信息系统开发中领域、技术和客户知识的学习效应: 一项实证研究



**摘要**：本研究检验了信息系统开发（ISD）中的学习效应(即先验经验的影响)。ISD 的固有特点是跨项目的不同任务、团队和项目复杂性的级别。这些特征挑战了我们对学习效应如何在 ISD 环境中发生的理解。本研究以转移学习理论为基础，探讨项目团队如何学习，以及在什么条件下学习效应更强或更弱。我们发现，当先前的项目和当前的项目共享相同的领域、技术或客户知识元素——领域、技术和客户是 ISD 最基本的知识类型时，ISD 项目团队在先前项目中的经验转化为当前 ISD 项目的绩效提升。此外，我们发现领域、技术和客户知识的学习效应是相互替代的，这些学习效应是强还是弱取决于 ISD 项目的团队和任务的复杂程度。本研究对学习效应、领域、技术、客户知识和项目复杂性的作用以及一般的组织学习文献做出了重要贡献。它还为 ISD 组织的项目人员配备和知识获取等实际问题提供了重要的管理见解。

**1 引言**

尽管在编程语言、开发方法和工具以及计算机科学和信息系统的正规培训方面取得了数十年的进步，但令人遗憾的是，信息系统开发（ISD）仍然受到频繁的成本和/或日程超支等性能问题的困扰(brooks1995，ruben stein 2007)。虽然成功率已逐步提高，但大多数ISD项目在成功交付预期系统方面面临挑战(ewusi-mensah2003、 elemam 和 koru2008)。

虽然可能有许多不同的因素导致 ISD 的低成功率，但通过识别与其增加（或减少）相关的因素来直接调查 ISD 绩效已成为最近信息系统文献中备受关注的焦点。 最值得注意的是，许多研究人员开始采用基于知识的学习观点，提出可以通过利用学习曲线（或经验曲线）效应来提高 ISD 绩效（例如，Boh 等人 2007，Narayanan 等人 2009）。

学习曲线效应(或简单地说，学习效应)背后的基本理念相当简单明了——重复的经验(或试验)创造出不断增长的知识储备，从而提高绩效(argo 2013)。根据学习迁移理论Theory of Transfer of Learning (ellis 1965，thorndike and woodworth1901，perkins and salomon1994，schunk2015) ，学习曲线的主要内涵是先前经验和当前任务(即跨试验)之间存在相同的元素。在这方面，学习效应通常是在交付的产品(或服务)是相同的情况下进行检查，并且在产品(或服务)级别定义相同经验(即相同的重复任务)的单元。软件工程和信息系统上下文中的大多数学习曲线研究(例如，boh 等人2007，narayanan 等人2009)也遵循类似的惯例。他们在信息系统维护(ISM)环境中衡量学习效应，在这种环境中，同一个系统被维护，并使用系统或子系统来表示相同的经验(例如，在同一个系统或子系统上工作的先前经验)。

虽然这种选择相同经验的单元适合 ISM 环境，但在 ISD 环境中可能不是这种情况，其中软件服务公司执行不同客户要求的各种 ISD 项目。尽管 ISD 与ISM有许多共性，但也包含了几个必然会影响学习效应的独特特征。 文献回顾强调了以下作为 ISD（与 ISM 相比）的独特特征：不同的任务（即，每个 ISD 项目开发不同的信息系统）（Huckman 等，2009）、不同的团队（即每个 ISD 项目包括一组不同的团队成员）（Huckman 和 Staats2011）和不同的复杂性（即，每个 ISD 项目都包含一个不同程度的复杂性）（Xia and Lee2005）。 在研究 ISD 环境中的学习效应时，这些是需要考虑的重要和突出的因素。

如果有效的学习迁移能够在同一个任务和同一个学习者之间发生(ellis1965，thorndike and woodworth1901，perkins and salomon 1994) ，那么上述维度中 ISD 的不同性质给学习迁移带来了挑战。这是否意味着学习效应不会存在于 ISD 的环境中？我们认为答案是“不”这个猜想是基于这样的观察，当我们仔细观察在 ISD 中开发的系统时，它们并不是完全不同的，而是在项目中共享共同的元素。这表明学习效应也应该存在于 ISD 中。问题在于学习效应是如何(即以何种形式)存在的。

Huckman 及其同事 (2009) 提供的实证结果为进一步研究该问题提供了良好的起点。 他们发现，在 ISD 中，当项目成员角色（即在先前项目中担任相同角色的经验，如项目经理或非经理）被用作相同经验的单元时，可以观察到学习效应。 事实上，角色经验是将 ISD 中重复经验概念化的一种重要而新颖的方式。 然而，鉴于 ISD 项目的多维性质，需要进一步的研究来表征其他维度中的相同经验并发现相关的学习效应，以扩展我们对 ISD 中学习如何发生的理解。

在这项研究中，我们建议使用所需的知识来表征相同的经验是了解 ISD 学习效应的一种有用且直观的方式。 ISD 是知识密集型工作，需要应用各种知识领域（Langer 等人 2014、Lee 等人 1995、Jiang 等人 1998）。 特别是，ISD 文献发现领域、技术和客户是最重要的知识领域（Gopal 和 Gosain2010、Khatri 等人 2006、Huckman 和 Staats2011、Ethiraj 等人 2005 年、Garud 1997 年）。 我们将 ISD 概念化为领域、技术和客户知识的组合应用，并认为不同的项目可能彼此共享共同的领域、技术和/或客户。 因此，我们的第一个研究问题提出以下问题：ISD 项目团队是否从领域、技术和客户方面的先前经验中受益，以提高项目绩效？

有趣的后续问题是这三类知识中的经验是否以及如何相互关联: 它们是相互干扰还是相互替代，以至于它们对表现的综合影响减弱了(即，负相互作用效应) ？随着经验/知识/知识水平的提高(例如，一种正面的相互作用效应) ，他们是否会相互补充以产生额外的协同效应？它们是否相互独立，以至于它们的影响仅仅是影响 ISD 性能的附加物？这些后续问题在理论上是重要的，因为现存的理论认为，一种类型的知识可能会干扰(underwood1957，levinthal 和 march1993)或替代(argo and ingram2000，walsh 和 ungson 1991)另一种类型的知识积累，或者一种类型的知识可能相反地促进另一种类型的获得(cohen 和 levinthal1990)。因此，我们的第二个研究问题提出了以下问题: 领域、技术和客户知识的学习效应是替代的、互补的还是累加的？

最后，众所周知，学习受到学习环境的影响(argo 2013)。在这项研究中，我们借鉴了相关的 ISD 文献(xia and lee2005，langer et al. 2014) ，并认为 ISD 项目复杂性是影响项目绩效的最重要的关联因素之一(xia and lee2005)。然而，目前还不清楚项目复杂性是否以及如何替代、补充领域、技术和客户体验的影响。因此，我们的第三个研究问题提出了以下问题: 项目复杂性如何调节这三种类型的经验的学习效应？

我们根据学习迁移理论(Ellis 1965, Royer1979, Day and Goldstone2012)回答了上述三个研究问题。利用来自一家著名的国际信息技术(IT)服务公司的ISD项目的大量档案数据集，我们观察了项目团队水平的领域、技术和客户知识的学习效应。研究还发现，领域、技术和客户知识的学习效应是相互替代的，并受到ISD项目复杂性的负向调节。

**2 理论与假设**

**2.1 组织学习曲线与学习迁移理论**

重复的经验可以提高组织的绩效。这种被广泛记录的现象通常被称为组织学习曲线效应或简单学习效应(Argote2013)。学习效应的理论基础可以追溯到学习迁移理论(Ellis1965, Schunk 2015)，该理论认为，任务执行的试验次数越多，整体表现就越好，因为当前执行的表现(即目标任务)体现了从以前的试验(即源任务)转移过来的学习。关键在于从源任务中获取的知识能否有效地转移到目标任务中。

从两种理论视角来解释学习迁移的机制。第一种是学习迁移的环境理论，它侧重于学习的环境因素，即重复任务的特征。例如，相同与不同(Thorndike和Woodworth1901)，近与远(Perkins和Salomon1994)，横向与纵向(Gagné1965)，特异性与非特异性(Ellis1965)，文字与图形(Royer1979)，并认为迁移的发生主要是因为重复任务的属性。其中最具代表性的理论是相同元素理论(Thorndike和Woodworth1901)，它表明只有当任务共享相同元素时才会在任务之间发生转移。Perkins和Salomon(1994)通过区分源任务和目标任务之间有显著重叠的“近任务”和无重叠的“远任务”，扩展了这一论点。迁移程度，即学习效应的强弱，与任务之间的重叠程度呈正相关。因此，如果任务之间有足够的重叠，任务性能不仅会随着特定任务的累积经验的增加而提高，而且会随着不同任务的经验的增加而提高。

第二种观点是学习迁移的认知理论(Royer1979, Day and Goldstone2012)，该理论关注的是学习者的认知过程，而不是任务。该观点认为，学习迁移的可能性取决于学习者检索存储在记忆中的相关先前经验的能力。这意味着，当学习者能够捕捉到过去经验和当前任务之间的联系时，学习迁移更有可能发生。相反，即使源任务和目标任务之间存在相似性，如果这些相似性在认知上难以识别，学习迁移也不太可能发生，最终的学习效应也会减弱。综上所述，重复任务之间的关系和学习者的认知过程都会影响学习迁移的程度。当过去的经验和当前的任务有更多相同的元素时，当过去的经验和当前的任务之间的相似性在认知上更容易识别时，学习效应就可能发生。

**2.2 ISD项目团队中的学习效应**

ISD情境可以被定义为“按订单制造”(或“按订单组装”)的生产过程，其中重复生产的最终产品或服务并不完全相同，而是由可自由使用的元素组装而成。每个信息系统交付的信息都不独特，但它们往往与之前开发的其他系统共享共同的业务规则和技术组件。在这种情况下，相同经验的单元应该在可重用元素的层面上，而不是最终产品，以便适当地反映学习效应(Kantor和Zangwill1991)，因为在这里，任务重复和学习迁移的基本单元是可重用元素(Ellis1965)。关键的问题是，什么是对重复元素的有用的概念化和分类?我们建议在ISD中使用的知识类型可以作为一种有用的方法来捕捉学习效应。我们将ISD概念化为**领域**、**技术**和**客户**知识的各种元素的应用，并提出相同经验的单元应该是这些知识元素。

**领域知识**指的是关于正在构建的信息系统的应用领域和系统将被使用的上下文的知识(Gopal和Gosain2010, Khatri等。2006)。例如，为制造公司开发一套生产计划系统，ISD项目小组必须了解基本的生产计划概念，例如物料清单、产能、安全库存、生产交货期等；和这些概念如何适应不同类型的生产计划,如总计划、主生产计划、物料需求计划等。**技术知识**指的是关于技术构建块以及如何在信息系统的实现中使用它们的知识，(Gopal和Gosain2010, Khatri等人，2006)。这包括与各种硬件和软件技术、它们如何工作以及如何集成它们相关的知识。技术知识的例子包括操作系统(例如，Windows, Linux)，编程语言(例如，Java, COBOL)，数据库技术(例如，ISAM，关系数据库，面向对象数据库)，网络技术(例如，局域网，防火墙，消息传递协议)等。**客户知识**是指设定目标信息系统的要求并使用该系统的组织及其用户的知识(Ethiraj et al.2005, Garud1997, Huckman and Staats2011)。每个客户都是不同的。每个公司都可能有独特的文化、组织结构、操作程序、决策过程、技术环境、IT能力、利益相关者等。对于ISD项目团队来说，正确理解独特的客户特征是非常重要的，因为ISD项目的成功往往受到它们的影响(Ethiraj等人，2005年，Garud1997年，Huckman和Staats 2011年)。领域、技术和客户知识对ISD来说是必不可少的，因为每一个ISD项目都需要这三种类型的知识，尽管这些类型中的特定元素在各个项目中可能不相同。

考虑以下程式化的例子：项目P1是一个使用Java编程语言(技术)为HomeApplianceCo(客户)开发的电子商务应用程序(领域)；项目P2也是一个电子商务应用程序(领域)，但使用ASP .NET编程语言(技术)为PharmaCo(客户)开发；最后项目P3是一个知识管理应用程序(领域)，使用C编程语言(技术)为咨询公司(客户)开发。严格地说，这三个项目是不同的，但是项目P1和P2共享来自同一个领域(即电子商务)的知识。虽然最终的系统并不相同，但是先前在项目P1中的经验可能对项目P2有一定的好处，因为这些项目共享共同的领域知识元素。相反，项目P3可能不会从先前的项目P1或P2经验中受益，因为它不与这些项目共享任何公共领域、技术或客户知识元素。

根据环境观点（Perkins 和 Salomon 1994、Thorndike 和 Woodworth 1901），学习效应的产生是跨任务重复执行相同元素的结果。 因此，一个ISD项目团队的先前经验，包括领域经验（即具有相同领域知识元素的经验）、技术经验（即具有相同技术知识元素的经验）和客户体验（即具有相同客户的经验） ，应该会导致学习效应（即性能提升）。基于上述讨论，我们提出以下基线假设：

H1A. ISD项目团队的领域经验对项目团队的绩效有积极的影响。

H1B. ISD项目团队的技术经验对项目团队的绩效有积极的影响。

H1C. ISD项目团队的客户经验对项目团队的绩效有积极的影响。

**2.3 领域、技术和客户经验之间的交互**

现在我们考虑领域、技术和客户经验如何相互作用并影响它们的整体学习效应。我们预计领域和技术经验将具有替代性交互作用。从问题解决的角度来看，领域知识为 ISD 项目团队提供了与问题空间相关的信息，而技术知识提供了与解决方案空间相关的信息（Adelson 和 Soloway1985，Khatri 等人 2006）。问题和解决方案空间可能并不总是完全独立的，而是相互关联的。在 ISD 中，领域和技术可以为 ISD 项目团队提供类似的知识。例如，电子商务应用程序通常是使用 Web 技术（例如 ASP .NET、JSP）构建的，其中电子商务应用程序的某些域功能（例如，用户身份验证、用户会话管理）受到以下特性的约束或表征使用的网络技术（例如，基于表单的身份验证、无记忆会话）。因此，使用 Web 技术可以提供有关电子商务领域要求的宝贵知识，这些知识也可以从电子商务应用程序中获得。同样，从事电子商务应用程序可以为开发人员提供与 Web 技术相关的知识，这些知识也可以从使用 Web 技术中获得。

我们预计客户经验和领域或技术经验也会产生替代性交互传递效应。有两个原因。首先，虽然它们作为一个整体并不完全相同，但两者都有一些共同的元素，这些元素为后续的 ISD 项目提供了相同的转移效应。客户经验为项目团队提供有关将引入目标信息系统的客户整体功能和技术背景的信息，包括客户的业务背景、操作程序、组织结构、文化、技术能力、技术架构、遗留系统和用户(Ethiraj et al.2005, Garud1997)。因此，尽管这些可能不会直接提供目标系统的功能和技术要求，但客户经验可能会提供有关领域（例如，业务上下文、操作程序等）和技术（例如，技术能力、架构、遗留问题等）的上下文信息，项目团队也可以使用过去在同一领域和/或技术中的经验来收集和理解这些信息。换句话说，如果一个项目团队有客户经验，那么该团队就已经有一定程度的能力来收集和理解有关目标系统的功能和技术要求以及约束的信息，因为客户经验部分地替代了领域和技术经验的作用。这意味着，在其他条件相同的情况下，具有客户经验的项目团队可能不需要与其他没有客户经验的团队具有相同级别的领域和技术经验来产生相同水平的性能。

其次，客户经验提高了ISD项目团队的沟通效率和有效性，这也有助于降低领域和技术经验的重要性。在ISD中，客户通常提供功能和技术上的需求及限制，政府新闻处计划小组收集及解释这些需求及限制，以便将它们转换为信息系统。因此，项目团队必须有能力与客户进行有效的沟通，从而引出需求并正确地理解它们(Davis1982)。虽然我们已经讨论了客户经验在提供信息方面的作用，类似于由领域和技术经验提供的信息，但客户经验也通过在供应商和客户之间培养信任，为ISD项目团队提供额外的好处。使双方能够建立有效和高效的沟通渠道(Ring和van de Ven1994)。这些渠道增强了项目团队准确收集客户需求的能力(Clark et al.2013)。我们假设如下：

H2A. ISD项目团队的领域经验和技术经验负向交互作用(替代交互作用)，当项目团队的领域经验越大，技术经验对项目团队绩效的正向作用越弱。

H2B. ISD项目团队的领域经验和客户经验负向交互作用(替代交互作用)，当项目团队拥有更多的客户经验时，领域经验对项目团队绩效的积极影响变得较弱。

H2C. ISD项目团队的技术经验与客户经验负向互动(替代交互作用)，当项目团队拥有更多的客户经验时，技术经验对项目团队绩效的积极影响变得较弱。

**2.4 ISD项目复杂度的调节作用**

复杂性被认为是影响ISD中任务执行者认知负荷和过程的重要因素(Wood1986, Xia and Lee2005)。我们将ISD上下文描述为涉及不同的任务(不同的领域、技术和客户)和跨ISD项目的团队(不同的项目团队成员)。任务和团队的具体结构将影响ISD项目的复杂性水平(Xia and Lee2005, Langer et al.2014)。任务复杂性通常被定义为将任务输入转换为输出的难度，因为任务元素之间存在相互依赖或多样性(Espinosaetal。而团队复杂性通常被定义为团队内部协调和沟通的难度，因为团队的规模和团队成员之间的相互依赖(Xia and Lee2005, Huckman et al.2009)。任务和团队的复杂性都已被证明会妨碍ISD项目的绩效(Brooks1995, Xia and Lee2005, Wood1986)。除了上述两种类型的ISD项目复杂性对ISD绩效的直接影响外，预计这两种类型的复杂性也会影响学习迁移的认知过程。

首先，任务复杂性会增加学习者需要处理的信息量，从而阻碍学习者的理解过程。随着任务的复杂性增加，ISD项目小组的成员更难理解当前任务的性质和本质，因此，他们更难找到以往经验与当前任务之间的联系。这进一步降低了项目团队成员检索相关经验的可能性，导致以往经验的转移更加困难。ISD本质上是一项知识型任务；因此，任务复杂性受到知识复杂性的影响。很明显，处理复杂知识的任务比处理不那么复杂知识的任务更难执行。由于每种知识类型都有其自身的学习效应，因此预计知识复杂性会影响其类型的学习效应。因此，我们假设如下：

H3A. ISD项目团队的技术经验与ISD项目的技术复杂性之间存在负向交互作用(抵消交互作用)，当项目的技术复杂性较高时，技术经验对项目团队绩效的积极影响就会减弱。

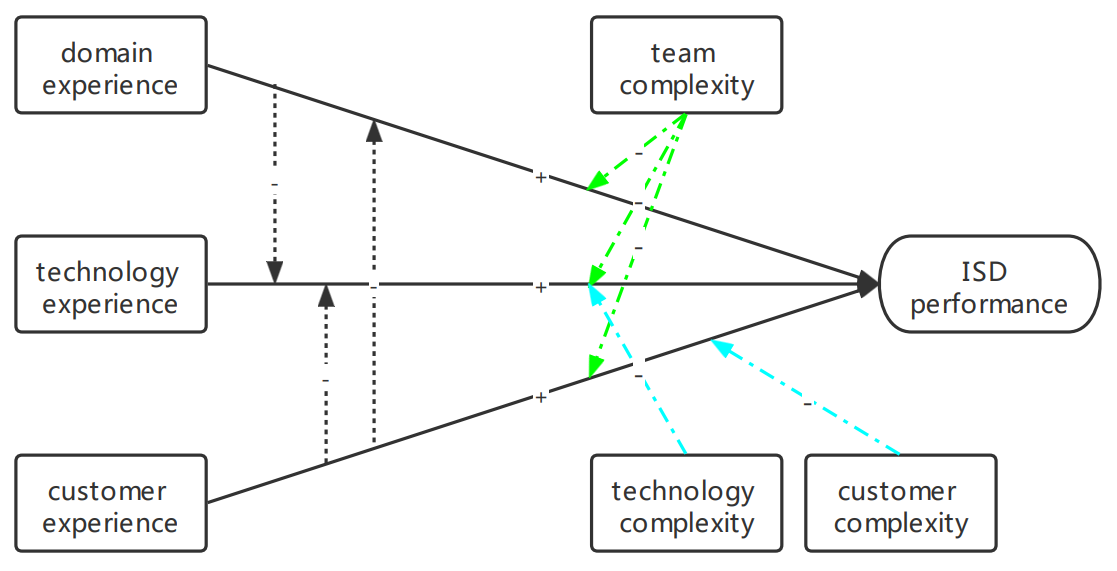
H3B. ISD项目团队的客户经验和ISD项目的客户复杂性负向交互(抵消交互作用)，当项目具有更高的客户复杂性时，客户体验对项目团队绩效的积极影响变得较弱。

其次，ISD是一个团队任务，依赖于团队成员之间的有效协调(Faraj和Sproull 2000)。团队的复杂性通过增加需要处理的信息量来降低学习者的理解和协调过程的表现。随着团队复杂性的增加，对于ISD项目团队来说，从单个团队成员中共同检索过去的相关经验，并协调个人过去的经验以有效地转移到当前的任务中变得更加困难。这最终会降低团队成员将他们之前的经验转移到当前项目中的可能性。因此，我们假设如下：

H4A. ISD项目团队的领域经验和项目团队的复杂性负向交互作用(抵消交互作用)，因此当项目具有更高的团队复杂性时，领域经验对项目团队绩效的积极影响会变弱。

H4B. ISD项目团队的技术经验和项目团队的复杂性负向交互作用(抵消交互作用)，当项目团队复杂性较高时，技术经验对项目团队绩效的积极影响就会减弱。

H4C. ISD项目团队的客户经验与项目团队复杂性负向交互(抵消交互作用)，当项目团队复杂性较高时，客户经验对项目团队绩效的积极影响变得较弱。



**3 研究方法**

**3.1 研究设置和数据收集**

我们的研究假设使用从一家著名的全球 IT 服务公司收集的档案数据进行了测试。 该公司于 2004 年获得了能力成熟度模型集成 (CMMI) 级别 5，为广泛行业的客户提供基于合同的定制 ISD 服务。 我们的数据集包含公司完成的所有 ISD 项目的详细信息，可追溯到公司 1980 年代后期成立，包括 (1) 项目所需的领域、技术和客户知识； (2) 参与项目的员工； (3) 员工在领域、技术和客户知识方面的先前经验。 我们的分析样本包括 2005 年至 2007 年间结束的 497个ISD 项目，涉及 2,393 名不重复员工。 大多数项目为大型开发项目，平均项目团队规模约为 9.2 人，平均持续时间为 10.1个月。

领域、技术和客户知识的不同方面使用公司通过结合行业中使用的专业标准(如IEEE软件工程知识体指南(Bourque et al.1999))开发的专有分类法进行分类。IEEE软件工程标准分类标准(1002 - 1987)(Tripp和Fendrich1987)， IEEE软件工程术语术语表标准(610.12-1990)(Radatz等人1990)，Gartner的信息系统应用和市场分类以及标准行业分类，以及来自内部和外部专家的意见。分类法中最详细层次上的元素，我们在本文中将其称为知识类别，用于描述ISD项目的知识需求，以及对员工的经验进行编目。领域知识类别的例子包括银行、保险、电子政府、军事、电子商务、人力资源、制造执行、产品生命周期管理和健康信息系统。技术知识类别的例子包括特定的编程语言(例如，Java, C, HTML, COBOL等)，建模技术(例如。例如，实体关系图、数据流图、用例图等)、数据库系统(如Oracle、Microsoft Access、DB2等)、操作系统(如Windows、HP-Unix等)和软件开发工具(如Eclipse、Visual Studio等)。客户简单地按公司名称分类(例如，三星电子等)。随着技术、应用领域和客户的行业和名称的发展，公司定期更新其分类。当分类法更新时，项目知识需求数据和员工经验记录也会更新，以便维护分类法和操作记录之间的数据完整性。

**3.2 测量**

**3.2.1 因变量：开发努力**

与之前在软件工程和信息系统中的大多数学习研究一致(例如，Boh et al.2007, Espinosa et al.2007, Narayanan et al.2009)，我们将开发努力作为主要的因变量，因为与其他性能度量相比，它是客观的，更适合于捕获学习效应。我们将努力(努力)运作化为项目完成所花费的实际劳动时间。就这一衡量标准而言，由于一个项目小组在完成ISD项目时所作的工作较少，则被视为取得较佳的表现，因此数目越少则表现越好。

开发努力与ISD项目的质量之间存在着权衡，即较少的努力可以以较低的质量为代价来完成项目(例如，较少的工作却有许多缺陷)，反之亦然。因此，需要对努力测量的质量进行控制，以更准确地反映绩效(Espinosa等。2007)。我们所研究的公司严格控制ISD项目的质量——每个ISD项目都必须通过一个由独立测试小组进行的广泛而彻底的最终发布测试，才被视为完成。因此，我们的因变量是无误差的(即，没有任何交付缺陷)，类似于Espinosa等人(2007)所使用的因变量。它衡量ISD项目的全部开发工作，直到项目将承诺的系统毫无错误地交付给客户。与之前的研究相似(Espinosa et al.2007, Narayanan et al.2009)，我们的努力数据也是倾斜的；因此，我们对它进行log。

**3.2.2 自变量：领域、技术和客户经验**

先前在所需领域()、技术()和客户()知识类别中的经验用项目团队成员在领域、技术和客户中使用这些知识类别的先前项目的总数来衡量。例如，如果一个团队成员以前使用过Java，并在一个项目中为A银行开发了一个零售电子银行应用程序，那么这个员工就被认为在零售电子银行方面积累了一个单位的经验(作为领域知识)，在Java方面积累了一个单位的经验(作为技术知识)，A银行一单位工作经验(作为客户知识)。

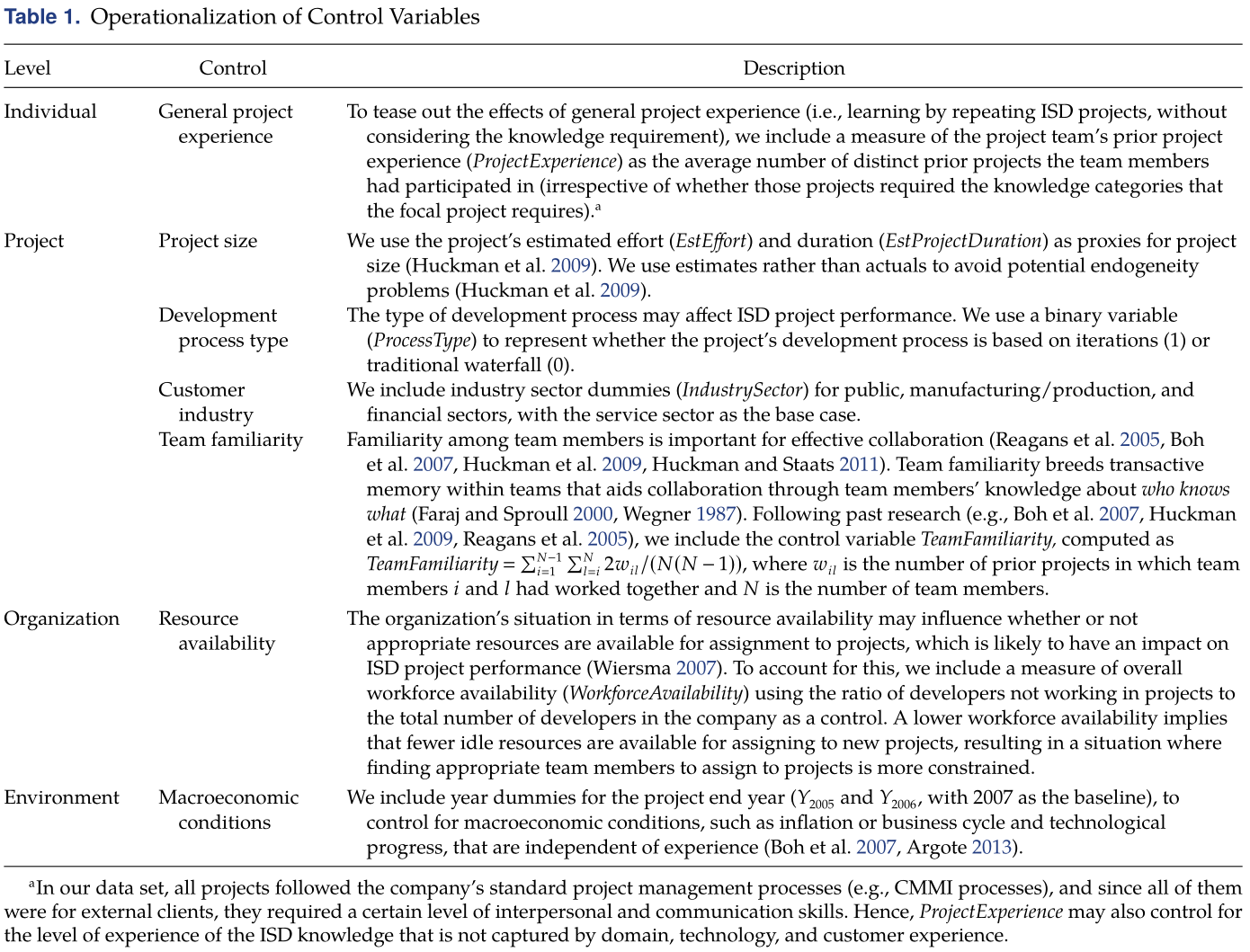
为了在项目团队级别上衡量经验，我们使用跨知识需求的平均值，该平均值由项目团队中开发人员的数量规范化。更正式地说，项目团队对于每个ISD知识类型的累积经验计算为。其中，；是当前项目团队中开发人员的数量；是对于给定的知识类型，当前项目所需的知识类别的数量；表示第个所需知识类别；而表示成员在知识类别上先前项目的不同数量。

**3.2.3 调节变量**

根据相关文献(Xia and Lee2005, Espinosa et al. 2007, Langer et al.2014)，我们使用项目团队成员数量表示团队复杂性()，项目所需的不同技术知识类别数量表示技术复杂性()。对于客户复杂性()，我们使用一个指示变量来表示客户是否(0)或(1)在IT服务公司所属的同一公司集团内。该公司是一个全球性集团的一部分，该集团由许多不同行业的不同公司组成。虽然集团中的每一家公司都是独立的，在不同的行业和环境中运行，但它们有许多共同之处，包括标准的通信系统、组织文化和业务流程。由于共享的通信系统、文化和流程，在集团内部与客户合作可能比与非集团内部的客户合作要简单。

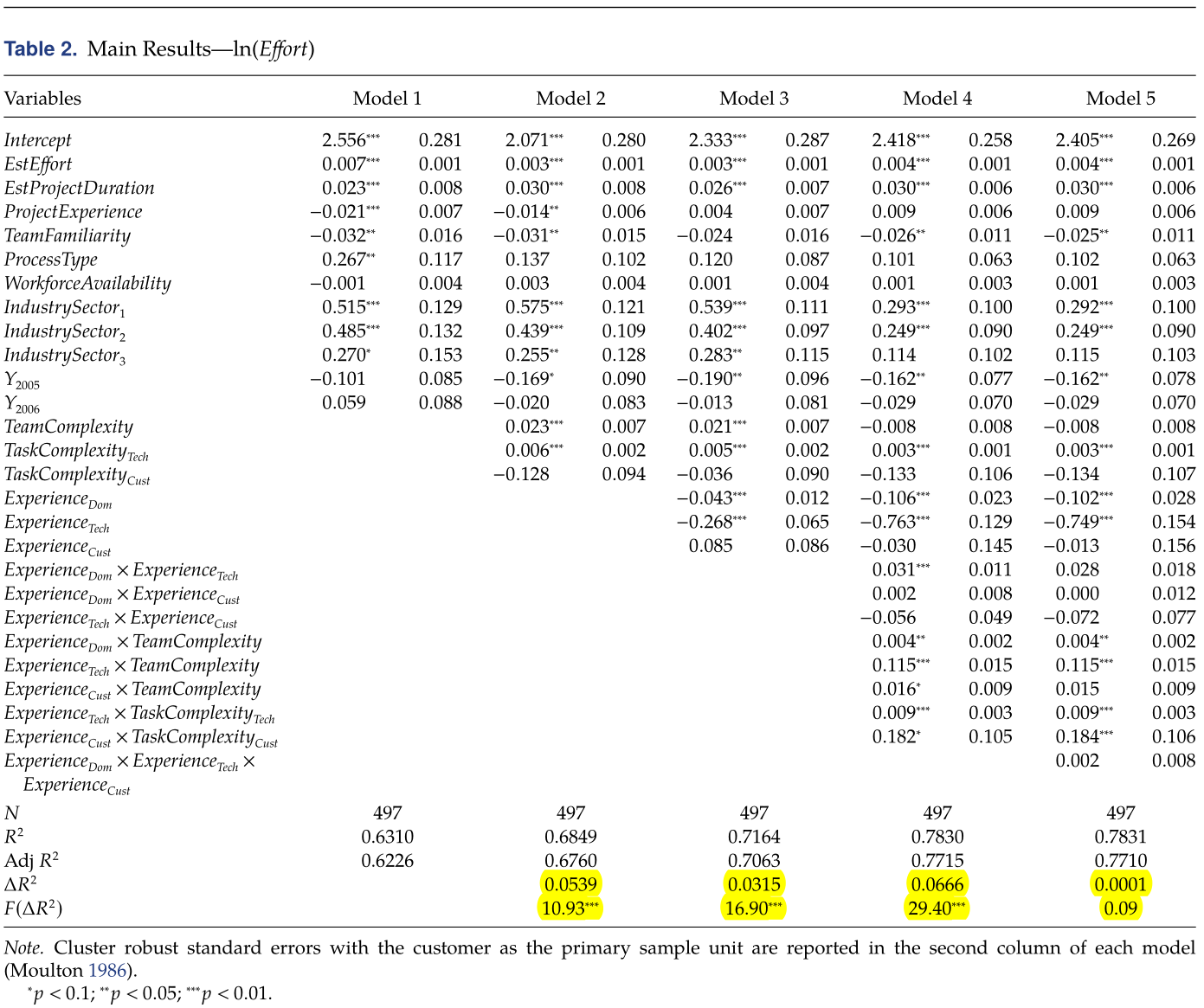
**3.2.4 控制变量**

ISD项目绩效可能受到与团队成员、项目、当前组织和外部环境相关的其他因素的影响。因此，我们的控制包括一般项目经验(团队成员级别)；项目规模、开发过程类型、客户行业和团队熟悉程度(项目级别)；资源可用性(组织级别)；宏观经济条件(外部环境)。有关这些操作化的详细信息，请参见表1。



**4 结果**

按照分析具有交互效应的模型的标准实践(Aiken等人，1991年)，我们分层次估计我们的普通最小二乘(OLS)线性回归模型(见表2)——一个仅包括控制变量的开发努力的基线模型(模型1)；加入调节变量(模型2)；添加领域、技术和客户体验自变量(模型3)；将所有双向交互变量相加(模型4)；最后，添加领域、技术和客户体验三者之间的交互变量(模型5)。这种分层的方法可以让我们检验这些变量的加入是否增加了模型的解释能力。结果表明，调节变量对模型1具有显著的解释能力()，这意味着调节变量在模型3中也需要作为控制变量。领域、技术和客户经验显著增加了解释能力()。双向交互项也显著增加了模型的解释能力()。然而，领域、技术和客户经验之间的三方交互项并没有增加解释力()。



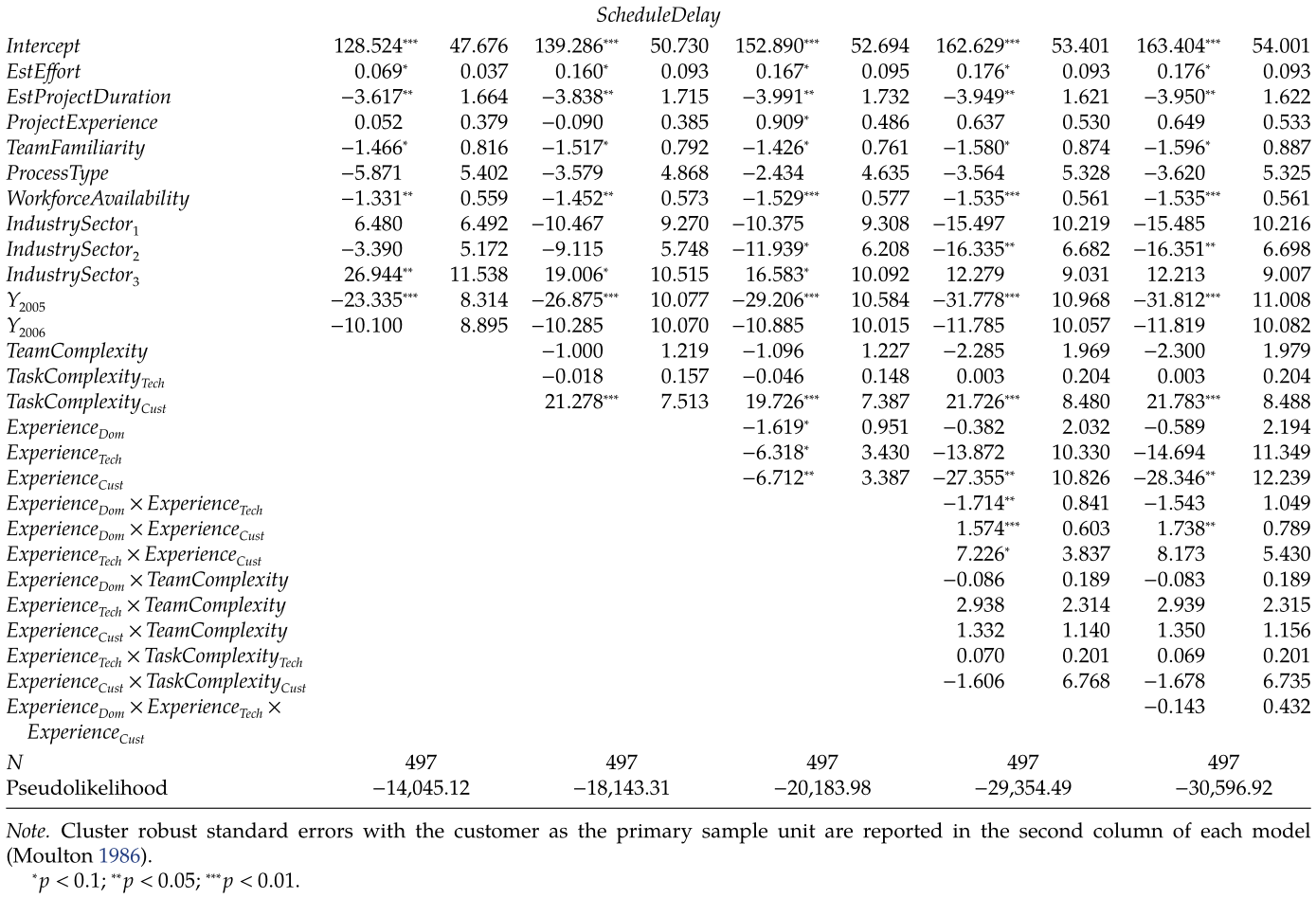
**4.1 主要结果**

模型3的结果表明，领域和技术经验具有显著的学习效应，而客户经验没有。因此，H1A和H1B成立，而H1C不成立。模型4用于检验假设2-4。H2认为领域、技术和客户经验是彼此的替代品，因此当ISD项目小组先前有一个维度的经验时，与不熟悉该维度的小组相比，其他两个维度的经验对ISD项目的益处会更少。支持H2A，不支持H2B和H2C。H3认为，在任务较复杂的ISD项目下，ISD项目小组在识别和检索相关的过往经验方面会有更大的困难，因此技术(H3A)和客户(H3B)经验对ISD项目表现的积极影响将会减弱。H3A和H3B都成立。H4假设在团队层面ISD项目较为复杂的情况下，ISD项目团队将更难协调团队成员先前经验的转移，从而削弱团队成员的领域(H4A)、技术(H4B)和客户(H4C)经验对ISD项目绩效的积极影响。所有三个假设(H4A-H4C)都得到了支持。

**4.2 补充分析：进度延迟**

软件工程和信息系统文献不断重申正确评估ISD项目绩效的挑战(例如，Banker和Kemerer1989)，因为ISD项目绩效有多个维度，它们可能并不总是以同步的方式工作(例如，低进度延迟但高努力，等等)。据观察，开发努力是在软件开发中捕捉学习效应的最合适的度量方法(Boh et al.2007, Espinosa et al.2007, Narayanan et al.2009)；这就是为什么我们在第4.1节的主要分析中使用它作为因变量。话虽如此，其他业绩指标仍可能提供额外的细致入微的见解。遵守计划的时间表就是这样一种可选择的性能维度，在之前的学习研究中已经使用过(Huckman等人，2009年，Huckman和Staats, 2011年)。因此，作为一个补充分析，我们使用计划延迟()作为一个额外的因变量，它计算为项目的实际结束日期和计划结束日期之间的差异。对于两个因变量，由于我们不能排除它们的误差项可能相关的可能性，我们使用表面不相关的回归(SUR)模型(Greene 2003)。结果如表3所示。

SUR开发努力模型产生的结果与表2中报告的OLS模型相同。结果表明，领域、技术和客户经验具有显著的学习效应。因此，进度延迟支持H1A-H1C。模型4（检验假设2-4）的结果发现，领域与客户经验、技术与客户经验之间存在替代交互作用。因此，支持H2B和H2C。在进度延迟模型中，领域与技术经验的交互作用不是替代性的。该模型不支持H3和H4。



**4.3 结果讨论**

总的来说，我们所有的假设都支持努力或进度延迟(或两者都支持)。在这11个假设中，领域经验和技术经验学习效应的两个假设(H1A和H1B)在努力和进度延迟模型中都得到了支持。工作复杂性(H3A和H3B)和团队复杂性(H4A、H4B和H4C)的调节作用以及领域经验和技术经验之间的替代交互作用(H2A)仅在努力模型中得到支持。在计划延迟模型中，只支持客户经验的学习效应(H1C)和客户经验与领域(H2B)、技术(H2C)经验的替代交互效应。

有趣的是，努力模型中没有支持的假设，但是在计划延迟模型中支持的假设(例如，H1C、H2B和H2C)都与客户经验相关。进度延迟由项目计划和实际情况决定。ISD项目小组可透过制定切合实际(或准确)的项目计划及/或缩短项目的实际持续时间，以减少进度延误。考虑到减少ISD项目的实际执行时间的困难(Brooks1995)，进度延迟很可能受到项目计划准确性的影响。我们推测，客户经验主要影响计划的开发方式，而不是实际产生的努力。客户经验学习效应(H1C)的结果表明，客户经验减少了进度延迟，但不影响努力，支持了这一观点。客户经验增加了ISD项目团队对客户的风格、条件、计划和限制的总体了解，也增加了项目团队与客户讨价还价的能力。增加了解及议价能力，可有助ISD项目小组制定更切合实际及准确的项目计划。这当然与领域和技术经验的好处相重叠，这些经验可以帮助ISD项目团队更好地理解目标应用程序的领域和技术平台，从而提高项目规划能力。另一方面，除非客户经验提供了与目标应用程序的领域和技术相关的可重用知识，否则客户经验似乎不太可能取代领域和技术经验在减少理解领域和技术问题的实际工作方面所带来的好处。客户可能会为不同的系统使用不同的技术环境，与焦点客户的许多经验可能是在不同的领域或技术设置中。在不同的领域和技术环境中与同一客户一起工作的以前的经验，对于取代以前在领域或技术方面的经验在减少实际工作方面的作用可能不是很有用。

努力和进度延迟结果之间的另一个主要差异是ISD项目复杂性的调节效应(假设3和4)。Similarto Espinosa等(2007)发现任务熟悉度和任务复杂性之间存在负向交互作用，我们发现随着两种类型的ISD项目复杂性的增加，这三种类型的经验的作用减弱。然而，有趣的是，这些调节效应只在努力方面被观察到，而不是在进度延迟方面。这可能是由于绝对和相对性能度量之间的差异。回想一下，努力是ISD性能的绝对度量(实际付出的努力)，而计划延迟是相对度量(计划日期和实际日期之间的差异)。ISD项目的复杂性可能会以类似的方式影响实际情况和计划，在这种情况下，相对度量可能无法捕获变更，而绝对度量可能可以。例如，ISD项目的复杂性可能会增加实际的持续时间(Xia和Lee2005)，但它也可能帮助项目团队，特别是那些有更多经验的团队，制定更现实的项目计划，这将有效地满足对所需额外努力的期望，并将减轻复杂性的抵消性缓和效应。在这种情况下，绝对度量(实际工作或持续时间)将捕获ISD项目复杂性的调节效应(H3和H4)，而相对度量(计划减去实际持续时间)可能不会。领域和技术经验之间的交互项对努力和进度延迟有混合的结果——对预期的努力发现了替代的交互效应(H2A)，而对进度延迟发现了互补的交互效应。这些混合结果表明，在减少实际努力方面，领域和技术经验是替代的；然而，同时拥有领域和技术经验对开发现实的项目计划具有协同的积极作用。

现在，我们把注意力转向讨论我们的结果的含义。关于以往经验对ISD项目绩效主要影响的结果(模型3)，如预期(假设1)所示，ISD项目团队的领域和技术经验对减少开发努力和进度延迟均有显著影响。而客户经验对于减少进度延迟有显著的影响。这突出了领域、技术和客户体验的重要性，并且在ISD环境中，这三种知识类型确实会产生学习效应，在ISD环境中，这三种知识类型被不同地组合在一起，为ISD项目中的不同客户交付各种类型的信息系统。这也表明，即使不同ISD项目的项目小组组成不同，小组成员也可以通过集体学习转移过去的经验，以完成后续的ISD任务。我们回归估计(表3中模型3)表明，对于一个典型的ISD项目，如果项目团队多一个单位的领域(或技术)经验，项目可以减少4.2%(或23.5%)的开发努力和1.6(或6.3)天的进度延迟，而多一个单位的客户经验，项目可以减少6.7天的进度延迟。

对于不同经验之间的交互作用效应(假设2)，结果表明，随着领域或技术经验的增加，客户经验的作用减弱。然而，这种效应只会发生在进度延迟的情况下，而不会发生在努力的情况下。对于领域经验和技术经验之间的交互作用，随着领域经验的增加，技术经验对努力的影响如预期的那样减小，但对进度延迟的正向影响放大。对于一个典型的项目，如果项目团队多了一个单位的领域(或技术)经验，当客户经验增加一个单位时，进度延迟的有益效应(即边际减少)将减少1.6(或7.2)天。此外，增加一个单位的技术经验，增加一个单位的领域经验所带来的边际工作量减少率将减少3%，而计划延迟的边际减少率将增加1.7天。

在减少开发努力方面，技术和客户经验的影响被任务复杂性所抵消(假设3)。对于一个典型的项目，如果项目的技术复杂性增加一个单位，增加一个单位的技术经验，开发工作边际降低率将减少2%。类似地，当客户不在同一集团(即高客户复杂性)时，增加一个单位的客户经验所带来的边际努力减少率将比在同一个集团（即低客户复杂性）时降低21.7%。对于ISD项目团队复杂性的调节效应，发现团队复杂性可以抵消领域、技术和客户经验在减少开发努力方面的影响(假设4)。对于一个典型的项目，当开发团队的规模增加一个单位时，增加一个单位的领域、技术和客户体验所带来的边际努力减少率将分别降低0.4%，27%和1.9%。

**4.4 结果的稳健性**

我们进行了几个附加的分析，以检验我们的结果的稳健性。为节省空间，本文仅对结果进行总结；详情见在线附录。首先，虽然我们使用估计的项目持续时间和努力来控制项目规模，但是其他的措施，如功能点的数量(Kemerer1993)已经被提议作为ISD项目规模的适当衡量。我们已经控制了功能点，并且发现在分析中包含功能点并不会改变我们的主要结果。其次，我们在分析中纳入了知识折旧(或遗忘)效应，以查看ISD中是否存在折旧效应，并验证我们的主要发现对于纳入可能的知识折旧是否具有健壮性。结果没有显示任何显著的折旧效应。第三，我们所研究的公司的一般政策是不让项目经理挑选特定的团队成员。相反，替代级的人力资源经理负责人员配备项目，以平衡全组织的资源需求。然而，项目经理可能仍然会对团队成员的选择施加影响，在这种情况下，结果可能会有偏见，因为项目团队是内生决定的(例如，项目经理可能倾向于选择表现更好的成员，而避开表现较差的候选人)。我们使用Huckman等人(2009)使用的方法检查了潜在的选择偏差，但没有观察到任何这种偏差的迹象。

**5 结论**

**5.1 贡献**

我们研究的主要贡献是更好地理解了学习效应是如何在ISD中发生的。由于ISD的不同特征(不同的任务、团队和项目复杂程度)，观察学习效应和理解学习的潜在机制一直是一个挑战。利用学习迁移理论，我们的研究提出了一种概念化相同经验的新颖方法，使用ISD知识的三种主要类型，即领域、技术和客户知识，结果表明在这三种知识类型的水平上，重复经验会产生显著的学习效应。

此外，这项研究更好地理解了三种基本 ISD 知识类型中的经验如何相互作用，这样当给定类型的经验更（或不那么）丰富时，ISD 项目团队成员可以更少（或更多）依赖另一种类型的经验来达到相同的项目绩效水平。例如，当ISD项目团队缺乏与客户打交道的经验时，它可能会更多地依赖领域和技术经验来有效地执行项目。我们的研究结果表明，客户经验虽然不同于领域或技术经验，但与领域和技术有共同的要素，因此学习效应是替代性的。发现不同类型的经验之间的交互作用对ISD和一般的组织学习文献都有重要的贡献，因为它使我们能够更好地理解在什么条件下，一种类型的经验是另一种经验的补充或替代(Argote和Miron-Spektor2011)。

我们的研究还表明，ISD项目的复杂性在本质上是不可减少的，因为对于更复杂的ISD项目，领域、技术和客户经验的有益影响显示出递减的回报。这意味着固有的复杂任务使项目团队更难有效地触发必要的认知过程来检索和利用其先前的经验，这表明对于更复杂的ISD项目，需要更多的先前经验来维持相同水平的 ISD 项目绩效。一些可能无法从重复的任务经验中获得的能力，如处理不确定性的“实用智能”，已被发现对更复杂的ISD项目更有益（Langer 等人，2014 年）。然而，当谈到任务经验时，文献普遍认为，随着 ISD 项目变得越来越复杂，固有的复杂性最终变得不可减少，并且无法通过开发人员的任务经验来补偿（Brooks1995），对于复杂的 ISD 任务，仅靠任务经验实现显著的改进是是不可能的（Espinosa et al.2007）。组织学习文献中的研究结果与此一致。例如，Argote等人(1995)在他们关于折纸任务的实验中表明，相对于更复杂的任务，在更简单的任务中，任务经验对性能的提高更强。本研究从学习迁移的角度为这种消极交互效应的产生提供了理论解释。

这项研究也提供了重要的管理启示。我们的经验发现为ISD的实际问题提供了见解，如如何分配团队成员到项目中以提高绩效，如何发展和保留ISD的专业知识，以及如何为信息系统专业人员设计职业发展路径。例如，项目人员配备通常被认为是信息系统的需求与团队成员的知识(即，以前的经验)相匹配的问题。规范的人员配备解决方案是仅让那些拥有与项目相关知识的开发人员来为项目团队配备人员。然而，在许多情况下，这并不总是可行的，因为有限的人力资源和/或由于专门团队成员先前参与其他项目而造成的限制。我们的结果表明，在其他条件相同的情况下，如果ISD项目团队更熟悉客户，领域和技术经验的影响力就会降低，如果项目更复杂，任何与领域、技术或客户的经验也会变得不那么有影响力。

**5.2 不足和未来展望**

和大多数研究一样，本研究也不是没有局限性的，在解释研究结果时应该考虑到这些局限性。一个局限是由于本研究中感兴趣的现象是微观层面变量(如个体团队成员的经验)对宏观层面变量(如项目团队绩效)的影响。据我们所知，对于这样的微观宏观情况，还没有统计技术。例如，层次线性模型通常处理宏观微观情况(即，宏观自变量和微观因变量)。目前，处理微观-宏观多层数据的唯一可行方法是将微观层面的变量聚合成宏观层面的变量(Chan 1998, Klein et al.1994)。在本文中，我们使用组平均聚合方法，该方法被认为在方法上适合于像我们这样的可加性组合模型(Chan1998)。这种聚合方法也被广泛用于捕获ISD环境中的团队水平的学习效应(例如，Bohetal。2007, Espinosa等人，2007,Huckman等人，2009,Langer等人，2014)，以及其他情况，如外科团队(如reagan等人，2005)。

另一个局限是，我们的分析是在知识类型水平，而不是在个人知识类别水平。将相似的知识类别划分为有意义的更高层次的类型，如领域、技术和客户，已经被许多相关研究采用(如Langer et al.2014, Bapna et al. 2013)，这种方法通常提供了有意义的见解。然而，内在层面的问题仍然存在。与其他类似的研究一样(例如，Bapna等人，2013年)，我们在知识类型层面上的发现(例如，领域、技术和客户知识)不一定适用于每一个特定的知识类别(例如，电子银行、Java等)。此外，其他类型的知识和技能，如项目管理知识(Henry et al.2007)、人际交往和沟通技能(Langer et al. 2014)和行为技能(Jiang et al.1998)，也与ISD项目相关。然而，我们将本研究的重点局限于ISD中使用的三种主要知识类型的分类，即领域、技术和客户知识。尽管如此，调查其他可能类型的知识的影响将是未来研究的一个富有成果的领域。

与之前的文献(如Espinosa et al.2007, Xia and Lee2005)一致，我们认为团队和任务复杂性是ISD复杂性的两个显著维度。然而，我们承认，ISD项目的复杂性可能涉及其他维度(例如，团队的地理分散也可能代表团队结构的复杂性)。研究ISD项目复杂性的其他有意义的维度的调节作用是有意义的。

最后，我们的经验数据(包括领域、技术和客户知识的分类)来自一家提供ISD服务的大型IT服务公司。这种集中的研究设置增加了结果的内部有效性。另一方面，这些结果可能会受到研究地点的特性的影响。在其他背景下的未来研究将有助于进一步推广这些结果。