

# RUCM

# 需求建模工具

北京航空航天大学安全关键系统与软件研究中心

2014-10-10

# 目 录

|                    |    |
|--------------------|----|
| 1. RUCM 建模方法 ..... | 3  |
| 2. 建模工具使用说明介绍..... | 4  |
| 3. 案例运行.....       | 7  |
| 附录 A.....          | 10 |

## 文档说明

本文档简要说明 RUCM 建模方法及其相关转换，其目的是为了快速掌握 RUCM 方法及其建模工具的使用，RUCM<sup>1</sup>方法及其相关研究论文可参考<sup>2,3</sup>。

## 文档简要介绍

第一章简要介绍研究方法，用于快速掌握 RUCM 需求建模方法；  
第二章介绍其配套工具的使用，用于指导工具使用，便于快速上手；  
第三章给出一个真实案例，用于加强方法与工具的了解与整体流程的熟悉。

---

<sup>1</sup> Yue, T., Briand, L. C. and Labiche, Y.: Facilitating the Transition from Use Case Models to Analysis Models: Approach and Experiments. Transactions on Software Engineering and Methodology (TOSEM), vol. 22(2011)

<sup>2</sup> Yue, T., Briand L., and Labiche, Y.: Automatically Deriving a UML Analysis Model from a Use Case Model. Simula Research Laboratory, Technical Report (2013)).

<sup>3</sup> Zhang, G., Yue, T., Wu, J., and Ali, S.: Zen-RUCM: A Tool for Supporting a Comprehensive and Extensible Use Case Modeling Framework. in ACM/IEEE 16th International Conference on Model Driven Engineering Languages and Systems (MODELS), Tools Demonstration(2013)

如何在需求阶段全面地、准确地描述软件需求是一个关键问题。需求描述了软件系统的行为、特性或属性，是用户和开发人员之间的桥梁。准确、完整的需求是指导系统后续建模、分析、开发和测试的根本依据。

## 1. RUCM 建模方法

RUCM (Restricted Use Case Modeling) 建模方法是一种基于用例建模 (Use Case Modeling) 的需求建模方法，其目标是降低软件需求模型的二义性，并支持分析模型的自动生成技术。该研究根据目前用例建模领域的研究成果与工业领域的实践经验，提出了一种改良的用例规约 (Use Case Specification) 模板，并提供了 26 条限制规则 (Restriction Rule) 以约束用例规约中自然语言的使用。依靠 RUCM 的用例模板和限制规则，可以有效降低用例规约的二义性，同时又保留自然语言易理解和易使用的优点。已通过受控实验表明，RUCM 方法是容易被理解和使用的，并且有效地降低了需求模型的二义性

RUCM 用例模板的整体结构与常见的用例模板类似，拥有用例名、用例简述、前置条件、后置条件、活动者、依赖关系和泛化关系等字段。模板包含一个基本事件流 (Basic Flow) 和任意多个分支事件流 (Alternative Flow)。模板的内容与主要作用见表 1。

表 1 结构化需求规格模板

| Use Case Specification                    |                          |                            |
|---|--------------------------|----------------------------|
| Use Case Name                             | 用例名称，通常以动词开头             |                            |
| Brief Description                         | 用例内容的简要描述                |                            |
| Precondition                              | 用例的前置条件，即用例执行前必须满足的条件    |                            |
| Primary Actor                             | 主要活动者，即触发用例执行的活动者        |                            |
| Secondary Actors                          | 次要活动者，除首要活动者外系统所依赖的其它活动者 |                            |
| Dependency                                | 依赖关系，描述与其它用例之间的包含和扩展关系   |                            |
| Generalization                            | 泛化关系，描述与其它用例之间的泛化关系      |                            |
| Basic Flow                                | 基本事件流，描述完全顺利执行用例时的事件流程   |                            |
|   | Steps(numbered)          | 事件流步骤                      |
|   | PostCondition            | 基本流的后置条件                   |
| Specific Alternative Flows                | 特定分支流                    |                            |
|   | RFS                      | 引用基本流程步骤的名字与步骤号，例如 RFS 3   |
|   | Steps(numbered)          | 事件流步骤                      |
|   | PostCondition            | 分支流后置条件                    |
| Global Alternative Flows                  | 全局分支流                    |                            |
|   | Guard Condition          | 进入该全局分支测试流程执行的守护条件         |
|   | Steps(numbered)          | 事件流步骤                      |
|   | PostCondition            | 分支流后置条件                    |
| Bounded Alternative Flows (Test Sequence) | 有界分支流                    |                            |
|   | RFS                      | 引用步骤的所在事件流的名字与步骤号，例如，RFS 3 |
|   | Steps(numbered)          | 事件流步骤                      |
|   | PostCondition            | 分支流后置条件                    |

事件流中的每个动作按顺序执行，相互不重叠。每个动作的描述必须是以下五种情况中的一种：（1）主要活动者向系统发送某个请求或数据；（2）系统对某个请求或数据进行验证；（3）系统自身的状态或属性发生的改变；（4）系统向主要活动者反馈结果；（5）系统向次要活动者发送请求。特别指出的是，活动者与活动者之间的交互是不需要在事件流中描述的，因为它与系统无关。

基本事件流描述了用例成功执行情况下的步骤。分支事件流描述了异常发生时的分支情况。分支流的进入依靠在基本流或其它分流中根据限制规则定义的条件决定。分支流分为三种，包括特定分支流 (Specific Alternative Flow)、有界分支流 (Bounded Alternative Flow) 和全局分支流 (Global Alternative Flow)。特定分支流是指从其它流中某个特定步骤产生的分

支流；有界分支流是指从多个步骤产生的分支流；全局分支流是指从基本流中所有步骤均可产生分支的分支流。RUCM 的每个分支流都包含一个独立的后置条件。

RUCM 的用例规约也使用自然语言来描述，但是 RUCM 方法通过 26 条限制规则对自然语言的使用进行了约束，见表 2。这 26 条限制规则可分为两类，一类用来限定自然语言的表述方式（R1-R16 具体见附录 A），另一类强制要求使用关键词来描述特定动作或流程（R17-R26）。定义限制规则的主要目的是降低自然语言的二义性，同时又保留了自然语言易使用和易理解的优点。

总的来看，RUCM 方法沿用了用例建模的基本概念，并提出了一个新的用例模板，它与传统用例模板的区别主要在于以下几点：

- （1）将活动者分为首要活动者（Primary Actor）和次要活动者（Secondary Actor）。首要活动者是触发当前用例的活动者，其他活动者为次要活动者。
- （2）对分支流进行分类，包括特定分支流、有界分支流和全局分支流。
- （3）每个分支流通过关键字 RFS（Reference Flow Step）明确描述分支流是从什么位置产生的。
- （4）每个分支流拥有一个独立的后置条件（Postcondition），而传统的模板往往仅有一个后置条件。
- （5）通过定义限制规则，有效降低了用例规约的二义性。

表 2 RUCM26 条限制规则

| 编号  | 描述                       | 编号  | 描述                   |
|-----|--------------------------|-----|----------------------|
| R1  | 事件流中句子的主语必须是系统或是活动者。     | R14 | 不要使用代词。              |
| R2  | 按次序描述流中的事件。              | R15 | 不要使用插入语。             |
| R3  | 不允许出现活动者和活动者之间的交互。       | R16 | 只用“系统”一词表示系统。        |
| R4  | 每句话描述一个动作（避免出现复合句）。      | R17 | INCLUDE USE CASE     |
| R5  | 只使用现在时。                  | R18 | EXTENDED BY USE CASE |
| R6  | 使用主动语态，避免被动语态。           | R19 | RFS                  |
| R7  | 清楚地表述活动者和系统之间的交互，避免出现省略。 | R20 | IF-THEN-ELSE-ELSEIF  |
| R8  | 只使用陈述句。                  | R21 | MEANWHILE            |
| R9  | 用词保持一致性。                 | R22 | VALIDATE THAT        |
| R10 | 不适用情态动词（如“可能”）。          | R23 | DO-UNTIL             |
| R11 | 避免使用副词。                  | R24 | ABORT                |
| R12 | 只使用简单句。                  | R25 | RESUME STEP          |
| R13 | 不使用否定意思的形容词。             | R26 | 每个分支流都有自己的后置条件。      |

2. 建模工具使用说明介绍

RUCM 根据用例模板提供用户编辑用例规格说明的功能，具体步骤如下：

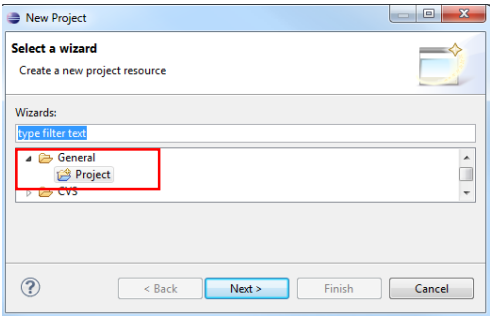


图 1 创建工程截图

- （1） 创建工程（container）

在 Eclipse 中选择基本的工程（如图 1 所示）即可，作为需求模型的容器。

(2) 新建需求模型文件

在 RUCM 目录下，新建需求模型（如图 2 所示），并选择容器与命名需求模型。

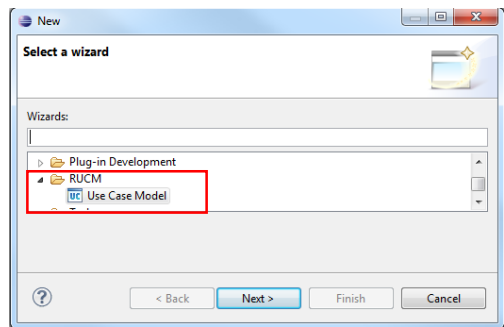


图 2 新建需求模型截图

需求建模编辑视图如图 3 所示，包括三个区域树形模型编辑器区域，需求模型概述编辑器区域与需求模型属性信息显示区域。

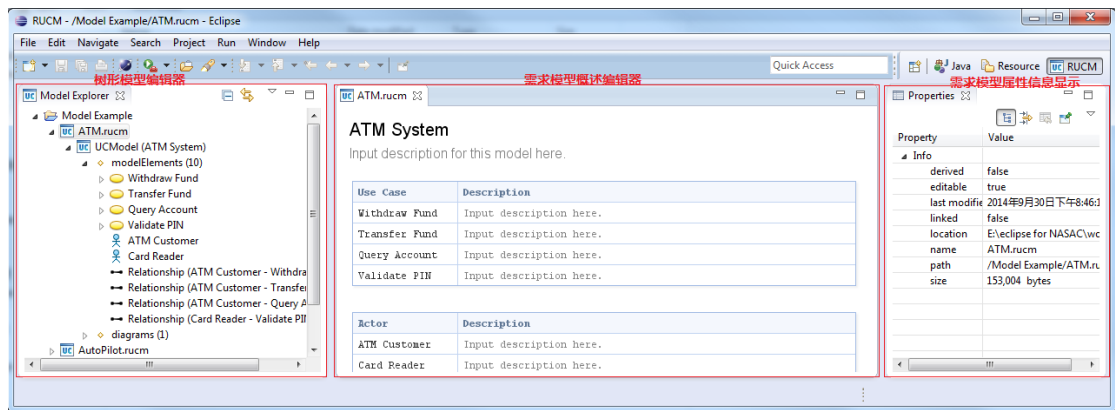


图 3 需求建模编辑视图截图

(3) 新建需求建模元素

如图 4 所示，选择 modelElements 右击，显示上下文工具栏，在 Add Element 下层目录选择关联关系/用例/参与者/包等新建需求元素。

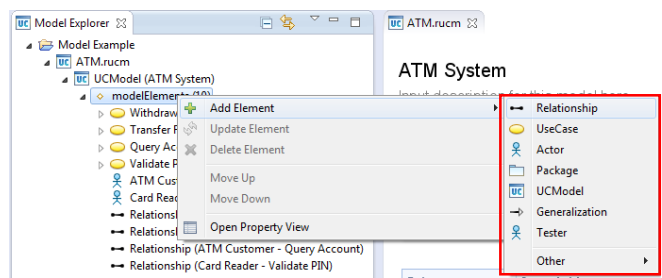


图 4 新建测试元素截图

(4) 新建需求规格说明

如图 5 所示，双击用例完成新建/打开用例规格说明。



图 5 新建需求规格截图

### (5) 编辑需求规格说明

如图 5 所示，双击右侧模型编辑器中的用例\需求规格说明启动需求规格编辑界面，如图 6 所示，编辑界面的要素与需求模板对应，包括 7 个基本编辑区（用例名称，简要概述，前置条件，主要活动者，次要活动者，依赖关系与泛化关系）以及 4 种事件流的操作，根据需求模板，每一个需求规格有且只有一个基本流，因此初始化的需求规格已存在一个基本流，其他分支流的操作可以通过点击事件流图标来完成添加/删除的操作。

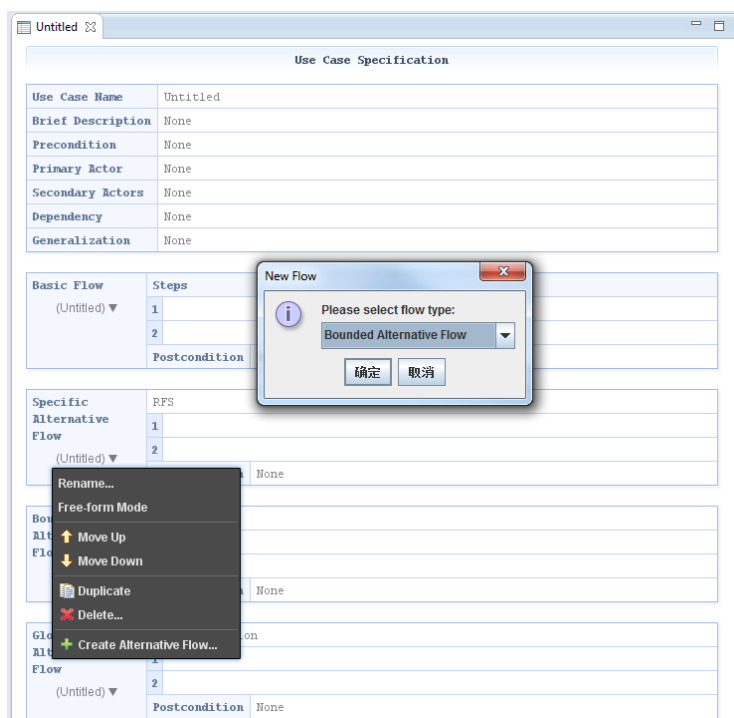


图 6 需求规格编辑界面截图

### (6) 模型解析功能

在 RUCM 建模工具中，通过编辑可以实现模型解析工作，如图 7 所示，左侧的模型实例与右侧的规格编辑器区域的模型元素一致；同时根据文本编辑实现语句的初步解析，如所示基本流第五句，编辑 *The system **VALIDATE THAT** the account number is valid.*，本工具会自动解析出语句特征并创建 ValidateThatNature 实例，并解析其处于为 The system，验证条件为 the account number is valid。

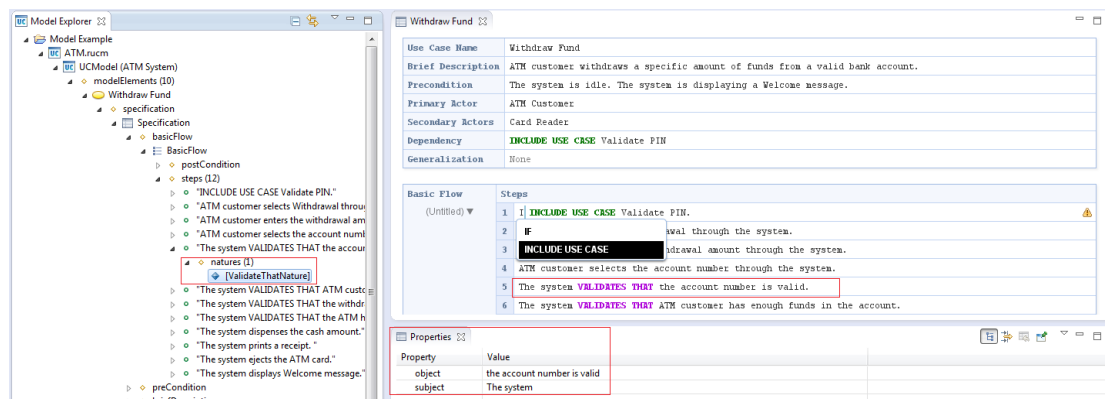


图 7 工具的模型解析功能器截图

### (5) 其他功能

为了方便用户输入与模型的基本验证，本工具提供关键字提示与高亮功能，同时依据限制规则语法，提供错误提示功能（如图 7）。

## 3. 案例运行

本部分选取 ATM 案例中的取款用例（Withdraw Fund），示例 RUCM 建模方法与其工具。

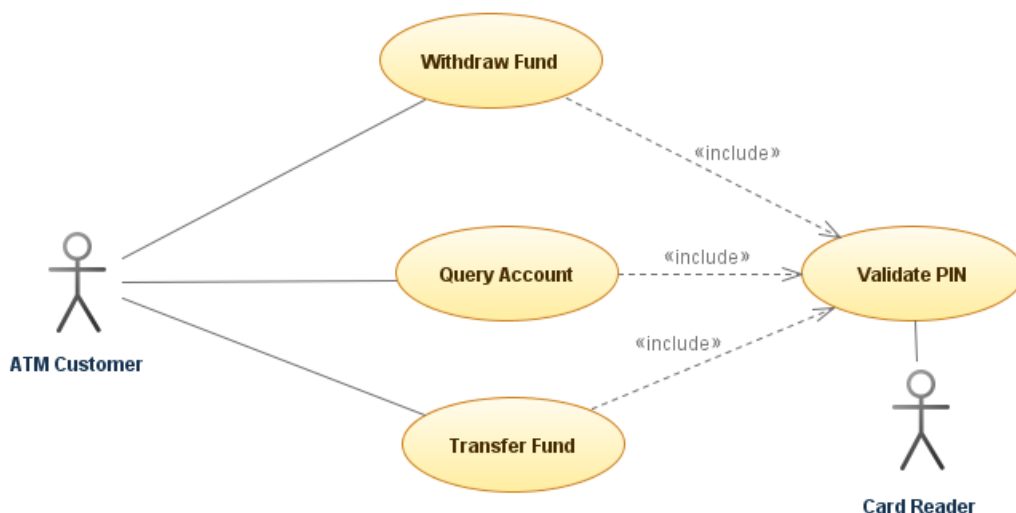


图 8 ATM 案例整体用例图截图

如图 8 所示，ATM 案例包括 2 个参与者（ATM customer 和 CardReader）与 3 个用例（Withdraw Fund，Query Account，Transfer Fund 和 Validate PIN）。通过使用 RUCM 需求建模工具可以对用例进行更为详细明确的描述，如图 9 所示的取款案例：

- (1) 前置条件：ATM 系统闲置状态，系统呈现欢迎界面
- (2) 首要参与者：ATM 客户
- (3) 次要参与者：读卡装置
- (4) 依赖关系：验证 PIN 码用例
- (5) 泛化关系：无
- (6) 基本流：
  - 1) 引用验证 PIN 码用例
  - 2) ATM 用户选择取款业务
  - 3) ATM 用户输入取款数目

- 4) ATM 用户选择取款账号
- 5) 系统验证账号是否有效(如果验证结果为真,进行下一步,否则进入分支流 alt2)

...

- 8) 系统验证 ATM 是否有足够的取款数额(如果验证为真,进行下一步,否则进入分支流 alt1)

后置条件: **ATM 用户已取款。**

- (7) 分支流 alt1:

- 1) 系统提示抱歉信息,同时系统退卡
- 2) 系统服务关闭

| Use Case Specification |  |
|------------------------|--|
| Use Case Name          | Withdraw Fund  |
| Brief Description      | ATM customer withdraws a specific amount of funds from a valid bank account. |
| Precondition           | The system is idle. The system is displaying a Welcome message.              |
| Primary Actor          | ATM Customer   |
| Secondary Actors       | Card Reader  |
| Dependency             | <b>INCLUDE USE CASE</b> Validate PIN   |
| Generalization         | None   |

|                            |  |
|----------------------------|--|
| Basic Flow<br>(Untitled) ▼ | Steps<br>1 <b>INCLUDE USE CASE</b> Validate PIN.<br>2 ATM customer selects Withdrawal through the system.<br>3 ATM customer enters the withdrawal amount through the system.<br>4 ATM customer selects the account number through the system.<br>5 The system <b>VALIDATES THAT</b> the account number is valid.<br>6 The system <b>VALIDATES THAT</b> ATM customer has enough funds in the account.<br>7 The system <b>VALIDATES THAT</b> the withdrawal amount does not exceed the daily limit of the account.<br>8 The system <b>VALIDATES THAT</b> the ATM has enough funds.<br>9 The system dispenses the cash amount.<br>10 The system prints a receipt.<br>11 The system ejects the ATM card.<br>12 The system displays Welcome message.<br>Postcondition ATM customer funds have been withdrawn. |
|----------------------------|--|

|                                       |  |
|---------------------------------------|--|
| Specific Alternative Flow<br>"alt1" ▼ | <b>REF 8</b><br>1 The system displays an apology message <b>MEANWHILE</b> the system ejects the ATM card.<br>2 The system shuts down.<br>3 <b>ABORT.</b><br>Postcondition ATM customer funds have not been withdrawn. The system is shut down. |
|---------------------------------------|--|

|                                      |   |
|--------------------------------------|---|
| Bounded Alternative Flow<br>"alt2" ▼ | <b>REF 5-7</b><br>1 The system displays an apology message <b>MEANWHILE</b> the system ejects the ATM card.<br>2 <b>ABORT.</b><br>Postcondition ATM customer funds have not been withdrawn. The system is idle. The system is displaying a Welcome message. |
|--------------------------------------|---|

|                                     |   |
|-------------------------------------|---|
| Global Alternative Flow<br>"alt3" ▼ | ATM customer enters Cancel.<br>1 The system cancels the transaction.<br>2 The system ejects the ATM card.<br>3 <b>ABORT.</b><br>Postcondition ATM customer PIN number has not been withdrawn. The system is idle. The system is displaying a Welcome message. |
|-------------------------------------|---|

图 9 取款用例的需求规格说明截图



3) 用例结束

后置条件：**ATM 用户没有取款。系统服务关闭。**

(8) 分支流 alt2:

1) 系统提示抱歉信息，同时系统退卡

2) 用例结束

后置条件：**ATM 用户没有取款。系统闲置。系统显示欢迎界面。**

(9) 分支流 alt3: 为全局分支流，如果在任何操作环节，用户选择取消业务，将进入本分支流。

1) 系统取消事务

2) 系统退卡

3) 用例结束

后置条件：**ATM 用户没有取款。系统服务关闭。**

根据如上描述，可以通过条件判断（基本流 5,6,7,8）进入分支描述不同的场景，同时可以对不同场景分别进行后置条件描述。

# 附录 A

| 编号  | 描述                                    | 解释  |
|-----|---------------------------------------|---|
| R1  | 语句的主语应该为系统（the system）或者参与者（an actor） | 强制正确描述事件流，同时保证可以描述 3 类交互：<br>1. 首要/次要外部参与者->系统：外部参与者向系统发送请求或者数据<br>2. 系统->系统：系统验证请求或者数据<br>3. 系统->首要/次要外部参与者：系统发送反馈信息给外部参与者 |
| R2  | 顺序描述事件流                               |   |
| R3  | 不允许出现外部参与者对外部参与者的动作                   |   |
| R4  | 每一个语句描述一个动作，避免出现复合谓语                  | 在一个语句中如果出现多个动作，难于区分其顺序  |
| R5  | 使用现在时                                 | 强制用户描述什么是系统执行的，什么是外部参与者参与的，强制在一条语句中必须包含语句与宾语。（在中文描述中可以忽略语态与时态，但是必须清楚描述主语与宾语）  |
| R6  | 使用主动语态，避免使用被动语态                       |   |
| R7  | 清楚的描述交互的主语为宾语                         |   |
| R8  | 使用陈述语句，避免使用疑问句等                       | 书写需求的基本规则   |
| R9  | 保持使用词语/单词的一致性                         | 保持使用同一个词语描述一件事物   |
| R10 | 避免使用情态动词                              | 用于避免情态动词与副词描述事件的不确定性（针对英语语法，在中文中需要保证语句描述的确定性），支持具有准确度量的副词描述，如一次等。   |
| R11 | 避免使用副词                                |   |
| R12 | 使用简单句                                 | 用于自动解析自然语言，减少二义性（主要针对英语的特点，在中文中保证严格遵守 R13 与 R14）  |
| R13 | 避免使用否定副词与形容词（如绝不，从不等），可以使用否定语句（如没有）   |   |
| R14 | 避免使用指示代词                              |   |
| R15 | 避免使用分词短语                              |   |
| R16 | 使用系统统一指代系统                            | 保持描述系统的统一性，减少二义性的描述   |