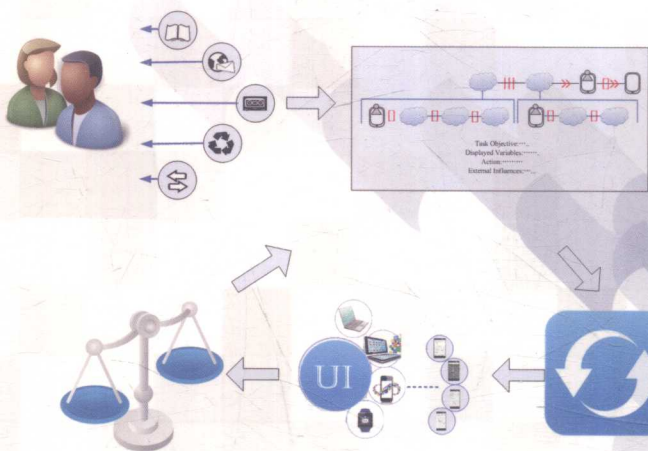


西安石油大学优秀学术著作出版基金资助



# 基于任务模型的用户界面设计与开发

..... 李娟妮 著 .....



中国石化出版社

[HTTP://WWW.SINOPEC-PRESS.COM](http://www.sinopec-press.com)

西安石油大学优秀学术著作出版基金资助

# 基于任务模型的 用户界面设计与开发

李娟妮 著



中国石化出版社

## 内容提要

本书系统地介绍了一种开发高可用性软件的方法——基于任务模型的用户界面开发。书中以文献资料和开发实例为基础,介绍了建立用户任务模型的意义,不同任务模型的特点,以及从任务模型到界面生成的过程;同时,还分析了适用于多用户、多设备、多平台的移动计算环境中的用户任务建模过程,以及从任务模型到界面原型的转换方法。

本书对提高移动交互式系统的可用性、降低开发成本等具有参考价值,也可以作为从事交互设计、视觉设计、产品开发和用户研究等专业的工程技术人员、科研工作者及高等院校师生的参考用书。

## 图书在版编目(CIP)数据

基于任务模型的用户界面设计与开发 / 李娟妮著.  
—北京:中国石化出版社,2018.9  
ISBN 978-7-5114-4978-8

I. ①基… II. ①李… III. ①人机界面-程序设计 IV. ①TP311.1

中国版本图书馆 CIP 数据核字(2018)第 184486 号

未经本社书面授权,本书任何部分不得被复制、抄袭,  
或者以任何形式或任何方式传播。版权所有,侵权必究。

### 中国石化出版社出版发行

地址:北京市朝阳区吉市口路9号  
邮编:100022 电话:(010)59964500  
发行部电话:(010)59964526  
<http://www.sinopec-press.com>  
E-mail:press@sinopec.com  
北京富泰印刷有限责任公司印刷  
全国各地新华书店经销

\*

710×1000毫米 16开本 10.25印张 210千字  
2018年9月第1版 2018年9月第1次印刷  
定价:52.00元

## ▣ 前 言

随着各种类型的计算设备大量涌入消费者市场，信息技术已经渗入人们生活的物理空间，当前的计算环境已由个人计算演化为普适计算。用户从具有专业知识和技能的工作者扩展到不同年龄、不同背景的普通大众，对于信息技术的使用也延伸到社会的各个领域。在普适计算环境中，人们希望随时、随地获取信息和计算服务。用户的非专业性、使用上下文的动态性、任务的不确定性、设备和开发平台的多样性给用户界面开发带来了很大的挑战，主要包括下述几个方面。①可用性方面：如何为不同设备开发同样方便易用，并具有一致用户体验的界面；如何为不同类型的用户在不同的使用环境中提供不同的界面；如何获取环境中的隐式数据，使桌面计算模式下的显式交互转换为普适计算模式下的隐式交互。②技术方面：开发者如何有效地为同一应用在不同技术环境（设备、交互通道、开发平台等）中分别设计实现不同的用户界面；如何有效管理多个版本界面开发的工程复杂性。

针对上述问题，本书试图给出一种开发高可用性系统的有效方法——基于任务模型的用户界面开发，并从3个层次进行了阐述。首先介绍了任务建模的基础知识，第一章从宏观层面描述什么是可用性，为什么在设计和开发过程中要保证系统的可用性；第二章介绍了典型的用

户任务建模方法；第三章描述了如何利用任务模型开发高可用性的交互式产品；随后，在第四章、第五章中系统地介绍了一种适用于移动环境中基于任务模型开发高可用性系统的方法和过程，并在第六章中给出设计实例，使读者更加容易理解抽象的理论方法如何应用到实际设计开发过程中；最后，在第七章中介绍了支持此开发方法的工具及评估过程。

本书在编写过程中，得到了西北大学华庆一老师、吴昊老师、姬祥老师，西安财经学院常言说老师等的支持和帮助，在此谨向他们表示诚挚的感谢。由于编者水平有限，书中难免会有疏漏和不足之处，希望广大读者提出宝贵意见和建议。

## 目 录

第一章 可用性设计 .....	1
第1节 可用性定义 .....	2
第2节 交互模型 .....	6
第3节 基于任务模型的开发过程 .....	9
参考文献 .....	12
第二章 用户的任务模型 .....	13
第1节 用户任务模型介绍 .....	14
第2节 任务建模过程的心理学基础 .....	17
第3节 几种典型的任务模型 .....	21
参考文献 .....	33
第三章 基于任务模型的界面开发 .....	37
第1节 CAMELEON 参考框架 .....	38
第2节 任务模型在 MBUID 中的作用 .....	44
参考文献 .....	47

<b>第四章 一种适用于移动计算环境的任务建模方法</b> .....	49
第1节 移动计算环境下的用户任务 .....	50
第2节 用户任务建模框架 .....	54
第3节 PCTBTA 任务元模型 .....	56
第4节 PCTBTA 任务建模过程 .....	69
第5节 PCTBTA 任务模型的图形化表示 .....	73
第6节 说明性实例 .....	76
参考文献 .....	80
<b>第五章 基于 PCTBTA 任务模型的用户界面开发</b> .....	82
第1节 基于 PCTBTA 任务模型的用户界面开发框架 .....	82
第2节 PCTBTA 任务模型中提取 ETS .....	85
第3节 ETS 到对话模型的转换 .....	93
第4节 对话模型到呈现模型的转换 .....	101
参考文献 .....	107
<b>第六章 应用实例</b> .....	109
第1节 “校园导航系统”的设计 .....	109
第2节 “飞机票务系统”的设计 .....	126
第3节 与其他方法的比较分析 .....	133
参考文献 .....	143
<b>第七章 任务建模支持工具</b> .....	144
第1节 任务建模工具介绍 .....	144
第2节 PCTBTA 任务建模的支持工具 .....	147
第3节 工具的对比评估 .....	154
参考文献 .....	156

## 第一章

# 可用性设计

信息化时代中，我们的生活、学习、工作都离不开各种各样的软件，如何评价一个软件的优劣，首先要需要考虑软件是否能帮助人们完成某个具体任务或者给人们带来精神上的享受，即它的存在价值。如办公软件能够辅助专业人员完成某项工作；游戏软件能够提供虚拟场景，给玩家带来身临其境的感受；音乐播放器软件能够通过播放音乐给用户带来精神上的享受。这些称为软件的功能性，它定义了开发人员必须实现的软件功能，使用户能够完成他们的任务，从而满足需求。但如果完成同样功能的软件有很多，用户便会选择那些符合他们的行为习惯、更容易学习和使用、效率更高的软件，这反应了软件的可用性。如在众多音乐播放器中，用户会选择一款最适合的产品，或许是它的歌曲库更全，或许是它提供的交互方式符合用户习惯，亦或它的界面风格更为用户喜欢。

在计算机软件发展的初期，由于软件开发人员稀缺，开发成本高等原因，人们只注重软件的功能性，普通用户需要接受相关的培训才能正确地使用软件。但随着开发技术的发展和软件需求量的增加，人们逐渐认识到，对于软件系统而言，即使它采用了最先进的技术，具有非常强大的功能，但如果它不能符合相应领域的业务流程，用户的使用习惯，则依然不会被用户所认可。正如可用性领域专家 Bevan N 在《Meeting User Needs for Quality》中所述的：一个软件只强调内部功能的强大，而忽略了用户的任



务和需求，那么软件的功能就变得与用户无关。

本章主要介绍了可用性的定义，开发出具有高可用性产品的设计基础，Norman 的执行—评估循环，以及基于任务模型的开发过程。

## 第 1 节 可用性定义

系统设计者的主要目标是设计出有用、好用的交互式系统。“有用”表示了系统的功能性，指该系统必须提供的服务，缺乏这样的服务，用户将不能完成目标。“好用”则表示了系统的可用性，指该系统如何将服务提供给用户，使用户可以方便、有效地完成任务。在移动计算环境中，信息技术已经从单一的桌面设备逐步延伸到各种新型移动设备之中，软件也不再局限于满足专业技术人员的办公需求，而是要满足普通大众生活和娱乐的需求。在此应用背景下，用户一方面将软件作为一种简单实用的工具，帮助他们完成既定的目标；另一方面，希望这个工具易学、易用，从而使他们可以将注意力集中在任务而非工具的使用上。因此，在信息技术与人们生活联系日益密切的今天，软件的可用性已经变得比功能性更为重要。

虽然可用性研究和实践问题越来越受到研究者和从业者的关注，但到目前为止，无论是学术界还是产业界，对于可用性还没有一个统一的定义（表 1-1）。

表 1-1 对于可用性的部分定义

提出者及提出时间	定义
Shackel (1991 年)	The capability in human functional terms to be used easily and effectively by the specified range of users, given specified training and user support, to fulfill the specified range of tasks within the specified range of environmental scenarios
ISO 9241—11 (Guidance on usability, 1998 年)	The extent to which a product can be used by specified users to achieve specified goals with effectiveness (task completion by users), efficiency (task in time) and satisfaction (responded by user in term of experience) in a specified context of use (users, tasks, equipment & environments)

续表

提出者及提出时间	定义
ISO 9126—1 (Quality model, 2000 年)	Quality in use: The capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in specified contexts of use

Shackel 将可用性定义为“按照人的功能特性，特定的用户在特定的情景中完成特定范围的任务的易用性和有效性”。他将可用性分为 4 个因素：有效性（Effectiveness）、可学性（Learnability）、灵活性（Flexibility）、态度（Attitude）。国际标准 ISO 9241 是关于办公室环境下交互式计算机系统的人类工效学国际标准，根据人类工效学和可用性原理，分别对各种硬件交互设备属性和软件用户界面设计问题作了详细的规定和建议，在 ISO 9241 的第十一部分中对可用性给出了详细描述和定义。ISO 9126 软件质量模型是评价软件质量的国际标准，定义了 6 种质量特性，并且描述了软件产品评估过程的模型。其中，第一部分所定义的软件质量特性可用来指定客户及使用者在功能性与非功能性方面的需要。除此之外，还有很多研究者通过实践积累，将抽象的可用性概念具体化为一些指标，为设计者提供参考。

Martijn van Welie 将当前的可用性概念划分为 3 个层次：

(1) 抽象层面。国际标准 ISO 9241 中将可用性定义为“一个产品可以被特定的用户在特定的使用上下文中，有效、高效并且满意地达到特定目标的程度”。其中，有效（Effectiveness）指用户完成特定目标的准确性和完整性；高效（Efficiency）指用户实现特定目标的准确性和完整性所消耗的资源；满意度（Satisfaction）指使用产品的舒适度和可接受程度。该标准根据人类工效学和可用性原理，分别对各种硬件交互设备属性和软件用户界面设计问题作了规定和建议。

(2) 可测量的“使用指标”层面。可用性概念本身是不可直接度量的，但与其相关的一些参数具有可度量的性质。能够被度量的可用性参数

可划分为两个范畴：主观的用户偏爱和客观的执行。前者表示用户对产品的喜爱程度，后者是用户对产品质量的评测。随着可用性在实践过程中的应用和发展，可用性的概念转向更具操作性和更为具体的参数指标。如 Nielsen 在其研究成果中将可用性指标划分为有效性（Efficiency）、可学习性（Learnability）、易记性（Memorability）、错误/安全（Errors/Safety）和满意度（Satisfaction）；Shneiderman 则通过用户使用产品时的执行速度（Speed of performance）、学习时间（Time to learn）、保留时间（Retention over time）、用户出错率（Rate of errors by users）和主观满意度（Subjective satisfaction）对可用性进行了度量。

（3）启发式原则和可用性原则层面。为了引导设计者开发出高可用性的产品，一些有关设计过程的启发式原则和可用性原则被提出。这些原则大多都基于设计经验产生，主要用于指导设计，或者作为评估原型和现有系统的基础。如 Nielsen 提出的 10 个可用性原则：系统状态的可视性，系统应与真实世界相符合，用户的控制权和自主权，一致性和标准化，帮助用户识别、诊断和修复错误，预防错误，依赖识别而非记忆，使用的灵活性和有效性，最小化设计，帮助及文档。

人机交互领域的研究和实践都分享了一个共同的目标——为用户提供一种高度可用性的交互式系统。可用性关注的是用户为了某种目的与系统交互的过程。因此，如图 1-1 所示，要提高可用性，设计者需要理解用户、用户的工作及用户与系统的交互。

（1）理解用户。设计开发系统的目的是给用户提供有用、好用的服务，因此，设计者需要理解用户的能力和限制。在使用软件时，用户潜意识中已经存在一个系统模型，它反映用户根据以往经验认定系统应该提供的服务，或者根据当前界面的内容和操作，推断出下一步应该得到什么结果，即用户的概念模型。当用户的概念模型与系统所呈现的内容、结构不一致时，会导致可用性问题。因此，在设计过程中需要考虑执行任务时促使用户做出某种行为的内在原因，即用户的心理模型，并以此为基础分析

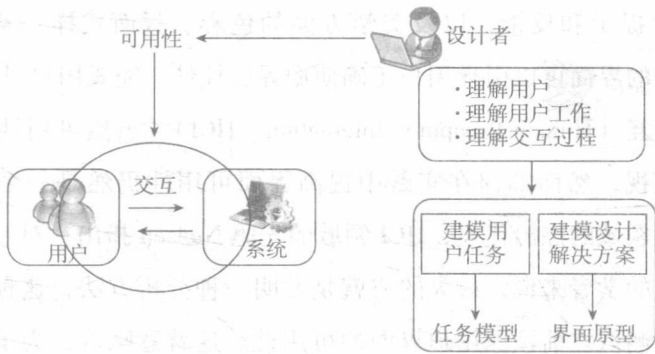


图 1-1 开发高可用系统的设计基础

用户与系统的交互活动，使设计出的系统更加容易被用户接受。

(2) 理解用户的工作。用户使用系统帮助他们完成工作，因而，理解用户要做的工作对开发高可用性系统有至关重要的作用。用户执行任务的目的是完成特定的目标，在设计过程中，设计者希望系统能够最大程度地替代用户执行某些操作，从而简化执行任务的过程，让用户快速完成目标。为此，设计者需要掌握与用户工作相关的内容，如：当前用户是如何进行工作的；用户希望即将生成（改进）的新系统能够帮助他们做什么；用户执行任务的环境；是否有新的、可用的技术可以简化任务结构，减少用户的操作；如何分配任务，使系统非但能被用户控制，而且可以有效、高效地被控制。

(3) 理解用户与系统之间的交互。用户与系统交互，实际上是与系统的一个子集——用户界面进行交互。在交互过程中，用户采用特定的方式操作界面控件，直至系统达到他们所期望的状态。为了使这个交互过程更加有效，设计者需要考虑 3 方面的内容：①交互过程中包含的对象及其关系，即为了完成用户任务需要什么对象，这些对象有什么属性和行为，以及对象之间的关系；②用户与系统的对话，对话是完成任务目标的步骤，正确的对话及对话之间的导航关系可以保证交互逻辑的正确性；③交互对象的视觉表示，视觉表示反映了交互对象在界面上的呈现方式，包括

操作方面的提示和反馈，以及美学方面的色彩、版面式样、风格、布局等，一个好的界面可以引导用户正确使用界面控件，完成用户目标。

人机交互 (Human Computer Interaction, HCI)。虽然可用性在 HCI 领域得到了重视，然而如何在实践中提高界面可用性仍然是一个困难的问题。可用性领域的杰出代表，HCI 领域的先驱 Nielsen 指出：对于许多从事可用性研究的学者来说，最大的成就是发明一种分析方法，这种方法可以让设计师在测试之前预测用户界面的可用性。这就意味着，为了设计出高可用性的系统，设计者必须准确描述两个方面的内容：用户任务和设计解决方案，即任务模型和解决方案模型的构建。前者将用户执行任务过程中所隐含的假设、要求等明确地表达出来，包含了与用户及其工作的各种可用性需求；后者是对基于任务模型生成界面的建模过程，它对于预测设计解决方案的可用性有着重要作用。

## 第 2 节 交互模型

在交互模型中，Norman 提出的执行—评估循环采用用户视角探讨人机界面问题，是目前人机交互领域中最具有影响力的交互模型。如图 1-2 所示，该循环共包括 7 个阶段：

- (1) 建立目标：用户计划完成任务后，外部世界所呈现的理想状态；一般采用用户任务语言来表示。
- (2) 形成操作意图：用户头脑中的目标转化而成的明确的意图。
- (3) 明确动作序列：根据用户意图而指定的计划执行的动作序列。
- (4) 执行动作序列：用户在物理世界中实际执行的动作序列。
- (5) 感知系统状态：动作执行后，用户感知系统所呈现的当前状态。
- (6) 解释系统状态：对感知到的系统状态进行解释，这是一个认知过程。
- (7) 根据目标对系统状态进行评估：用户将感知到的系统状态和头脑中预想的理想状态进行比较。

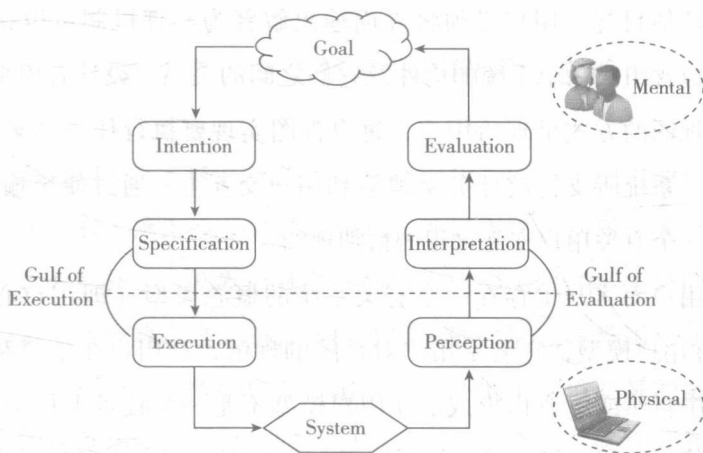


图 1-2 Norman 执行—评估循环

Norman 指出：系统使用物理学属性表示其当前状态，而用户则开始于心理学属性表示的目标。目标和系统状态在内容和形式上均存在很大区别。因此，如果不消除它们之间的隔阂，用户在使用系统时会出现很多可用性问题。这里提出的隔阂，Norman 将其概括为执行隔阂（Gulfs of Execution）和评估隔阂（Gulfs of Evaluation）。

（1）执行隔阂：指用户为了达到目标在头脑中预设的动作序列和系统所允许的动作序列之间的差别。

（2）评估隔阂：指完成任务后系统所呈现的状态和用户头脑中的理想状态之间的差别。

可用性设计的目的就是尽可能减少隔阂。设计者可以通过合适的设计改进界面的输入控制和输出表示，以满足用户的心理学需求。用户也可以通过训练和经验以满足系统的物理需求。

对于设计者而言，通过分析用户需求会产生一个有关系统设计的概念模型（即系统体系结构），称为设计者模型。它能够将模糊的用户任务需求和业务需求概念化、具体化，形成技术上精确的、一致的系统表示。为了达到目标，用户必须将目标转换为用户界面的输入机制可以接受的形

式；为了评估目标，用户必须将界面输出解释为心理机制可以接受的形式：通常包含组成一个系统的构件及它们之间的关系。设计者模型采用一个或多个视图的方式呈现给用户，每个视图实现要执行任务的某个子集。在视图中，系统所支持的任务被映射到用户交互上，通过使用输入/输出机制形成一个有关用户交互过程的精细规约。

对于用户而言，也存在一个有关系统的概念模型（即用户心理学模型），简称用户模型，它表示用户对系统的理解，是用户在学习和使用计算机过程中自然或下意识形成的。用户模型不是一个技术上精确的模型，它会随着使用时间、使用场景的不同而发生改变。用户模型的形成受特定用户在特定场景中完成特定任务的心理学行为模型的影响。在与系统交互的过程中，用户必须具有某种心理模型，以此规划和执行与系统交互的动作，预测和评估系统的行为，并且减少其中所包括的认知努力。

在以前的软件系统中，界面是领域内容的直接表示，设计人员预先定义交互过程和交互风格，旨在让用户能够正确理解和使用交互构建，其反映了设计者头脑中的系统模型；而在移动环境中，对于面向大众生活娱乐的系统，用户不能理解也不愿意理解界面上呈现的那些晦涩的领域概念，只想将软件作为一种简单实用的工具，帮助他们完成既定的目标。用户在使用软件时，潜意识中已经存在一个系统模型，它反映了用户根据以往的经验认定系统应该提供的服务，以及根据当前界面的内容和操作，推断出下一步应该得到什么结果，即用户的概念模型。对于一个高可用性的软件系统，设计者所设计的界面模型应该与用户的概念模型相一致。然而在移动环境中，用户头脑中潜在的那个系统模型是不精确的，具有二义性的，受用户经验和使用环境的影响。如何将用户头脑中的意图翻译成计算机中的对象和它们的操作，需要设计者深入分析用户任务，理解用户在实际环境中为了完成目标执行任务的过程，并对任务分析的数据进行结构化、层次化的组织，最终建立系统设计过程中的任务模型。

### 第3节 基于任务模型的开发过程

移动环境的交互系统具有普适化和个性化的特点，用户的任务相对于传统环境也具有不确定性、多样性和易受外界干扰等性质。准确地分析动态环境中用户的任务，开发出具有好的用户体验的软件系统目前已成为人机交互领域的研究热点之一。怎样更加准确地去描述各类用户在不同使用情境中的交互过程？系统如何去自适应动态环境中环境变量的影响？面对各种不同的设备和开发平台，如何能够简化开发方法，降低开发成本，实现界面原型的自动（半自动）生成？针对这些问题，基于任务分析和模型转换的方法提供了一个可行的解（图1-3）。

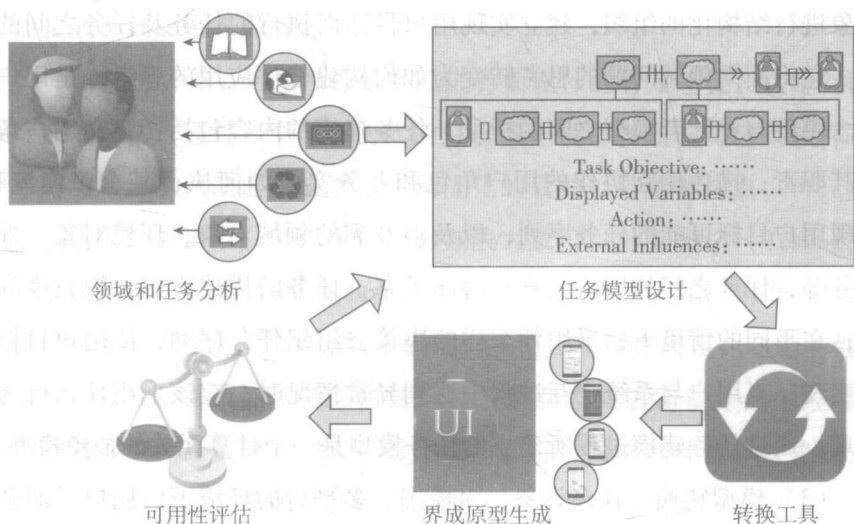


图1-3 基于任务模型的开发过程

(1) 领域和任务分析。通过访谈、观察、问卷等方法从用户真实工作的环境中了解用户执行业务或者相关活动的细节，并采用场景或故事描述的方式记录相关过程。设计者和开发者通过对问题领域的分析，发掘重要的业务领域概念，并建立业务领域概念之间的关系；使设计者和最终



用户对领域中的核心概念、工作流程和相关规则的认识达成一致。通过分析过程生成的任务描述是系统设计者和开发者与用户交流的有力工具。在领域和任务分析过程中，通过用户角色的划分，分析业务对象和业务实体，并根据内部规则将他们有机地联系在一起。在移动环境中，交互式系统在帮助用户有效、高效完成任务的同时，更加注重用户的体验，所以在分析过程中，除了功能性需求所涉及的领域概念、领域规则外，针对移动交互式系统个性化和动态性的特点，还需要分析用户的特性及与用户任务相关的环境信息，将传统静态环境中的高度统一性需求转化为动态环境中特定用户类型在特定的使用上下文中的特殊需求，使得交互式系统更加符合用户习惯，易于操作使用。

(2) 任务建模。通过领域和任务分析过程，将与用户任务相关的领域对象进行结构化的组织，建立实现用户目标而执行的任务及任务之间的关系。它将面向领域问题的思考转变为如何构建软件应用的思考，是用户的概念模型与设计者模型之间的桥梁。任务建模的内容包括：定义用户角色及其职责；描述相互协作的用户角色和业务实体如何执行某个工作流程；实现用户目标可能的任务序列，以及涉及到的领域对象、环境对象、外部服务等；任务之间的层次关系和时序关系，任务的描述方式，颗粒度的定义；在不同的情境下，采用什么样的决策去组织任务序列，使用户目标能够实现；当用户与系统交互过程中遇到异常情况时，应该采用什么行为去处理。通过任务建模过程所建立的任务模型是一个计算无关的抽象模型。

(3) 模型转换。在多设备、多模态、多平台的环境下，用户界面开发工作的复杂程度大大提高，目前的一种研究趋势是采用模型转换的技术支持用户界面设计和构造过程的自动或半自动化。基于模型的开发环境提供开发工具，使设计者能够建立并对高层抽象模型逐步求精，以生成具体的用户界面原型。从抽象模型空间至具体模型空间的映射存在多种映射形式和转换种类，如：任务活动单元映射为对话执行顺序；任务执行条件映射为对话的执行似能状态；任务的输入/输出需求映射为对话的输入/输出需