

作者: Martin Glinz

需求工程专业认证术语表

含中文的术语词典

专业需求工程师认证的标准术语表，供学习和考试之用。术语的中文定义仅为阅读方便，如有错误，不清晰或与英文定义不一致的地方，则以英文定义为准！

英文原版版本号 V1.6, 2014 年 5 月发布

中文版版本号 V1.2, 2018 年 4 月发布

中文版由钱发强、朱姝翻译

Version of the original English edition V1.6, May 2014

Version of the Chinese edition V1.2, April 2018

Translated by Faqiang Qian and Shu Zhu



**University of
Zurich**^{UZH}

Department of Informatics

Requirements
Engineering
Research
Group



作者介绍

马丁·格林兹 (Martin Glinz) 是瑞士苏黎世大学信息专业的教授，担任需求工程研究室主任。他的研究范围涉及需求工程和软件工程，特别是与建模、验证和质量保证相关的工作以及软件工程教学研究。

马丁在亚琛工业大学取得数学硕士和计算机科学（自然科学）博士。加入苏黎世大学之前，他在工业界工作了十年。这十年里，他主要从事软件工程研究、开发、培训和咨询。

他在需求工程的学术界和工业界拥有 25 年的经验积累，是需求工程杂志的编委之一。从 2006 至 2009 年，他担任 IEEE 国际需求工程会议的指导委员会主席。

使用条款

如果承认版权，并将其引用在专题讨论的材料中，个人和培训机构就可以使用这个词汇表作为专题讨论的基础。任何人想在广告中使用这个术语表，都需要 IREB e. V. 对此用途的书面许可。

任何个人或团体，如果承认作者的版权，标明 IREB e. V. 为资料来源和此文件的所有者的话，即可以使用这个词汇表作为论文、书籍或其它衍生出版物的基础。

© 2011 – 2014 国际需求工程 IREB e. V. 编委会和 Martin Glinz

版权所有。电子拷贝或硬拷贝允许个人或用于培训目的之用。未经作者或 IREB e. V. 事先的书面许可，禁止以任何其他方式的复制、存储于检索系统中，或以任何形式或通过任何手段进行传播，如通过电子的、机械的、复印的、录制的或其它的方式。

鸣谢

中文版译者注：本大纲的中文翻译由夏勇、洪浪、刘璘校对，特此鸣谢。
Note by the translator for Chinese version: Special acknowledgements are due to Yong Xia, Lang Hong and Lin Liu who reviewed the Chinese translation carefully.

版本更新记录

版本	日期	备注
1.0	2013 年 10 月	初始版本
1.1	2018 年 1 月	修改页眉/页脚 修正格式化和错别字问题 更正术语“需求规格说明”
1.2	2018 年 4 月	更正版本更新记录



目录

鸣谢 3

版本更新记录 3

目录 4

IREB 专业需求工程师认证（CPRE） 5

导言 6

第一部分：定义与缩略词 7

 术语定义 7

 缩略语表 24

第二部分：词典 25

 英文 - 中文词典 25

参考资料 32

IREB 专业需求工程师认证（CPRE）

国际需求工程委员会成立于 2007 年。它是由有着深厚需求工程背景知识的专家组成，覆盖广泛的领域，包括工业、咨询、研究和教育。他们中的许多人由于对这个领域所做的贡献而闻名于世。

委员会成员创建了一个需求工程领域的教程和认证，即专业需求工程师认证，修完此课程便能获得证书。其目的是建立需求工程领域既定的标准化的、高质量的教学和持续的培训，从而提高需求工程的实践。本文中提供的术语表是 IREB CPRE 的标准术语表。2007 年，国际需求工程委员会在德国、奥地利和瑞士率先取得成功。自从提供了英语课程之后，专业需求工程师认证 (CPRE) 便被推广到越来越多的国家，并已真正地走向国际化。它的西班牙语和葡萄牙语（巴西）课程正在翻译中。

认证过程中所涉及的四个主要参与方：国际需求工程委员会 (IREB)，被认可的培训机构，不同国家的认证机构，然后就是参加培训课程的学员和考生。

IREB 安排课程表、出相应的考试题、定义并规范考试过程、授权认证机构管理考试、确认培训机构的培训课程是否符合专业需求工程师认证课程。在不同国家，IREB 委托的专业认证机构负责管理证书考试。

形式上，IREB 的课程类似于其他已建立起来的教学和持续培训标准的课程（例如，国际软件测试认证委员会（ISTQB）认证测试人员），并兼顾了 ISO 和 IEEE 的相关标准。初级课程包括需求工程的基本知识，包括需求获取、编写文档、验证和管理。IREB 认证所覆盖的内容可以在公开课程中查阅到。

通过其教程，IREB 提供一套在培训中所需要材料、培训内容以及达到学习目标和进行实际演练所需的时间等的指导方针。这套需求工程专业词汇术语表和其它补充材料是对 IREB 教学大纲的补充。

有关国际需求工程委员会（IREB e. V.）和专业需求工程师认证的所有信息都能在 IREB 的网站上找到。

<http://www.ireb.org>

导言

如果想要查找需求工程术语的定义，你可以在网上搜索找到几乎所有术语的定义。然而，这样的搜索很费时，而且结果的质量无法保证。通常情况下，查找到的不同来源的定义之间存在歧义。现有的需求工程的教科书中的术语表通常只针对本教科书所涵盖的章节。一直缺乏将术语系统地翻译成英语以外的主要语种。

本术语表的目的是收集需求工程术语的现有的知识，并仔细地、连贯地定义核心术语。在超过一个定义的情况下，或从不同视角有不同的定义时，多个定义或视角都将包括在内。对于既具有一般的含义，又有需求工程上下文特定的含义的术语，两个含义都将包括在内。重要的术语将补充提示和附加信息。此外，所有术语均翻译成了德语。其他语言的翻译，尤其是法语和西班牙语，已在计划中。

这个术语表是对国际需求工程委员会（IREB）认可的课本的补充。在即将出版的由 Klaus Pohl 和 Chris Rupp 编写的教材“需求工程基础”一书中的术语定义和这个词汇表中的术语定义已经做了双向核准。

参考文献中列出了我所定义术语的来源。我不去列举个别定义来源，因为我觉得不能简单地通过复制和粘贴去编译从现有的各种来源中得到的定义，而是根据当今的用法，仔细地连贯地重新表述所有的定义。当然，这并不排除一些定义或其中某些部分逐字引用了所列出的文献之一，我并不打算彻底改造世界。有些定义是与他人协作的结果。基于本人在需求工程领域超过 25 年的经历，我在学术界和工业界使用术语的个人知识和经验，是另一个我不能也不用引用的来源。在第 54 页的“来源”部分，给出了对定义来源及协作工作的鸣谢。

这是一个不断推进的工作。目前已经定义了 128 个术语，这在很大程度上覆盖了基本术语。就流程、项目管理和产品管理方面的相关术语，还存在一些欠缺。一些专门的术语，如需求获取或解决冲突的具体技术术语还待定义。我打算逐步填补这些空白。

本术语表由两部分组成：英文的术语定义及其它语言的词典。我希望学员和需求工程的专业人士会发现这种结构非常有用，并认为值得去使用此术语表。

我想感谢 IREB 的成员，特别是 Karol Frühauf, Colin Hood, Klaus Pohl, Chris Rupp 及他的学者团队, Thorsten Weyer, 感谢他们对此术语表所做的贡献和许多建议，这些建议帮助提高了质量。我也非常感谢很多人耐心地等待了一年多，等待这个术语表的出现。最后，我要感谢我的妻子 Angelika。如果没有她的爱心、耐心和理解，我的大部分专业工作，包括这个术语表将不可能实现。

Martin Glinz, 2011 年 5 月于苏黎世

第一部分：定义与缩略词

术语定义

Acceptance	验收	评估一套系统是否满足其所有需求的过程
Acceptance test	验收测试	评估一套系统是否满足其所有需求的测试
Activity diagram	活动图	是 UML 中的一个图类，它模拟一个系统或组件中的动作流，包括数据流和那些必要的责任划分。
Actor	施动者	<ol style="list-style-type: none">1. 在需求工程中一般是指系统环境中与系统进行交互的一个人、一套系统或技术设备。2. 在面向目标的需求工程中特指为了实现某些目标可能会采取行动并处理信息的一个人、一套系统或技术设备。
Adequacy (of a requirement)	(需求的) 充分性	一个需求表达出利益相关者真正的愿望和需要的程度（即他们在陈述需求时实际上在心里想的那些需求）。
Application Domain	应用领域	现实世界中那些确定系统环境的相关部分。
Artifact	制品	系统开发的中间或最终结果，例如需求规格说明书。
Attribute	属性	一个实体的特性。
Baseline	基线	制品的一个稳定的、变更控制的配置。基线用于版本规划和版本定义及项目管理，如估算工作量。

Behavior model	行为模型	描述一个系统或组件行为的模型，比如状态机、活动图。
Bug	缺陷	见 Defect
Cardinality	基数	<p>1. 在建模中表示对象在一种关系中的最小和最大数目。</p> <p>在 UML 中，用多重数 (Multiplicity) 表示基数 (Cardinality)。</p> <p>2. 在数学上表示一个集合的元素数量。</p>
Change control board	变更控制委员会	一个由客户和供应商代表组成的一个来决定变更请求的委员会。简称：CCB
Change request	变更请求	在需求工程中指一个论据充分的请求，要求改变一个或多个已经设置了基线的需求。
Changeability (of an artifact)	(制品的) 易改变性	制品能够满足所需改动的程度。
Checking (requirements)	检查 (需求)	包括验证需求和检查需求质量，如明确性或可理解性。 注：一些文献给出了验证的更广泛的定义，并且认为检查和验证是同义词。
Class	类	表示一组在结构上、操作方式和行为上相同类型的对象。
Class diagram	类图	一个类模型的图形表达。
Class model	类模型	包含一组类和它们之间关系的模型。

Completeness (of requirements)	(需求的) 完整性	<ol style="list-style-type: none">1. 对于一个单一的需求：一个需求包含所有必要的信息的程度。2. 对于需求规格说明书：需求规格说明书所包含的对于开发一个满足利益相关者愿望和需要的系统所有必要的信息的程度。
Compliance	遵从	制品遵从标准、法规、法律或其它正式实施的文件的能力。系统常常需要服从系统所在地的标准、法规 and 法律的制约。只有从检查需求的遵从性开始，才能确保这种遵从性的系统性。
Component	组件	<ol style="list-style-type: none">1. 一般而言指系统的可定界的部分。2. 在软件架构中指一套共同提供服务的相关联的对象或类。 <p>注：孤立地看，组件本身是一个系统。</p>
Configuration	配置	逻辑上连贯的各个单位的相容的集合。单位是在一个版本中的各个可识别的制品或制品部件（如需求）。
Conformity (of requirements)	(需求的) 符合性	需求规格说明符合某些标准的程度。
Consistency (of requirements)	(需求的) 一致性	对一套需求的描述无前后矛盾程度。
Constraint	约束，边界条件	一种需求，它限制解决方案在那些必要的满足给定的功能需求和质量需求之外的空间。

Context	环境，上下文	<p>1. 一般是指对理解现象或言论所需要的思维和含义网。</p> <p>2. 在需求工程中特指系统周边对理解系统和其需求紧密相关的环境部分。</p> <p>第二个含义也称为系统环境。</p>
Context Boundary	环境边界	<p>一个系统的环境和那些对系统和其需求无关紧要的应用范围之间的边界。</p> <p>环境边界将待开发的系统周边相关的部分从不相关的部分，即那些不影响系统的开发，因而在需求工程的过程中不必考虑的部分，分离开来。</p>
Context diagram	环境图，上下文图	<p>1. 环境模型的一种图形表达。</p> <p>2. 在结构化分析中，环境图（上下文图）是分层次数据流图的最顶层。</p>
Context model	环境模型，上下文模型	描述系统在其环境中的模型。
Correctness	正确性	包含在制品中的信息可被证明是真实的程度。在需求工程中，正确性经常被用作充分性的代名词。
Customer	客户	接受产品或服务的个人或组织。也请参见利益相关者。
Customer requirements specification	客户需求说明	从客户的角度对一个系统所需功能的粗略描述。通常由客户提供。

Dataflow Diagram	数据流图	以进程（也称为活动）、数据存储和数据流的形式对系统或组件的功能进行建模的图形。输入数据流触发处理进程，处理程序然后消费接收到的数据，进行转换，读写数据存储器中的持久性数据，然后产生新的数据流。该数据流可能是中间结果将触发其它进程，或是最终结果而退出系统。
Decision table	决策表	一种系统地描述一个复杂、多重准则决策逻辑的图表结构表示。
Defect	缺陷	一个制品中的缺点、不正确的描述或制作。也称故障
Domain	领域，范围	（对某些给定的事情）相关东西的范围，例如，一个应用范围。
Effectiveness	有效性	某些事情以其应该发生的方式所实际发生的程度。在需求工程中主要是指，系统能够让用户实现他们在系统需求说明中所列出的目标的程度。
Efficiency	效率	以最小的资源消耗所达到一种结果的程度。
Elicitation (of requirements)	(需求的) 获取	见Requirements elicitation
End user	最终用户	见 User
Entity	实体	<ol style="list-style-type: none"> 一般而言，实体是指一个元素或元素的集合，它可以表示任何可想象的产品，例如，一个系统、现实的一部分、一件东西、一个组织、一个过程等。 在实体关系建模中指单个对象，它具有一个标识，并且不依赖于另一个对象。

Entity-relationship diagram	实体关系图	实体关系模型的图形表示。 缩写为 ERD
Entity-relationship model	实体关系模型	一个系统或一个应用域数据的数据模型。一个实体关系模型由一组具有各自特性的并且通过关系链接的实体类型组成。 缩写：ERM，ER 模型
Error	错误	观察到的行为或结果和规定的行为或结果之间的差异。 错误通常是在某一制品中存在的故障或缺陷的表征。 在英语口语中错误和故障之间有时没有区别。
Fault	故障	见 Defect
Fault Tolerance	容错性	尽管有（硬件或软件上的）故障存在，系统继续进行正常操作的能力。 故障容限可以作为质量需求。
Feature	特征	一套为利益相关者提供价值的系统可界定的特征。 一般包括多种需求，并用于与利益相关者在更高度的抽象上的沟通以及表达可变的或供选的特征。
Functional Requirement	功能性需求	有关由系统（或组件或服务）的一项功能应该提供的行为结果的需求。
Functionality	功能性	一套系统如由功能性需求所规定的的能力。
Glossary	术语表	在一些领域中相关术语定义的集合。通常情况下，术语表还包含交叉引用词、同义词、同音异义词、首字母缩写词和缩略语词。

Goal	目标	事情的一个理想状态（即利益相关者想要达到的目标）。 目标描述利益相关者的意图。它们可能互相冲突。
Goal model	目标模型	表示某些事情目标的模型，为子模型的一种有序的结构。
Homonym	同形同音异义词	一个术语和另一个术语写法相同，但具有不同的含义。例如，bill 为银行票据，而 bill 作为一个（材料的）列表，它们是同形同音异义词。
Inspection	审查	一种评审，由一个专家小组按给定的标准检查制品。专家的审查意见收集起来后再做整理。
Kind of requirement	需求类型	有多种类型的需求。需求工程主要关心的是系统需求。除此之外，还有项目需求和工艺需求。 需求通常分为功能需求、质量需求和约束（边界条件）。后两者也被称为非功能性需求。
Language	语言	一套结构化的表达和交流信息的符号。符号是用于交流的元素：表达一种语言、记号、手势等。
Maintainability	易维护性	一套软件系统能够修改的难易程度，以纠正故障或适配系统不断变化的需求。 易维护性可作为质量需求。

Model	模型	<p>一个存在的现实或一个将要创建的现实的抽象表达。</p> <p>这个定义涵盖了需求工程中最常见的情况，但稍微有点偏窄。更广泛地说，模型是现有实体或将要创建的实体的表达，而实体表示现实的任何部分或任何其它可能的元素或现象，包括其它模型的组合。就模型而言，实体称为原件。</p> <p>在需求工程中，需求可用模型来说明。</p> <p>注：此处定义的实体是它一般性的含义，不同于它在实体关系模型中的含义。</p>
Modeling Language	建模语言	以一种特定的形式来表达模型的语言。可以是文字、图形、符号或它们的某种组合。
Multiplicity	多重性	见 Cardinality
Non-functional Requirement	非功能性需求	<p>一个质量需求或一个边界条件。</p> <p>性能需求可被视为另一类的非功能性需求。在本术语表中，性能需求被认为是质量需求的一个子类。</p> <p>同义词：额外功能性需求（Extra-functional requirement）</p>
Performance Requirement	性能需求	<p>描述性能特性（时间、速度、体积、容量、吞吐量）的需求。</p> <p>在本术语表中认为是质量需求的一个子类，但也可以认为是它自己的一个非功能性需求类。</p>
Phrase template	句型模板	一个以自然语言表达一条需求语句的句法结构的模板。

Portability	可移植性	一套系统可以转移到另一个平台上（而不影响其功能）的难易程度。 可移植性可以作为质量需求。
Priority (of a requirement)	（需求的）优先级	根据给定的原则，一个需求相较于其它需求的重要性。
Process verb	过程动词，过程词	用自然语言编写的需求中，用来表征所需动作的动词。
Prototype	原型	1. 在制造业：在大规模生产开始前所制造的一件（产品）。 2. 在软件工程中：一套可执行的软件，它预先执行系统的关键部分。 在需求工程中，原型是用于获取需求和验证需求的一种手段。
Quality	质量	实体的一组固有特性满足需求的程度。 实体可以是一个系统、服务、产品、制品、过程、人、组织等等。固有特性是一个实体的显著特征或属性，它固有于实体，但没有明确地指定给实体。 这是在工业中通常使用的质量的概念。需要注意的是，在这个定义上，质量只意味着如在需求说明中陈述的使用的合适性。它不同于口语中意味着卓越的品质质量概念。
Quality Requirement	质量需求	涉及质量的需求，是非功能性的需求。
Redundancy	冗余	相同的信息或资源的多重出现。
Release	发布	已经发布的用于客户安装和使用的配置。

Reliability	可靠性	一套系统在规定的条件下运行时，维持其功能和性能达到规定的水平的能力。 可靠性可作为质量需求。
Requirement	需求	<ol style="list-style-type: none"> 1. 用户为解决某一问题或达到某个目标所需要的条件或能力。 2. 系统或系统部件必须满足或处理的条件或能力以满足合同、标准、规格说明或其它正式的强制性文件的要求。 3. 如在（1）、（2）中所述的条件或能力的文档表达。 <p>注：上面的定义是 1990 年 IEEE610.12 标准的一个经典定义。</p> <p>另外，我们还提供一个更现代的定义：</p> <ol style="list-style-type: none"> 1. 一个利益相关者所感觉到的需要。 2. 一套系统必须具备的一项能力或性能。 3. 对需要、能力或性能的文档表达。
Requirements analysis	需求分析	<ol style="list-style-type: none"> 1. 分析已经获取的需求以理解和编写这些需求的文档。 2. 需求工程的同义词。
Requirements Baseline	需求基线	一组需求的基准线。
Requirements Discovery	需求获取	见 Requirements elicitation
Requirements document	需求文档	一份由需求规格说明组成的文档。常作为需求规格说明书的代名词。

Requirements Elicitation	需求获取	从现有的需求来源中寻找、捕获和强化需求的过程。可包括重建或创建需求。 别名：需求发现 (Requirements discovery)
Requirements engineer	需求工程师	与利益相关者合作，进行需求获取、编制需求文档、验证和管理需求的人员。
Requirements Engineering	需求工程	以一套系统化和规范化的方法对需求进行规范说明和管理，有以下目标： <ol style="list-style-type: none"> 1. 了解相关需求，就这些需求在利益相关者中达成共识，按照给定的标准编写需求文档，并且系统地管理需求。 2. 了解并记载利益相关者的愿望和需求， 3. 制定需求规格说明书和管理需求，以将所提供的系统不符合利益相关者愿望和需求的风险降到最低。 缩写：RE 注：所有这三个目标指出了需求工程的重要方面：（1）过程导向（2）聚焦利益相关者（3）风险和价值考量的重要性。
Requirements management	需求管理	管理现有需求和与需求相关的制品的过程。特别包括需求的存储、变更和追踪（可追踪性）。
Requirements model	需求模型	以对需求进行规格说明为目的而已经创建的模型。
Requirements source	需求来源	导出需求的来源。典型的来源有利益相关者、文档、现有的系统和观察。

Requirements specification	需求规格说明	<p>通常对一个满足给定条件的系统或组件系统地描述的需求集合。</p> <p>在某些情况下，我们区分客户需求说明书（通常由客户编写）和系统需求规格说明书或软件需求规格说明书（由供应商编写）。需求规格说明也可以表示对需求进行规格说明的活动。</p>
Requirements template	需求文档模板	<p>需求的句法结构的蓝图。</p> <p>语句模板是为以自然语言编写需求的一个特定的需求模板。</p>
Review	评审	<p>一个正式组织的、由专家小组检查一个制品的活动。</p> <p>可对内容和一致性进行检查。</p>
Risk	风险	<p>威胁到成功的事件，例如威胁到开发或运行一套系统成功的事件。通常以其概率和潜在的损害做风险评估。</p>
Safety	安全性（使用的安全性）	<p>运行一套系统不会导致对人、财产或环境造成损害所达到的可接受的概率水平的能力。</p> <p>安全性需求可以作为质量需求或功能性方面的需求加以说明。</p>
Scenario	场景，情景实例	<ol style="list-style-type: none"> 1. 描述一个潜在的能促成想要达到的（或不想要的）结果的事件序列。 2. 合作伙伴之间，特别是系统和外部施动者之间的一个有序的交互序列。可以是一个具体的序列（实例场景）或一组潜在的序列（类型场景、用例）。 3. 在 UML 中：执行用例跟踪。
Scope (of a system)	系统范围	<p>在开发一个系统时，可以规划和设计的东西的范围。</p>

Security	安全性（信息的安全性）	<p>（a）系统保护其数据和资源，防止未经授权使用的能力</p> <p>（b）系统保护其合法用户不被设备拒绝服务的能力。</p>
Semantics	语义	一个符号或一组符号在一种语言中的含义。
Semi-formal	半正式的, 半形式化的	<p>某些事情在一定程度上是正式的，但又不完全是正式的。</p> <p>如果一个制品包含正式的部分，但还没有全部正式化，这个制品称为半正式的。通常情况下，一个半正式的制品具有一个定义完整的句法，但语义定义不完整。</p>
Sequence diagram	顺序图, 时序图	是 UML 中的一个图类，它描述所选定的一组对象和（或）施动者之间按顺序的交互。
Software requirements specification	软件需求规格说明	有关软件系统的需求规格说明。简称：SRS
Source (of a requirement)	（需求的）来源	见 Requirements source
Specification	规格说明	<p>对一个满足给定准则的实体（一个系统，一个设备等）的属性所作的系统性的描述。</p> <p>可以是关于需求属性（需求规格说明）或实现属性（例如，产品技术规格说明）的说明书。</p>
Specification language	规格说明语言	一个为了表达规格说明书而创造的人造语言。

Stakeholder	利益相关者	一个对系统的需求有（直接或间接）影响的个人或组织。 间接影响也包括一个人或组织受系统影响的情况。
Standard	标准	一套对理解、制造或执行某些东西的统一的规则。
State machine	状态机	一个用一组有限的状态和状态转换来描述系统或组件行为的模型。状态转换是由事件触发，可以反过来触发动作和新的事件。 相关术语：具有原子态状的状态机称为有限状态机。具有分层和（或）正交分解状态的状态机称为状态图。
State-transition diagram	状态转换图，状态迁移图	状态机的图表表示。
Statechart	状态图	具有分层和（或）正交分解状态的状态机。
Steering committee	指导委员会	负责督导项目的委员会。
Structured Analysis	结构式分析，结构化分析	一种基于数据流图的层次结构对系统的功能进行规格说明的方法。数据流以及持久性数据在数据字典中有定义。环境图模写输入数据流的来源和输出数据流的目的地。
Supplier	供应商	为客户提供产品或服务的个人或组织。
Synonym	同义词	一个与另一个有相同含义的词。
Syntax	句法	在一种语言中构建有结构的符号的规则。

System	系统	<p>1. 通常情况下指顺序和结构的原则。</p> <p>2. 在信息学中指连贯的、可界定的一套通过协调的动作而提供服务的组件。</p> <p>需求工程注重对由软件组件、技术要素（电脑硬件、设备、传感器，...）和组织要素（人、岗位、业务流程，...）组成的系统的需求做规范说明。</p> <p>注：一个系统可以包含其它系统。因此，一个系统的组件或服务也可认为是系统。</p>
System boundary	系统边界	<p>一个系统和它周边环境之间的边界。</p> <p>系统边界把待开发系统从它的环境中分开，也就是说，系统边界把可以通过开发过程修改或改变的现实的一部分，从不能改变或修改的环境方面分离开来。</p>
System context	系统环境	<p>与定义和理解待开发系统需求相关的系统周边的环境部分。</p>
System requirement	系统需求	<p>涉及一个系统或系统组件需求。</p>
System requirements specification	系统需求规格说明	<p>涉及一个系统的需求规格说明书。</p> <p>经常看作是需求规格的代名词。</p>
Tool (in software engineering)	（软件工程）工具	<p>一套帮助开发、运行和维护系统的（软件）系统。</p> <p>在需求工程中，工具用来支持需求管理以及建模、编写文档和验证需求。</p>

Traceability (of requirements)	(需求的) 可追踪性	<p>(1) 向后追溯一个需求到它的起源的能力； (2) 向前追溯一个需求至其对应设计和实现的能力； (3) 追踪一个需求与其它需求之间依赖关系（反之亦然）的能力。起源可以是利益相关者、文档、逻辑依据等。</p> <p>回到需求起源的可追踪性也称为前 RS 可追踪性。相反，对一个需求前行至实现设计和编程的可追踪性也称为后 RS 可追踪性。RS 代表需求规格说明。</p> <p>有时，对需求的逻辑依据的可追踪性看作是它自身的可追踪性类别。</p>
UML	UML	统一建模语言（Unified Modeling Language）的缩写，一种为问题或解决方案建模的标准化的语言。
Unambiguity (of requirements)	(需求的) 明确性	一个需求的表达不能让不同的人有不同的理解的明晰程度。
Usability	可用性	<p>系统让用户理解、学习、使用、喜欢的能力。</p> <p>可用性（或其部分）可以作为质量需求。</p>
Use case	用例，用况	<p>描述可能在施动者和系统之间发生的交互，当系统执行时则提供附加价值。</p> <p>从用户（或其它外部施动者）的角度来看，用例对系统做规格说明：每一个用例都描述一些系统必须向用例中所涉及的施动者提供的功能。</p>
Use case diagram	用例图，用况图	<p>是 UML 中模写施动者和系统用例的一类图表。</p> <p>施动者和用例之间的边界构成了系统边界。</p>
User	用户	一个使用系统所提供的功能的人。也称为终端用户。

Validation (of requirements)	(需求的) 确认	<p>检查需求文档是否与利益相关者的需求相匹配的过程。</p> <p>注：一些文献定义了更广泛的需求确认，还包括检查质量需求，如明确性或可理解性，从而认为是确认的同义词。</p>
Verifiability (of requirements)	(需求的) 可验证性	<p>可以检查系统满足需求的程度，如通过定义验收测试例、测量或检查程序等来检验。</p>
Version (of an entity)	(实体的) 版本	<p>如果一个实体存在于多重的、有时序的事件中，每次事件的发生都是通过修改它的前导事件而创建的，那么每次事件的发生就是该实体的一个版本。</p>
View	视图	<p>对一个制品摘选只是那些目前感兴趣的部分。</p> <p>视图可以提取或聚合一个制品的多个部分。</p>
Viewpoint	观点，视角	<p>对系统需求的一个特定的视角。</p> <p>典型的观点是利益相关者或利益相关者群的看法（如，最终用户的观点，运营商的观点）。然而，也可以是某个专题的观点，如从安全性角度出发的观点。</p> <p>注：这个定义与国际标准 ISO/IEC42010:2007（IEEE 标准 1471-2000）对架构观点（architectural viewpoint）的定义有所不同。</p>
Walkthrough	走查，预排	<p>评审的一种方式。评审时，制品的作者引导专家小组对制品系统地查看一遍。专家的审查意见收集起来后再做整理。</p>

缩略语表

CCB	Change Control Board	变更控制委员会
CPRE	Certified Professional for Requirements Engineering	专业需求工程师认证
ER	Entity-Relationship	实体关系
ERD	Entity-Relationship Diagram	实体关系图
ERM	Entity-Relationship Model	实体关系模型
IREB	International Requirements Engineering Board	国际需求工程委员会
RE	Requirements Engineering	需求工程
SRS	Software Requirements Specification	软件需求规格说明
UML	Unified Modeling Language	统一建模语言

第二部分：词典

英文 - 中文词典

英文	中文
Acceptance	验收
Acceptance test	验收测试
Activity diagram	活动图
Actor	施动者，行为者
Adequacy (of a requirement)	(需求的) 充分性，恰当性
Application domain	应用领域，应用范围
Artifact	制品
Attribute	属性
Baseline	基线
Behavior model	行为模型
Bug	缺陷
Cardinality	基数
Change control board	变更控制委员会
Change request	变更请求
Changeability (of an artifact)	(制品的) 易改变性
Checking (requirements)	检查 (需求)
Class	类
Class diagram	类图

Class model	类模型
Completeness (of requirements)	(需求的) 完整性
Compliance	遵从, 符合
Component	组件
Configuration	配置
Conformity (of requirement)	(需求的) 符合性
Consistency (of requirements)	(需求的) 一致性
Constraint	约束, 边界条件
Context	环境, 上下文
Context boundary	环境边界
Context diagram	(系统) 环境图, 关系图
Context model	环境模型
Correctness	正确性
Customer	客户
Customer requirements specification	客户需求说明
Dataflow diagram	数据流图
Decision table	决策表
Defect, Bug, Fault	缺陷
Domain	领域, 范围, 域
Effectiveness	有效性
Efficiency	效率

Elicitation (of requirements)	(需求的) 获取
End user	最终用户, 终端用户
Entity	实体
Entity-relationship diagram	实体关系图
Entity-relationship model	实体关系模型
Error	错误
Fault	故障
Fault Tolerance	故障容限
Feature	特性
Functional requirement	功能性需求
Functionality	功能性
Glossary	术语表
Goal	目标
Goal model	目标模型
Homonym	同形同音异义词
Inspection	审查
Kind of requirement	需求类型
Language	语言
Maintainability	易维护性
Model	模型
Modeling language	建模语言

Multiplicity	多重性
Non-functional requirement	非功能性需求
Performance requirement	性能需求
Phrase template	句型模板
Portability	可移植性
Priority (of a requirement)	(需求的) 优先级
Process verb	过程动词, 过程词
Prototype	原型
Quality	质量
Quality requirement	质量需求
Redundancy	冗余
Release	发布
Reliability	可靠性
Requirement	需求
Requirement analysis	需求分析
Requirements baseline	需求基线
Requirement discovery	需求获取
Requirements document	需求文档
Requirements elicitation	需求获取
Requirements engineer	需求工程师
Requirements Engineering	需求工程

Requirements management	需求管理
Requirements model	需求模型
Requirements source	需求来源
Requirements specification	需求规格说明
Requirements template	需求文档模板
Review	评审
Risk	风险
Safety	安全性（使用的安全性）
Scenario	场景，情景实例
Scope of a system	系统范围
Security	安全性（信息的安全性）
Semantics	语义
Semi-formal	半正式的
Sequence diagram	顺序图
Software requirements specification	软件需求规格说明
Source (of a requirement)	（需求的）来源
Specification	规格说明
Specification language	规格说明语言
Stakeholder	利益相关者
Standard	标准
State machine	状态机

Statechart	状态图
State-transition diagram	状态转换图, 状态迁移图
Steering committee	指导委员会
Structured Analysis	结构式分析, 结构化分析
Supplier	供应商
Synonym	同义词
Syntax	句法
System	系统
System boundary	系统边界
System context	系统环境
System requirement	系统需求
System requirements specification	系统规格说明
Tool (in software engineering)	(软件工程) 工具
Traceability (of requirements)	(需求的) 可追踪性
UML	统一建模语言 UML
Unambiguity (of requirements)	(需求的) 明确性, 无二义性
Usability	可用性
Use case	用例, 用况
Use case diagram	用例图, 用况图
User	用户
Validation (of requirements)	(需求的) 确认

Verifiability (of requirements)	(需求的) 可验证性
Version (of an entity)	(实体的) 版本
View	视图, 看法
Viewpoint	观点
Walkthrough	走查, 预排

参考资料

- Gause, D.C., G.M. Weinberg (1989). *Exploring Requirements: Quality before Design*. New York: Dorset House.
- Glinz, M. (2010). Course notes on *Informatik II (Modeling)*, *Requirements Engineering I*, and *Software Engineering*.
<http://www.ifi.uzh.ch/req/teaching/>
- Glinz, M. (2007). On Non-Functional Requirements. *Proceedings of the 15th IEEE International Requirements Engineering Conference*, Delhi, India. 21-26.
- Glinz, M., R. Wieringa (2007). Stakeholders in Requirements Engineering (Guest Editors' Introduction). *IEEE Software* **24**, 2. 18-20.
- Gotel, O., A. Finkelstein (1994). An Analysis of the Requirements Traceability Problem, *Proceedings of the First International Conference on Requirements Engineering*, Colorado Springs. 94-101.
- IEEE (1990). *Standard Glossary of Software Engineering Terminology*. IEEE Std 610.12-1990.
- IEEE (1993). *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Standard 830-1993.
- IREB (2010). *Certified Professional for Requirements Engineering Foundation Level Syllabus*, Version 2.1. <http://www.certified-re.de/en/syllabi/foundation-level.html>
- ISO/IEC (2001). *Software Engineering—Product Quality—Part 1: Quality Model*. ISO/IEC Standard 9126-1:2001, International Organization for Standardization.
- ISO/IEC (2007). *Systems and Software Engineering — Recommended Practice for Architectural Description of Software-Intensive Systems*. ISO/IEC Standard 42010: 2007 (equal to IEEE Std 1471-2000) International Organization for Standardization.
- ISO (2005). *Quality Management Systems—Fundamentals and Vocabulary*. ISO Standard 9000:2005, International Organization for Standardization.

Mylopoulos, J. (2006). *Goal-Oriented Requirements Engineering: Part II*. Presentation slides of keynote talk at the 14th IEEE International Requirements Engineering Conference (RE' 06), Minneapolis, USA.

Pohl, K. (2007). *Requirements Engineering: Grundlagen, Prinzipien, Techniken*. Heidelberg: dpunkt.

Pohl, K. (2010). *Requirements Engineering: Fundamentals, Principles, and Techniques*. Berlin-Heidelberg: Springer.

Pohl, K., Rupp, C. (2010). *Basiswissen Requirements Engineering*. 2. Auflage. Heidelberg: dpunkt.

Pohl, K., Rupp, C. (2011). *Requirements Engineering Fundamentals*. Santa Barbara, Ca.: RockyNook.

Robertson, S., Robertson, J. (2006). *Mastering the Requirements Process*. 2nd edition, Addison-Wesley.

Rupp, C. et al. (2009). *Requirements Engineering und -Management: Professionelle, iterative Anforderungsanalyse für die Praxis*. 5. Auflage. München: Hanser.

Stachowiak, H. (1973). *Allgemeine Modelltheorie*. Wien: Springer.

Wikipedia. <http://de.wikipedia.org> und <http://en.wikipedia.org>. Visited Jan-Mar 2010.

Zowghi, D., C. Coulin (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools. In A. Aurum, C. Wohlin: *Engineering and Managing Software Requirements*. Berlin: Springer. 19-46.