和 [SimpleTank](#)。

下表是 **赋值** 的属性：

属性	描述
状态变量名称(State Variable Name)	将被赋予一个新值的状态变量名称
新值(New Value)	将被赋予的新值
赋值 (更多) , Assignments(More)	一系列重复的赋值
赋值 (更多) 状态变量名称, Assignments(More).State Variable Name	将被赋予一个新值的状态变量的名称
赋予 (更多) 的新值, Assignments(More).New Value	将被赋予的新值

[请将关于该话题的评论发邮件给我们。](#)

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

合并 (Batch)

合并

合并步骤和批量逻辑元素(BatchLogic Element)一起使用，用来将多个实体相匹配，将那些实体合并成批，并将合并的成员附加在父实体上。

下表是**批量**的属性：

属性	描述
批量逻辑元素名称 (BatchLogic Name)	为此' 合并' 步骤定义' 合并逻辑' 的批量逻辑元素名称
种类(Category)	指定是父实体或是成员实体 (将被附加在父实体上) 执行该合并步骤

[请将关于该话题的评论发邮件给我们。](#)

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

消耗 (Consume)

消耗

“消耗”步骤可以消耗掉指定数量的物料——第一层物料，或者展开的物料清单。在第一层的话，这些指定数量的物料被直接消耗掉。在物料清单层，列在“物料清单”中的物料被令牌所消耗，这个数量等于清单中的数量乘上第一层的数量（通常为1）。令牌在这一步骤等待，直到所有数量都被分配完毕。当物料逐步

到达，分配量也会逐步的增加。只有第一个在物料队列中等待的实体才能够得到分配。消耗步骤和生产步骤（Produce）一起使用，如果在消耗步骤中的所有物料清单的数量还不可获得的话，令牌在该步骤等待。在模型的其他地方，应使用生产步骤增加物料数量，一旦生产完毕，消耗步骤立即使用这些产生的物料。

下面列出的是 **消耗** 的属性：

属性	描述
消耗物种类(Consumption Type)	一种用来指定需要被消耗的物料的方法。
物料名称(Material Name)	在这一步骤要被指定消耗的物料，或者要被消耗的物料清单。
数量 (Quantity)	消耗的数量。
活动名称(Activity Name)	具有该物料消耗信息的活动名称。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

新建 (Create)

新建

“新建”步骤可以用来新建指定的‘对象实例’类型的新的‘对象实现’。与新创建的对象关联的令牌将退出‘Created’退出点。如果您使用‘CopyParentObject’来指定创建类型（CreateType），那么所有状态值会与原对象复制到新对象。如果您使用‘NewObject’，新创建的对象会拥有系统默认的状态值。请注意，退出‘Created’退出点的新令牌会拥有和原执行令牌完全相同的状态值（不管是‘复制’还是‘新建’）。因此，如果创建了一个‘新对象’，这可以让您使用令牌的状态（只要您愿意）作为一个临时存放点，用来复制选中的状态值。

当使用CreateType=‘NewObject’创建对象时，它被放在设施视图的自由空间内，在对象实例的位置。如果使用CreateType=‘CopyParentObject’或‘CopyAssociatedObject’创建对象时，新对象初始位置和被复制的原对象的位置相同，且在自由空间。因此，您需要在‘新建’步骤之后，直接使用Transfer步骤，将新对象放入到一个指定的位置，比如一个连接上，或者一个站点或节点上。

从新建步骤出来有两个退出点，包括原始退出点和Created退出点。新建实体的令牌将首先从‘Created’退出点退出，然后继续前进直到碰到该令牌或者相关联的实体对象的延迟类型的步骤。然后，原始的令牌将从‘Original’退出点离开。可以使用函数Math.Epsilon作为延迟步骤的时间，用来延迟被创建的令牌，从而影响事件处理的次序。

关于新建步骤的使用，参考小模型CONWIP和DynamicallyCreatingVehicles。

下面列出的是 **‘新建’** 的属性：

属性	描述
----	----

新建类别 (Create Type)	创建的方法。‘新对象’ 创建指定实例类型的新对象。‘复制父对象’创建父对象的副本。‘复制关联对象’创建该‘新建’ 步骤执行令牌的相关联的对象的副本。
对象实例名称 (Object Instance Name)	对象实例类型，可以以此创建新的‘对象实现’。
对象数量(Number of Objects)	要创建的对象数量。这个属性应该特指取整的表达式。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

判断(Decide)

判断

“判断”步骤可以通过过程逻辑来确定令牌的流向。“判断”步骤判断一个表达式，所以在该步骤被引用的属性必须是表示式类型的属性。

到达的令牌退出‘True’或‘False’退出点是基于指定的概率或者条件。

关于如何使用“判断”步骤，请参考小模型 [SequentialProcessingByBatchSpecifiedInTable](#) 和

[MoveableOperator](#)。

下面列出的是**判断**的属性：

属性	描述
判断类型 (Decide Type)	表明了这个决策是基于某种概率还是某种条件。
表达式 (Expression)	由某种概率或者条件形成的表达式。如果是一种概率，那么输入 0 到 1 之间选择‘真’值的概率（包括 0 和 1）。如果是一种条件，则输入逻辑条件。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

延迟 (Delay)

延迟

“延迟”步骤根据指定的时间长度，延迟到达的令牌。

注意：如果延迟时间表达式返回一个负数值，那么 Simio 将产生一个错误。通常您在延迟步骤的 Delay Time 属性里使用了无界的分布函数，比如正态分布，有可能产生负数。参考 [分布函数](#) 部分了解更多内容。

下面列出的是**延迟**的属性：

 Forward Thinking		
新一代仿真技术 基于智能对象的仿真	Simio 用户参考指南(过程步骤) (迅合科技 版权所有)	版本 3.53.6762 2011 年 5 月

属性	描述
延迟时间(Delay Time)	延迟的时间长度。这个属性可以使用服从某种分布的随机样本来指定。
可中断的(Interruptible)	表明在该步骤被延迟的令牌可以通过 Interrupt 步骤进行中断。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

销毁(Destroy)

销毁

“销毁”步骤用来销毁父对象，或者执行令牌相关联的对象。

只有支持动态创建的对象类型才能够被销毁。被销毁对象的所有过程被立即终止。所有计划事件（例如步骤的延迟）从日历上被取消。如果父对象被销毁，则令牌并不退出这一步骤。

下面列出的是**销毁**的属性：

属性	描述
销毁类型(Destroy Type)	要被销毁的对象，特指与执行令牌相关的对象或者其父对象。

Simio 文档

断开连接 (Disengage)

断开连接

“断开连接”步骤用来将实体的位置从其父对象连接的位置上解锁，使得连接和实体都可以独立地移动。

“Engage”步骤则用来连接一个实体。

一旦解除锁定，实体就可以自由地以自己的速度从连接上穿过，而连接也能按照自己的速度、不受实体移动（或实体停止）的影响自由移动。

注意：本步骤只能在连接对象内部使用。

Simio 文档

放下 (Dropoff)

放下

“放下”步骤由一个运输器对象使用、将装载的实体放下到运输器目前的节点。

使用“Pickup”步骤来捡起一个等待装运的实体。

注意：如果延迟时间表达式产生负值的话，Simio 将产生一个错误信息。当您在“放下”步骤的卸载时间属性（**Unload Time**）使用了无界的概率分布，比如正态分布函数，可能会产生负数。查看[分布函数](#)了解更多

信息。

下面列出的是**放下**的属性：

属性	描述
堵塞的行动(Blocked Action)	堵塞的行动 特指如果一个实体被卸载到一个外部节点的相关对象、而该对象没有足够容量立即接受该实体时,应该采取的措施(等待或失败)。
卸载时间(Unload Time)	从运输器上卸载实体所需要的时间。

注意：该步骤只能在运输器对象的内部使用。

Simio文档

结束活动 (EndActivity)

结束活动

“结束活动”步骤在一个作业内结束当前活动。和作业元素 (Operation) 和活动元素 (Activity) 配套使用。

下面列出的是**结束活动**的属性：

属性	描述
作业 (Operation)	预留的作业名称

Simio文档

结束作业 (EndOperation)

结束作业

“结束作业”步骤用于 结束某个操作。

该步骤释放父类资源，并将一切冲突的作业移至以后。

下面列出的是**结束操作**的属性：

属性	描述
作业 (Operation)	预留的作业名称

Simio文档

停止转移(EndTransfer)

停止转移

“停止转移”步骤用来表明和当前执行令牌相关的实体对象已被完全移入了一个对象或站点。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

连接 (Engage)

连接

“连接”步骤把一个实体的位置锁定到其‘父连接对象’的位置。这个方法把实体排列在连接上的方法由连接的实体排列属性 (EntityAlignment) 决定。断开连接(Disengage)步骤是用来解锁实体。

一旦连接, 实体会一直保持连接, 直到用‘断开连接’步骤断开。实体使用连接的实体排列属性设置中的“任何位置”(AnyLocation) 或“单元位置”(CellLocation) 来建立连接。如选择了“单元位置”, 那么实体只能沿着连接的长度在固定的单元位置建立连接。在下一个单元到达之前, 实体要一直等待无法建立连接。如选‘任何位置’, 那么实体能在连接的任何位置建立连接而无需等待。建立连接后, 实体和可移动连接的位置被排在一起而且必须行动一致。当一个连接的实体到达连接末端时, 它会迫使连接停止、在连接的位置不再保持同步。

 注意: 该步骤只能在连接对象的内部使用。

Simio 文档

执行 (Execute)

执行

“执行”步骤用来执行指定的过程。

过程拥有者是父对象或者相关实体。到达的令牌要么立即退出, 要么基于行动属性 (Action) 来等待进程的结束。

下面列出的是 **执行** 操作的属性:

属性	描述
过程名称 (Process Name)	要执行的过程名称。
行动 (Action)	过程在执行时, 要采取的行动。(WaitUntilCompleted 或 Continue)

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

故障(Fail)

故障

“故障”步骤启动了一个故障元素的停顿期。使用Repair步骤结束当机的出现。

下面列出的是**故障**的属性:

属性	描述
----	----

 Forward Thinking		
新一代仿真技术 基于智能对象的仿真	Simio 用户参考指南(过程步骤) (迅合科技 版权所有)	版本 3.53.6762 2011 年 5 月

故障名称 (Failure Name)	即将开始的故障类型。
---------------------	------------

Simio文档

激发(Fire)

激发

“激发”步骤用来激发一个对象事件。可以和Wait步骤一起使用。

关于如何使用激发步骤，请参见小模型 [EntityMovementInFreeSpace](#) 和 [EntityStopsOnLink](#)。

下面列出的是**激发**的属性：

属性	描述
事件名称 (Event Name)	即将激发的事件名。

请将关于该话题的评论发邮件给我们。

2006-2011版权所有, [Simio LLC](#)有限公司保留一切权利。

Simio文档

插入 (Insert)

插入

“插入”步骤用来将对象插入到指定的队列。请参阅存储元素 (Storage) 来定义模型的定制队列。

下面列出的是**插入**的属性：

属性	描述
队列状态名称 (Queue State Name)	要插入对象的那个队列状态的名称。
对象种类 (Object Type)	明确规定了这一步骤是插入父对象还是与执行令牌相关的对象。
排列位置 (Rank Placement)	如果规定了排列位置，这将覆盖队列状态的排名规则。如果队列的排列规则是先进先出或者后进先出，这个表达式被解释成放置插入对象的特定的排序索引。如果队列的排列是最小值优先或者最大值优先，那么这个表达式被解释成插入对象的排列值。

请将关于该话题的评论发邮件给我们。

2006-2011版权所有, [Simio LLC](#)有限公司保留一切权利。

Simio文档

中断 (Interrupt)

中断

中断步骤可以用于打断过程的延迟。在Delay过程步骤中被延迟的令牌是可以被中断的，如果延迟步骤的'Interruptible'属性为True时。为增加处理中断的逻辑（比如重新安排线路的逻辑），中断步骤可以新建和被中断项目相关联的令牌，这些令牌将从该步骤的'Interrupted'退出点离开。

中断步骤可以用于模拟资源先占（Preemption）的情形，允许模型逻辑中断一个过程延迟活动，将该过程使用的资源释放并用于一个更高优先度的活动中。这个方法提供了非常灵活的先占能力，能够精确地控制哪些实体被其他高优先度的活动所先占，以及这些被先占的实体后续如何处理。

在标准对象库中，能够被中断步骤所中断的对象有：服务器，合并器，分离器。在这几个对象中，他们的处理时间延迟属性为“Interruptible”。如果要中断服务器，合并器或者分离器的令牌，只需将中断步骤的过程名属性指定为“对象名.OnEnteredProcessing”。

当令牌进入中断步骤时，它会中断指定的过程，然后令牌会从“Original”退出点离开步骤，然后进行后续处理。当这个原始的令牌结束其步骤（或者开始一个新的延迟类型步骤），那么和被中断的实体相关联的令牌被创建，这个新令牌会从步骤的“Interrupted”退出点离开，并继续后面指定的步骤。

关于如何使用Interrupt步骤，请参考小模型InterruptibleOperator, InterruptingAcrossMultipleServers, 和

[InterruptingServerWithMultipleCapacity](#)。

下面的列表为中断步骤的属性：

属性	描述
Process Name（过程名）	可中断的延迟活动的过程名
Selection Rule（选择规则）	从一个或者多个对象中选取哪个过程延迟被中断的规则
Interrupted Process Action （被中断过程的行动）	表明被中断的过程延迟被如何处理。如果行动为“EndProcess”，那么被打断的过程延迟会立即终止。如果为“ResumeProcess”，那么中断步骤为了处理中断发生创建了一个新令牌，当结束令牌的处理后，被中断的过程延迟会重新恢复，并将剩余的延迟时间继续下去。
Process Names(More)（更多的过程名）	额外的可中断的延迟活动的过程名。
Limit（限度）	中断的过程延迟活动的最大数目。
Save Number Interrupted（保存被中断的个数）	可选的离散状态变量，用来存储被中断的过程延迟活动的数目，这个状态变量指定被运用于执行 Interrupt 步骤的原始令牌。
Save Remaining Time（保存剩余时间）	可选的离散状态变量，用来存储一个被中断的过程延迟的剩余时间（小时）。这个状态变量的指定应用于那些离开 Interrupt 步骤的“Interrupted”退出点的任何令牌。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio文档

移动 (Move)

移动步骤可以用来请求一个或多个‘可移动资源’的移动，这些移动资源已被它们的父对象抓取或者被执行令牌所关联的对象所抓取。在资源到达请求的位置之前，执行令牌将被暂存在Move步骤中。

该步骤能够和Seize步骤/Release步骤配套使用。

下面列出的是 **移动** 的属性值：

属性	描述
Resource Move Requests (资源移动请求)	请求移动的对象。
Move Requests.Object Type (请求移动对象类型)	请求移动的资源对象的指定方法。有 specific (特定对象)， FromList (从链表获取)， ParentObject (父对象)。
Move Requests.Object Name (请求移动对象名)	请求移动的资源对象的名字。
Move Requests.Object List Name (请求移动的对象链表名)	被请求移动的一个或多个资源对象是从哪个链表中获取的，指定其链表名 (Lists)。
Move Requests.Destination Node (移动请求的目的地节点)	被请求移动的资源到达的特定的节点位置。如果未指定，并且请求该移动的上级对象是某个节点的实体对象，那么这个默认的属性值就是上级实体的当前节点。
Move Requests.Number Of Objects (移动请求的对象个数)	被请求移动的资源对象的个数。
Move Requests.Move Priority (移动请求的移动优先级)	移动请求的优先级，如果某个资源从多个成员中选择访问目的地时，使用了 PlanVisit/SelectVisit 步骤，那么这个属性可以指定移动请求的优先级。如果未指定，并且请求该移动的上级对象是一个实体对象，那么这个属性的默认值为那个实体的 Priority 状态值。
Move Requests.Move Request Order (移动请求的请求顺序)	那些被其上级对象获取的资源们的请求移动顺序，有 FirstSeized 和 LastSeized 。
Move Requests.Move Request Condition (请求移动的条件)	一个可选条件，用来评估每个被获取的资源对象，对象被请求移动前，这个条件必须为 True 方可移动。 在条件表达式中，使用关键字 ‘Candidate’ 从对象集合里引用某个对象 (例如 Candidate.Object.ID)。
Owner (所有者)	被请求移动的资源对象的拥有者，即获取该资源的对象。可以指定为：和当前执行令牌关联对象，或父对象。

Simio文档

通知 (Notify)

通知

通知步骤能够用来输出一个用户自定义的跟踪信息或者预警信息。

在Run菜单栏上面设置Warning Level (在Advanced Options下) 可以决定这些警告信息在何处被显示。任何的跟踪信息都会被显示在Trace窗口中。

下面列出的是 **通知** 的属性值:

属性	描述
Notification Type (通知类型)	输出信息的上下文。如果指定为Trace (跟踪信息),并且模型允许跟踪,那么这些信息会在跟踪窗口中被显示。如果指定为Warning (警告),那么这些信息会按照模型的警告设定来处理。
Message Heading (信息标题)	可选的文字信息,用来表示输出信息的标题。可以被指定为字符串表达式,或者一个实际的字符串,用双引号括起。
Message Content(信息内容)	可选的文字信息,用来表示输出信息的内容。可以被指定为字符串表达式,或者一个实际的字符串,用双引号括起。
Notify Condition (通知的条件)	可选的条件,如果被指定的话,那么只有当条件值评价为True的时候,通知信息才会输出给用户。

[请将关于该话题的评论发邮件给我们。](#)

2006-2011 版权所有, Simio LLC有限公司保留一切权利。

Simio文档

停靠 (Park)

停靠

“停靠”步骤用来把实体对象转移进一个节点的停靠站。

下面列出的是**停靠**的属性:

属性	描述
停靠类型	被停放的实体对象,特指与正在执行的令牌相关的对象或者父对象。
节点名称	拥有给实体停放的停放站的节点名称。默认是实体的当前节点或者如果实体不在节点上,则是系统中能够发现的第一个节点。

Simio文档

拾取(Pickup)

拾取

“拾取”步骤是用于运输器对象在运输器当前节点的装载拾取队列中拾取一个实体。

等待的实体按照拾取队列的顺序接受检查。注意实体和运输器要同时‘接受’所选实体的拾取。如果实体/运输器接收了拾取请求，拾取步骤会把实体从节点的拾取队列(RidePickupQueue)中转移出来并使用标准的转移机制转移到运输器的装载站点（由EndTransfer步骤完成转移）。令牌一直被保存在拾取步骤,直到实体转移完成。如果在运输器上的装载站中没有提供移入过程(TransferIn Process)，那么默认认为是零时间的转移。依照拾取尝试的结果,令牌会从两个退出点中的一个退出。如果拾取成功,则它会退出主要退出点。如果拾取尝试失败，它会退出第二个退出点。被运输器拾取的实体会在随后使用Dropoff步骤放下。

注意:如果延迟时间表达式为负数,那么Simio会报错。这个情形在您使用无界的分布函数时发生，无界的分布函数有可能产生负数，例如正态分布，指定Pickup步骤的Load Time。可以参考[分布函数](#)了解更多信息。

下面列出的是拾取的属性:

属性	描述
Selection Goal (选择目标)	从多个候选对象中拾取时，用来排列拾取优先级的目标。
Load Time (加载时间)	装载一个‘乘客’到运输器上的所需延迟。

注意，该步骤只能从运输器对象的内部使用。

[请将关于该话题的评论发邮件给我们。](#)

2006-2011版权所有, Simio LLC有限公司保留一切权利。

Simio文档

计划访问(PlanVisit)

计划访问

“计划访问”步骤是实体对象用来搜索并接受系统中的一个访问请求，随后实体可以使用SelectVisit步骤设定其终点节点为访问位置。该步骤搜索的访问类型包括在Ride步骤引发的装载拾取请求，以及在Seize步骤和Move步骤处引发的移动请求等。

这次访问路径的选择可以建立在指定目标选择的基础上。这个目标可以是优先级或距离的最小/最大值。当选择了一次访问时，在全局访问请求的队列中的实体会按照选择目标确定的顺序接受检查。候选实体只有当实体接受后才会被选择。一旦实体被选中，运输器会执行同样的过程。因此不管是候选实体还是运输器，都必须在接受访问预定前,同意这个访问请求。

下面列出的是 **计划访问** 的属性:

属性	描述
选择目标(Selection Goal)	当有多个候选对象时, 用来排列拾取目的地的优先级目标。
注意: 该步骤只能在实体对象的内部被使用。	

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

生产(Produce)

生产

“生产”步骤生成指定数量的物料——可以是第一层的物料, 或者展开的物料清单。

注意: 如果延迟时间的表达式返回负值的话, Simio 将产生一个错误信息。当你在生产步骤的 **Production Delay Time** 属性中使用了一个无界的分布函数时, 比如整态分布, 可能会产生负数值。查看 [分布函数](#) 部分获得更多信息。

请通过小模型 [ScheduledMaterialArrivals](#) 和 [WorkstationWithMaterialConsumptionAndReplenish](#) 了解如何使用生产步骤。

下面列出的是 **生产** 的属性:

属性	描述
Production Type(生产种类)	指定被生产的物料的方法。有 Material(物料) BillOfMaterial(物料清单) , OperationSpecified(作业指定的) 。
Material Name(物料名称)	在该步骤被生产的特定物料, 或者被生产的物料清单。(取决于 Material 元素的定义)。
Quantity(数量)	要生产的数量。
Production Delay Time(生产的延迟时间)	物料产生之前的延迟时间。执行令牌总是立刻地离开生产步骤, 但是要产生的物料并不会马上出现, 直到该时间延迟发生以后。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

释放 (Release)

释放

“释放”步骤 代表父对象或与执行令牌相关的对象, 用于释放一个或多个对象的能力。

该步骤可以和'捕获'步骤一同使用。

了解释放步骤使用实例，请参阅小模型 [OverflowWIP](#), [ResourcesWithWorkSchedules](#), [SeizingVehicle](#) 以及 [SelectingResourceFromList](#)。

下面列出的是**释放**的属性：

属性	描述
拥有者 (Owner)	拥有被释放的容量单位的对象。特指与执行令牌 相关的对象或父对象。
释放物 (Releases)	要被释放的对象。
释放对象种类 (Releases.Object Type)	指定释放对象的方法。
释放对象名称 (Releases.Object Name)	被释放的对象类型的名称。
释放对象链表名 (Releases.Object List Name)	对象链表的名称，表中一个或多个对象要被释放。
对象释放数量 (Releases.Number Of Objects)	要释放的对象的数量。
每个对象释放单位 (Releases.Units Per Object)	每个对象要被释放的容量单位数量。
释放顺序 (Releases.Release Order)	释放被拥有者所拥有的对象们的顺序。
释放条件 (Releases.Release Condition)	一个可选条件,用来评估每个被拥有的对象,只有当该条件为 True 时,才能释放相应对象,在条件表达式内,使用关键字 'Candidate'来对某个成员集合的对象进行引用.(例如 Candidate.Object.ID)。
释放后的过程 (Releases.On Released Process)	一个可选的过程,在释放发生之后,与释放对象相关的令牌来执行这个可选过程。

Simio文档

移除(Remove)

移除

“移除”步骤用来从一个指定队列中移除对象。“移除”步骤具有两个退出点，' Original' 和 ' Removed' 。与移除实体相关的令牌会退出这一步骤的 ' Removed' 退出点，继续处理直到遇见该令牌或相关实体对应的延迟类型的步骤。然后原始令牌才会从 ' Original' 退出点退出。可以在移除的令牌的延迟步骤中使用 `Math.Epsilon` 数学函数影响事件处理的次序。

Interrupt步骤是不同的,但是有相似性。它可以用于中断一个延迟过程。所以假如您想取消某个实体对于一个资源能力的请求,或者取消实体的转移请求,使用'移除'步骤从队列中移去这个实体,但是假如您想取消正在延迟过程中的一个实体,那么是用**Interrupt**步骤来取消这个延迟。

从一个存储元素队列移除

用户可以在**Simio**里建立他们自己的队列,并将实体放置到自定义队列中(插入步骤),从队列里移除实体对象(移除步骤)。请参阅存储元素(**Storage**)在模型中定义客户定制的队列。

从资源分配队列中移除

从资源的**Allocation Queue**(分配队列)中移去一个实体,是非常强大的功能,能够实现通常所说的'中途退出'功能。它取消了一个**Seize**的意图。如果实体早已请求了任何资源类型的对象的能力的话,那么它在资源的分配队列等待,(即 **Server1.AllocationQueue**),实体可以从该队列被移出,因此它们对于对象能力的请求也会被取消。使用该功能的一个实际例子是,当客户在队列中等待时间过长,决定不在接受服务而离开服务大厅,客户早已请求了服务器的服务能力,并在服务器的分配队列中等待,使用移除步骤可以取消他们的服务请求。

从站点的输入队列移除

从站点的**EntryQueue**(进入队列)或者连接对象的输入队列移除一个实体,通常可以解释为取消转移步骤的转移请求,换句话说,实体不再等待进入该站点。一旦发生这个取消的行动,那么执行该**Transfer**步骤的令牌终止了它的处理,(即被销毁了)。

从连接对象的输入队列中移除

从**Link**对象的输入队列移除一个实体,意味着取消了该实体等待进入该连接的转移请求。

从合并队列中移除

从批次逻辑元素(**BatchLogic**)的父队列(**ParentQueue**)中移除一个实体,允许用户当实体在合并步骤等待批次的形成时,取消该合并步骤的活动。相同地,从批次逻辑元素的成员队列(**MemberQueue**)中移除一个对象,允许用户取消那些等待合并的成员的合并活动。

从线路组元素的线路请求队列中移除

从线路组元素(**RoutingGroup**)的**RouteRequestQueue**中移除一个实体,即取消了**Route**步骤等待将实体的线路安排到一组被封锁的节点目的地的动作。



目前, *Simio*不支持从下列类型的队列中移除对象, 节点的 *RidePickupQueue*, 物料元素的**AllocationQueue**, 网络的**VisitRequestQueue**, 实体的**VisitRequestQueue**, 实体的**BatchQueue**, 站点的 *MemberQueue*

下面列出的是**移除**步骤的属性:

属性	描述
队列状态名 (Queue State Name)	要移除对象的队列状态的名称。
移除类型 (Removal Type)	详细说明这一步骤是要移除父对象、与执行令牌相关的对象或者在指定排名的对象。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

修复 (Repair)

修复

“修复”步骤终止了一个故障元素的停顿期。

下面列出的是 **修复** 的属性:

属性	描述
故障名称 (Failure Name)	要终止的故障元素的名称。

Simio 文档

恢复 (Resume)

恢复

“恢复”步骤用来继续一个进程或者继续一个对象的移动。

如果恢复类型是 'Process', 这一进程的拥有者可以是父对象或与执行令牌相关联的对象。

如果恢复类型是 'ParentObjectMovement' 或 'AssociatedObjectMovement', 则父对象或相关对象的移动会继续。该对象必须是一个连接对象或智能主体对象。

" 恢复 " 步骤可以和 ' 暂停 ' 步骤一起使用。

了解使用 ' 恢复 ' 步骤的实例, 可参阅小模型集 [EntityStopsOnLink](#),

[VehicleFinishesAndParksWhenOffShift](#) 以及 [VehicleStopsWhenServerOffShift](#)。

下面列出的是 **恢复** 的属性:

属性	描述
恢复类型 (Resume Type)	要继续的行为或逻辑的类型。
过程名称 (Process Name)	要恢复的过程名称。

Simio 文档

搭载 (Ride)

搭载

“搭载”步骤用来为一个节点上的实体对象启动运输装载请求。

下面列出的是‘**搭载**’的属性：

属性	描述
运输器类型(TransporterType)	要选择并使用的运输器类型。
运输器名称(Transporter Name)	实体会使用的指定的运输器名称。
运输器链表名 (Transporter List Name)	实体会使用的运输器类型的链表名称。
预留方法(ReservationMethod)	选择并预留运输器给一个实体的方法。
选择目标 (Selection Goal)	有多个候选对象时,用来排列运输器优先级的目标。
选择表达式 (Selection Expression)	标准表达式, 为每个候选的运输器作评估, 以最小值或最大值作为选择目标。在表达式一开始使用关键字 Candidate 。
选择条件(Selection Condition)	评估每个候选运输器的可选条件, 必须为 True 才能够真正使用运输器。在表达式一开始使用关键字 Candidate 。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

路线(Route)

路线

“路线”步骤是和 RoutingGroup 元素一起使用, 为一个实体对象安排到目的地的路线 (目的地从候选节点列表中选择)。

下面列出的是**路线**的属性：

属性	描述
对象类型(Object Type)	指定该步骤安排线路的对象是父对象, 或者是与执行令牌相关的对象。
路线组名称 (Routing Group Name)	要从中选择目的地节点的 Routing Group(路线组)名称。
选择目标 (Selection Goal)	从路线组中选择目的地节点的目标值。SmallestDistance(最短距离),LargestDistance(最长距离),PreferredOrder(最佳顺序),Cyclic(循环的),Random(随机),SmallestValue(最小值),LargestValue(最大值)。

选择表达式 (Selection Expression)	当 Selection Goal 选择为 SmallestValue(最小值),LargestValue(最大值)时, 每个目的地的评价标准是以某个表达值的最小值或最大值作为选择目标。在表达式一开始使用关键字 Candidate。
选择条件 (Selection Condition)	评估每个候选目的地的可选条件,必须为 True 才能够被选择。在表达式一开始使用关键字 Candidate。
开始节点 (Source Node)	可选的节点位置, 被用来计算到此的距离 ('distance-to') 而不是实体对象本身。
阻塞路线规则 (Blocked Routing Rule)	当存在封锁路线的情形时, 用来定义目的地选择的规则。
排除表达式 (Exclusion Expression)	当指定时, 在仿真开始运行的时候会判断某个表达式, 决定是否该步骤排除在运行之外。如果表达式=1, 那么该步骤会被排除运行, 任何的令牌都会直接流向主要退出点, 如果表达式=2, 且该步骤有第二退出点, 那么该步骤被排除, 任何令牌会从第二退出点流出。如果该表达式为其他值, 尤其是 0, 那么步骤会在过程处于活动状态。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

搜索 (Search)

搜索

“搜索”步骤可以用来搜索队列、对象的集合或者表单 (Table)。

返回值表达式可以被指定评估所有上下文的搜索中找到的对象。执行令牌的返回值 (ReturnValue) 返回该值。该值然后被存储在令牌上, 能够稍后在过程中被参考。

这一步骤也被用来创建与搜索发现的对象相关联的新令牌。它允许由某一个对象所执行的过程, 作用于模型中的不同对象上。这些令牌会退出该步骤的'Found'退出点。

搜索步骤有两个退出点, 包括'Original'退出点和'Found'退出点。首先, 被发现的实体的令牌会从'Found'点退出, 除非碰到和该令牌或者关联实体相关的延迟类型的步骤。接着原始令牌会从'Original'退出。函数 Math.Epsilon 可以被用于 Delay 步骤中, 来延迟被发现的对象的令牌, 从而影响事件处理的先后次序。

原始令牌永远是从 Original 分支出去的, 不管搜索步骤是否找到任何对象。所有在 Original 退出部分存在的步骤都会被执行。仅当在搜索步骤中找到某些对象时, 在 Found 分支才会建立新的令牌。因此, 假如搜索步骤找到某对象后, 在 Found 部分存在的所有步骤都会被执行。但是, 如果没有找到, 那么在 Found 部分存在的任何步骤都不会被执行 (因为在这里并没有创建令牌)。

如果需要了解搜索步骤的实例, 请参考小模型 CONWIP, SearchTables, RoutingWithoutPaths, VehicleFinishesAndParksWhenOffShift, 和 VehicleStopsWhenServerOffShift。

更多关于搜索步骤的信息，可以在[搜索-讨论与实例](#)页面找到。

下面列出的是[搜索 \(Search\)](#) 步骤的属性：

属性	描述
集合类型(Collection Type)	要执行的搜索类型，允许的类型值有对象实例(ObjectInstance),对象链表(ObjectList)，被抓取的对象(SeizedObjects)，队列状态(QueueState)，表格行(TableRows)，节点列表(NodeList)。
对象实例名称(Object Instance Name)	要搜索的对象实例名称。
对象链表名称(Object List Name)	要搜索的对象链表的名称。
所有者(Owner)	要搜索的资源对象列表是被哪个对象所获取的。
队列状态名称(Queue State Name)	要搜索的队列状态的名称。公共队列状态值的例子有：Input@Server1.RidePickUpQueue, Server1.Processing.Cotents,Vehicle.VisitRequestQueue。
搜索类型(Search Type)	执行的搜索种类。如果最小化返回值(MinimizeReturnValue)或最大化返回值(MaximizeReturnValue)被选择，它可以，让您在指定的限度内，在搜索的集合里找到那些使评估每个候选对象的返回值表达式最小化或最大化的对象。如果 Forward (向前) 或者 BackWard (向后) 和起始索引一起使用时，可以在对象集合里进行选取。
起始索引(Starting Index)	从 1 开始的搜索的起始索引。默认时，向前搜索时开始索引是集合中的第一个对象，如是向后搜索则是集合中的最后一个对象。
匹配条件(Matching Condition)	可选的匹配条件,用于过滤搜索。只有符合这个匹配标准的对象才会被选择。在表达式的开始使用关键字 Candidate。
限制(Limit)	能找到的对象数量的最大值。
返回值(ReturnValue)	对每一个搜索所得对象进行评估的表达式。被发现的对象的表达式值为原执行令牌的 ReturnValue 状态所返回。请在表达式的开始部分使用关键字 Candidate。
保存发现的索引值(Save Index Found)	可选的离散变量，让您保存在集合中被发现的最后一个对象的索引值。
保存发现的数量(Save Number Found)	可选的离散变量，让您保存搜索找到的对象数目。

Simio文档

搜索-讨论与实例

案例

使用搜索步骤来保存被抓取对象的 ID 标识

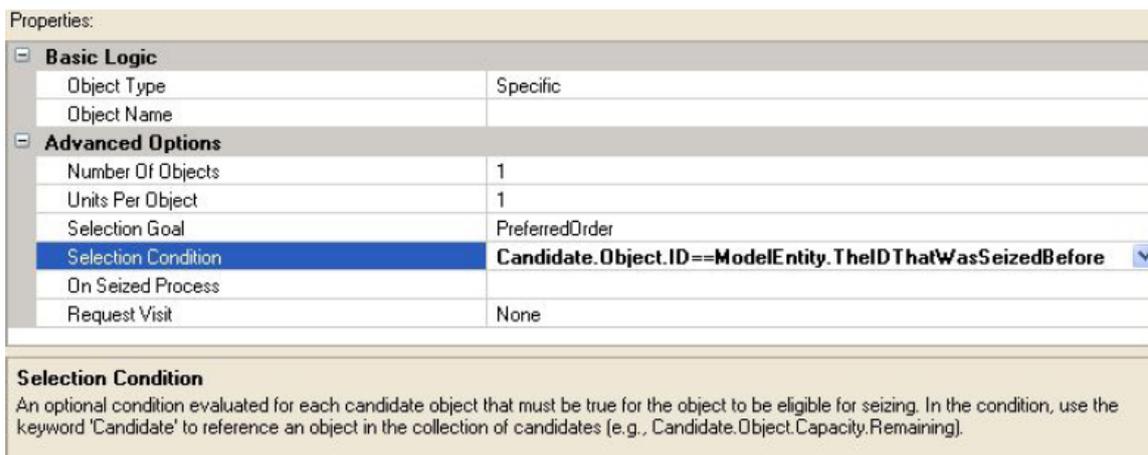
- 在这个案例中，使用者应牢记列表上的哪个资源被捕获了。因此，当实体回到这个步骤时，就能捕获和第一次访问该过程时所获得的同一个资源。
在实体捕获了对象之后，使用者应该保存被捕获对象的标识。使用搜索步骤来采集那些被实体捕获的对象，并得到最后所获取的对象的标识。



Search Step Properties Window

搜索步骤属性窗口

在令牌离开搜索步骤时，把令牌的返回值 `Token.ReturnValue`（它含有标识号）指定给实体状态值，实体便能够随身携带这个信息。当您使用另一个 `Seize` 步骤时，您可以在捕获步骤中使用选择条件(Selection Condition)来捕获那个您想再一次捕获的同一资源。例如，您保存了被捕获的资源的标识号，并赋值给了一个名为“`TheIDThatWasSeizedBefore`”的状态值，`Seize` 步骤中的属性如下图所示。



捕获步骤属性窗口

使用搜索步骤查询表格中的某个数值

在这个例子中, 用户希望能够在 Simio 的数据表(Table)中搜索和某个条件所匹配的某行。假定一个实体具有离散状态值, 名为 MyCustomState, 假设某个数据表中包含一个属性叫做 CustomStateID。每个实体的 MyCustomState 值是 10, 20, 30 或 40。数据表的 CustomStateID 列有四行, 分别是 10, 20, 30, 40。该数据表还包含一列, 它是节点属性的, 包含某个特定实体所要去的节点。使用搜索步骤, 一个和实体相关联的令牌会找到 ModelEntity.MyCustomState 等于 Table1.CustomStateID 的那一行, 在搜索步骤的 Found 分支, 用一个 SetNode 步骤来设定实体的目的地节点为 DataTable.NodeProperty。

使用搜索步骤来搜索表格, 能够在仿真运行开始, 对对象的 WIP 做初始化。

Simio文档

捕获(Seize)

捕获

‘捕获步骤’代表父对象或和执行令牌相关的对象, 用于捕获一个或多个对象的能力。

每个对象代表着具备能力的资源。对象资源必须被整体捕获或释放, 并能够被父对象或那些和执行令牌相关的对象所拥有。这些将被拥有者所捕获的对象可能来自于一个特定的对象, 或是从一个对象链表中被选中的。如果是特定的对象, 那么它可能是一个静态的对象, 或者从许多复制对象中选取的对象样本。如果是从对象列表中选取的, 或者所需对象的数量比可以得到的少的话, ‘资源选择规则’将被用于选择可得的对象。对象列表既包括复制的对象(即动态对象成员)也包括非复制对象(静态运行空间对象, 固定对象)。如果是对象群体的话, 那么所有的成员都适合被选取, 对象的容量会在稍后被释放, 或者通过特定的名称, 或者通过对象列表名。若是在对象列表名中被释放的, 那些由特定列表所拥有的对象就能够正常地被释放。他们释放的规则是根据次序来释放的, 可以指定为FirstSeized(第一次捕获)或最后捕获(LastSeized)原则。

资源可以被分配到父对象或是关联的对象上。如果被父对象捕获, 那么它必须被父对象释放。如果资源是被关联对象捕获的, 那么它可以被关联的对象在多个对象之间携带。

捕获步骤必须和Release步骤一起使用。

了解更多关于捕获步骤的信息, 参考小模型集[OverflowWIP](#), [ResourcesWithWorkSchedules](#), [SeizingVehicle](#)和[SelectingResourceFromList](#)。

下面列出了 **捕获** 的属性:

属性	描述
捕获 (Seizes)	将被捕获的对象。
捕获的对象类型 (Seizes.Object Type)	指定对象被捕获的方法。
捕获的对象名 (Seizes.Object Name)	将被捕获的对象类型的名称。
捕获的对象列表名 (Seizes.Object List Name)	将被选择捕获的一个或多个对象的对象列表名称。
捕获对象请求移动 (Seizes.Request Move)	表明被捕获的对象是否需要请求移动到某个指定地点。执行令牌将被保留在Seize步骤中直到被捕获对象到达请求的地点。
捕获对象的目的地节点 (Seizes.Destination Node)	被捕获对象被请求移动到的特定目的地节点。如果未

	指定，且实施捕获动作的拥有者是在某个节点的实体对象，那么该属性默认值为拥有者实体的当前节点。
捕获对象移动优先级 (Seizes.Move Priority)	如果被捕获对象使用PlanVisit/SelectVisit步骤从多个目的地成员中选择访问目的地的话，该属性指定移动的优先度。如果未指定，且实施捕获动作的拥有者对象是一个实体对象，那么该属性的默认值是该实体的优先级状态值。
捕获对象数量 (Seizes.Number Of Objects)	将被捕获的对象数量。
捕获每个对象的单位 (Seizes.Units Per Object)	每个对象被捕获的能力单位 (依据每个对象的容量类型，可得的能力单位)。
捕获选择目标 (Seizes.Selection Goal)	选择捕获对象的目标
捕获选择表达式 (Seizes.Selection Expression)	用来评价每个候选对象的表达式，在表达式前面使用关键词Candidate，用来从要被捕获的候选对象集中引用某个对象。(比如 Candidate.Object.Capacity.Remaining)
捕获的选择条件 (Seizes.Selection Condition)	一个可选的条件,对每个候选对象进行评估,只有为True时,对象才有资格被捕获。在条件前面使用关键字Candidate来引用候选对象集中的某个对象。(比如,Candidate.Object.Capacity.Remaining)
捕获的过程 (Seizes.On Seized Process)	在捕获发生之后,被捕获对象所关联的令牌所执行的一个附加过程。
拥有者 (Owner)	这个对象将分配捕获的能力。特指和执行令牌相关联的对象或父对象。
捕获作业资源 (Seize Operation Resources)	表明是否同时捕获活动资源(为拥有者实体的当前作业所指定的资源)。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

选择放下(SelectDropoff)

选择放下

选择放下步骤将正在运载的实体的放下位置，设置为父运输器对象的目的地节点。

如果目的地被选中，令牌在主要退出点退出，否则它将从Failed退出点退出。

下表是 **选择放下** 的属性：

属性	描述
选择目标 (Selection Goal)	在众多候选对象中，用来给放下的目的地优先级进行排序的目标。有 SmallestDistance(最短距离),LargestDistance(最长距离),SmallestPriority(最小优先级),LargestPriority(最大优先级),FirstInQueue(队列中的第一个) 注意：该步骤只能在运输器对象内部使用。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

选择访问 (SelectVisit)

选择访问

该步骤被实体对象所使用,用于设定其目的地节点到访问请求的地点(由搭载或者移动步骤发起)。

为了可以被本步骤选取,访问请求首先必须被实体所接受(使用PlanVisit)。如果目的地节点被选择,那么令牌从主要退出点退出,否则从故障退出点退出。

下表是**选择访问**的属性:

属性	描述
选择目标 (Selection Goal)	当从众多的候选点中选择访问时,用来对访问优先级排序的目标

 注意：该步骤只能在实体对象的内部被使用。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

设置网络(SetNetwork)

设置网络

'设置网络'步骤可以被用于建立实体所使用的在节点位置之间行驶的连接网络。与NetWork元素一起使用。

下表是**设置网络**的属性:

属性	描述
对象类型(Object Type)	指明该步骤是否设置父对象的网络值,或是设置和执行令牌相关联对象的网络值。
新网络名(New Network Name)	将被设置的新网络的名字。

Simio 文档

设置节点(SetNode)

设置节点

设置节点可以被用作设置任何实体对象的‘目的地’节点。

关于使用设置节点步骤的实例，参考小模型[SearchTables](#)。

下表是 [设置节点](#) 的属性：

属性	描述
对象类型(Object Type)	表明该步骤是否是 设置父对象的终点节点的值，还是设置和执行令牌相关联对象的终点节点的值。
目的地类型 (Destination Type)	该方法用来选择目的地节点。“By Sequence”类型要求实体对象已被指定了序列列表。目的地节点值将被设置为序列列表中的下一个节点值。
节点名字(Node Name)	一个特定节点对象的名字，用于设置目的地的值。
表达式(Expression)	评价整型的节点对象标识的表达式。即目的地节点的ID值。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

设置表格(SetTable)

设置表格

‘设置表格’步骤给一个令牌或对象指定数据表或序列列表。表格的当前行索引值由RowIndex属性所指定。令牌或对象一次可以和多个表格的数据绑定。要实现该功能，只需要用两个SetTable步骤分别引用感兴趣的表格。

了解如何使用设置表格步骤，请参考小模型：

[EntityFollowsSequenceWithTable](#), [EntityFollowsSequenceWithTable2](#), [LeveledArrivals](#), [SelectEntityTypeFromTable](#)和[UsingRelationalTables](#)。

下表为‘设置表格’步骤的属性：

属性	描述
对象类型(Object Type)	说明该步骤指定表格是给父对象，还是给执行令牌，或是执行令牌相关联对象。
表格名称(Table Name)	序列列表或数据表的名称
列数量(Row Number)	表格的行索引值，是从 1 开始的。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

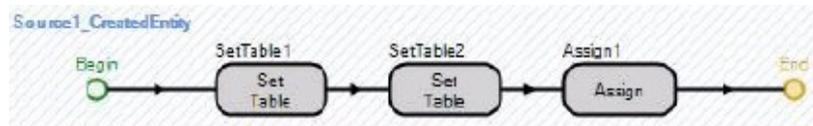
设置表格步骤 – 讨论和实例

例子

□ 设置两个不同的表格到令牌

- 在这个例子中，用户希望设定一个令牌指向两个不同的表格，然后在一个Assign步骤中引用两个表格中的属性值。第一个SetTable步骤设定令牌指向Table1的第一行，第二个SetTable步骤则设定

同一个令牌指向Table2的第二行，Assign步骤从两个表格里引用属性值，赋值到其New Value表达式中。



Properties: SetTable1 (SetTable Step Instance)

Basic Logic	
Table Name	Table1
Row Number	1
Advanced Options	
Object Type	Token
Exclusion Expression	
General	

Properties: SetTable2 (SetTable Step Instance)

Basic Logic	
Table Name	Table2
Row Number	2
Advanced Options	
Object Type	Token
Exclusion Expression	
General	

Properties: Assign1 (Assign Step Instance)

Basic Logic	
State Variable Name	Total
New Value	Table1.Length_1 + Table2.Length_2
Assignments (More)	0 Rows
Advanced Options	
General	

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

开始活动(StartActivity)

开始活动

'开始活动' 步骤开始被预留作业中的下一个活动。和作业 (Operation) 元素一起使用。

下表是开始活动的属性:

属性	描述
作业 (Operation)	要预留的作业名称。

Simio 文档

开始作业(StartOperation)

开始作业

‘开始作业’意为开始一项作业活动，和作业元素一同使用。

一项作业必须在容量有空余和有空闲时间的时候才能开始。要求的时间空隙是以父资源对象的工作时间表为基础的最佳时间—这个时间段通过缓冲时间得到扩展，缓冲时间用来表示次要资源（Secondary Resource），物料，运行等引起的拖延时间。如果容量有空余，最佳的事务时间空隙也有空余，那么，它将捕获父资源并开始作业。如果实际操作时间超过了空余的时间空隙，那么未来的作业时间将会被挤占。

下表是‘开始作业’的属性：

属性	描述
作业（Operation）	需要开始的作业名称。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

预设事件(Subscribe)

预设事件

预设事件用于给过程增加一个新的触发事件，表明一旦事件发生，过程将被执行。

使用预设事件的例子是，一个服务器在开始加工前捕获了某种资源，这种资源在使用过程中的状态可能是 On Shift(当班)或者 Off Shift(下班)。如果您希望服务器在资源停止的时候也停止加工过程，预设动作将触发一个过程，当在资源容量改变（Resource.CapacityChanged）这个事件发生时，暂停服务器，此时容量恰好为0。在这个例子中，暂停服务器的过程将使用暂停步骤（Suspend）来中止服务器中进入加工的事件（OnEnteredProcessing）。由于这种过程不是标准库服务器对象的公共过程，使用者需要建立一个服务器对象的子类，重写进入的过程（OnEnteredProcess）并将公共属性改为‘True’。改变后，过程将在暂停步骤中出现。

同时，工作站对象当任何附属资源停工或者开工时，利用这些步骤，来暂停或恢复当前工作的处理。

关于使用预设事件步骤的小例子，请参考模型 [EntityStopsOnLink](#)。

下面是 预设事件 的属性：

属性	描述
事件名称（Event Name）	为一个过程新增加的触发事件的名称。
过程名称（Process Name）	新触发的事件使用的过程名。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, Simio LLC 有限公司保留一切权利。

Simio 文档

暂停（Suspend）

暂停

‘暂停步骤’被用来中止一个过程或对象的移动。

如果说中止类型（SuspendType）是“过程”，则过程的拥有者就是父对象或与执行令牌相关联的对象。在过程中的所有令牌在当前的状态下是被冻结的。如果一个过程在还未开始的时候就被中止了，则当它“开始”的时候它将立刻中止。如果中止类型是“ParentObjectMovement”或“AssociatedObjectMovement”，父对象或相关对象的移动将被中止。此时，对象必须是连接对象或智能主体对象（Agent）。

‘暂停步骤’可以和‘恢复’步骤一起使用。

了解使用暂停步骤的实例，请参考小模型集 [EntityStopsOnLink](#), [VehicleFinishesAndParksWhenOffShift](#) 和

[VehicleStopsWhenServerOffShift](#)。

下表是 **暂停** 的属性：

属性	描述
暂停类型（Suspend Type）	中止行为或逻辑的类型
过程名称（Process Name）	中止的过程的名称

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

计数 (Tally)

计数

计数步骤是当每个令牌到达这个步骤时对，对其观察值进行记录。

被记录下来的值可以是表达式，可以记录为到达 Tally 步骤的时间间隔。这些值被一个记数统计元素（Tally Statistics）所记录。

下表是 **计数** 的属性：

属性	描述
值类型(Value Type)	将被记录的观察值的类型。值类型“Expression”记录了指定表达式的值。值类型“TimeBetween”记录了到达记数步骤的时间间隔。
计数统计元素名(Tally Statistic Name)	观察值被记录至某个 Tally Statistic 元素。
值(Value)	将被记录的表达式值。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

转移(Transfer)

转移

转移步骤可以将与执行令牌相关的实体对象在对象之间和不同的位置类型之间移动，比如在自由空间和对象之间。

在把一个实体转移到某节点时，而该实体还没有一个目的地的时，转移步骤自动地将该节点设置为实体

的目的地。

用户可以使用**Transfer**步骤将实体在对象位置之间移动，而移动的实体不需要使用对象的外部节点进入或离开对象。所以，比如，你可以使用转移步骤，立即将实体从对象 A 的内部移动到 B 对象外部的节点。

 在目前时点，只有将实体通过对象的外部节点转移到或者移出某个对象，才是最**安全**的，因为绝大多数的对象是假设实体是通过其节点进入或者退出的。比如，外部逻辑也许是将实体迅速地从服务器的内部转移到外部的其他位置，然而，如果这样的话，在服务器内部运行着的令牌并不知道实体已经转移出去了，它还是假定实体仍旧在服务器内部。令牌会继续地尝试处理该实体，从而产生错误。

了解更多关于转移步骤的实例，请参考小模型集 [CONWIP](#), [DynamicallyCreatingVehicles](#), 和 [EntityMovementInFreeSpace](#)。

下表是 **转移步骤** 的属性：

属性	描述
出发地 (From)	实体对象进行转移的出发地类型。
目的地 (To)	实体对象将被转移到的目的地类型。
节点名称(Node Name)	实体将被转移到的节点。
外部节点名称 (External Node Name)	实体将被转移到其父对象的外部节点。
站点名称 (Station Name)	实体即将被转移进入的站点的名称。
对象类型 (Object Type)	将被转移的实体对象，指定为该步骤中，和执行令牌相关的对象，或父对象。
堵塞的行为(Blocked Action)	如果在这个步骤，实体由于被堵塞无法立刻进行转移而采取的行为（等待或放弃）。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio 文档

拆分处理(Unbatch)

拆分处理

该步骤可以用于从执行令牌相关的父实体对象中去除合并的成员 (Batch Members)。

这一步骤将从拥有者实体中，从上一层的合并对象里，移走指定数量的成员实体。批次是在使用合并步骤时产生的。

下表是**拆分处理**的属性：

属性	描述
数量 (Quantity)	从父实体中拆分的成员实体的数量。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio文档

结束停靠(UnPark)

结束停靠

该步骤用于将一个实体对象从节点的停靠地点（ParkingStation）移走。

下表是 **结束停靠** 的属性：

属性	描述
结束停靠类型(UnPark Type)	被移走的实体对象。特指和执行令牌相关的对象或父对象。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio文档

取消预设事件(UnSubscribe)

取消预设事件

该步骤用于取消某个过程的预设的触发事件。和预设事件步骤配套使用。查看 [预设事件步骤](#) 了解使用。

下表是 ‘**取消预设事件**’ 的属性：

预设动作	描述
事件名称 (Event Name)	被取消作为某个过程的触发事件的名称。
过程名称 (Process Name)	被取消该触发事件应用的过程名称。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio文档

访问节点(VisitNode)

访问节点

访问节点步骤在节点对象的逻辑内部使用, 用来开始与执行令牌相关实体的访问节点过程 (OnVisitingNode 过程)。令牌等待直到实体的访问节点过程已经完成, 然后令牌才退出该步骤。

 此步骤只能在节点对象的内部使用。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。

Simio文档

等待(Wait)

等待

该步骤将到达的令牌保存在步骤中, 直到某个指定的事件发生, 令牌才继续流动。

令牌根据他们到达的先后顺序在等待步骤中被释放 (即先进先出)。多个令牌同时在等待步骤中等待的情

况下，在指定事件被激发时，所有令牌都被释放。

下表是 **等待** 的属性：

属性	描述
事件名称 (Event Name)	到达的令牌存放于等待步骤中，直到该事件发生。
释放条件 (Release Condition)	除非这个可选的条件为 " 真 " ，否则即使在指定事件发生时，令牌也不会被释放。

请将关于该话题的评论发邮件给我们。

2006-2011 版权所有, **Simio LLC** 有限公司保留一切权利。