

RPA 全攻略之技术篇——七大秘籍，助你“玩转” RPA

(来源：安永中国公众号，2019-09-27)

前言

机器人流程自动化（RPA, Robotic Process Automation）无疑是目前商用最为成熟、应用最为广泛、收益最为显著的数字化技术之一，其应用场景从早期的财务税务领域，不断地向其它业务领域扩展和尝试，包括人力资源管理、信息系统运维、供应链管理、客服中心、法务、内审等。



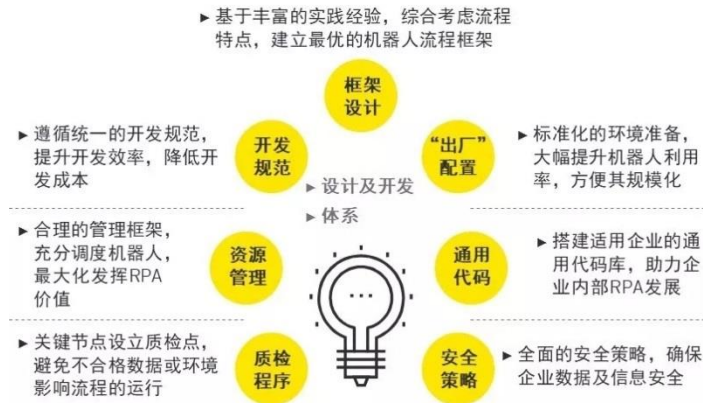
图一 RPA 可应用的不同业务领域

在大量实施和交流的过程中，安永发现：RPA 的实施方法决定了其运行稳定性及后续运维复杂性。一个优秀的设计理念及实施过程，不仅可以保证 RPA 运行稳定，并且方便后续的运维和优化。

安永基于以往丰富的 RPA 实施案例，并结合主流 RPA 产品和业务流程，梳理总结出一套行之有效的 RPA 设计理念。基于这套设计理念，可以有效地规避企业在部署 RPA 中经常遇到的难题，如 RPA 运行经常不够稳定意外终止运行、流程异常中断后需要人工调试重启、自动化

流程增加导致代码管理难度直线上升、RPA 开发成本居高不下、运维工作量大等等。

这套设计理念包括七大部分，分别是：框架设计、开发规范、出厂设置、资源管理、通用代码、质检程序、安全策略，以下内容将详细对七大部分展开说明。



RPA 七大设计理念

1. 框架设计

要保证信息系统的建设成功以及可持续发展，一份优秀的框架设计方案必不可少。同样，在 RPA 项目实施过程中，一份针对目标流程的优秀框架设计，是保证流程顺利自动化并持续稳定运行的关键。

“我们需要端到端的实现自动化流程”；“可不可以自由地添加我想要机器人运行的组织机构”；“出现这种情况需要及时地向业务人员反馈，甚至有必要的时候要终止流程”；“最近网络不是很稳定，假如 RPA 停止了能不能继续上次的运行结果继续运行”；“我们明天可能业务量会增加一倍，还需要进行二次开发么”……

这些问题常常出现在 RPA 项目的需求沟通阶段。客户不同的角色导致了他们对 RPA 流程存在不同的需求方向，而如何满足不同的需求，则是对 RPA 框架设计最大的考验。

基于大量的开发经验和成果，安永认为，在进行流程自动化的框

架设计与开发的过程中，应当遵循四大原则：“安全、灵活、稳定、高效”，同时也要考虑到业务标准化对未来的可延展性需求。

为了保证这四大核心原则在框架设计的过程中都得到了充分的考量，我们从业务流程的长度、复杂度、关键流转节点、检核点、校验逻辑等流程内部影响因素出发，结合机器人运行时间、运行时长、运行环境等外部影响因素。从需求衔接、本地化参数与配置、风控与恢复机制、结构化开发、快速拓展需求、全局性维护考量这六大方面实现全面、完整的框架设计，并在流程执行前、执行中、执行后三个阶段将设计思想贯穿整个流程。

例如：

流程执行前：

进行大量的环境检查和分析，包括输入文档，配置文件，初始运行环境状态等等。

流程执行中：

(1) 根据流程运行涉及的系统、流程执行节点、流程长度等因素将整个流程进行切分，确保不同功能模块的低耦合性。

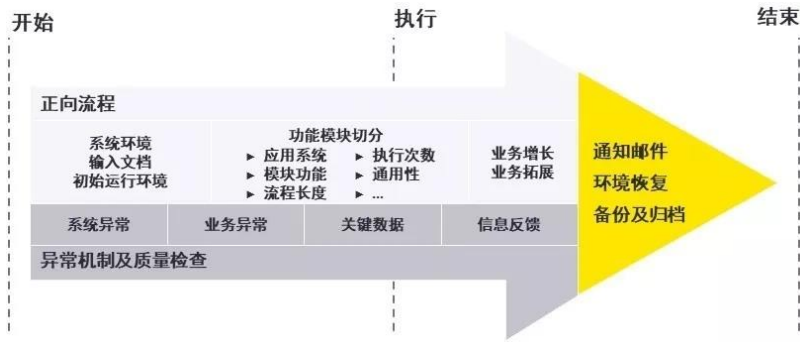
(2) 充分考量未来业务增长或拓展，预留衔接位置。

(3) 梳理常见业务异常状态和可预见的系统异常状态，将其按照异常类型、后续影响、特殊性等维度进行分类，在关键节点进行异常捕获，根据异常的不同划分进行不同的自动化流程指向，确保异常发生时能及时停止、及时跳过当前子流程并继续运行或者重新尝试执行。

(4) 贯穿整个流程的关键数据节点的质量检查及信息反馈，提高整个自动化流程质量。

流程执行后：

按需进行执行结果的反馈、运行环境的恢复以及所有运行相关数据的备份归档等，以便后续流程的运行以及历史记录追溯。



图三 框架设计考虑因素

2. 开发规范

“不以规矩，不能成方圆”，作为 RPA 的实施方，我们的首要目标是确保项目的顺利落地，一套严格的实施标准能从各个方面推进项目快速、准确、高质量的完成。安永从大量的项目经验中总结并设立了一套开发规范与标准，从注释、日志、排版、目录、版本、命名等多个维度出发，应用在整个项目进行中，从而提高项目效率和质量。



图四 安永提倡的各类开发规范

安永认为，从流程步入实施阶段开始，就需要建立统一的工作目录及配置管理，构建出清晰的工作目录结构，并且在关键文件进行

版本控制时按照配置管理的要求，在更新后对关键信息进行标注。例如：针对不同部门、不同业务性质进行流程目录的设定。在独立流程中建立不同系统功能模块子目录。

在模块代码的开发过程中，确保统一的文件命名以及注释的规范性，例如：参数变量的命名采用驼峰命名法或者帕斯卡命名法等，并且不能使用编程语言的保留字。标准化相同系统功能文件的词缀便于阅读，要求增加关键节点的注释说明等。

示例：

变量命名采用驼峰命名，如：*dateTable*, *userName*。

参数命名采用帕斯卡命名法+后缀 *Arg*，如：*DateTableArg*, *UserNameArg*。

参数命名采用匈牙利命名法，如 *in_DataTable*, *out_UserName*。

当整个项目的进行都处在一个标准化的要求下，整个自动化项目实施周期的各个阶段都可从中受益，从开发阶段的效率提升，到测试阶段的迅速解决异常情况，再到运维阶段的代码易读等等各类优势。同时也提升了相关代码的质量及友好性，便于交付后客户自身进行代码管理。

3. “出厂”设置

当企业中 RPA 流程越来越多后，往往需要不断增加机器人数量以满足大量业务流程的执行，而在机器人数量逐渐增加的过程中，企业则会遇见机器人部署麻烦，自动化流程迁移难，机器人资源排班分配难的窘境。

原因是自动化的流程运行时往往需要特定的环境，而一旦机器人都是针对固定流程进行基础配置的话，当出现机器人需要增加资源时，

则要求重新部署并配置对应的生产环境。甚至在出现灾备的情况下，就需要对照现有的不同机器人重新配置新增的机器人，不仅费时费力导致无法快速恢复机器人还容易导致流程运行异常。

安永认为，应该在开发流程前进行对机器人“出厂”设置的标准化，针对客户的需求与现有配置的实际情况，形成一部全流程通用的配置说明书。在每次诞生新的机器人时，只需要按照说明书标准进行配置，即可满足大部分通用全流程的运行。

这样，机器人就从繁杂的“原料”成为了可利用的资源，只要有需求就可以直接拿来使用，再配合控制中心，就可以实现对机器人的规模化运用，提升 RPA 工作效率。



图五 RPA 出厂设置

为了最大化 RPA 使用的便捷性，对配置说明书的编写应做到尽可能的明确与精细。例如对于前台机器人配置，应包括系统环境、分辨率、本地化应用、流程涉及系统等，对于机器人软件环境配置，应该包括 RPA 工具版本、Office 工具版本、客户系统应用版本等，甚至 Java、压缩工具的版本。示例：

Type	No.	Item	Details/Versions	Remark
Basic environment	1	OS	Windows 10	enterprise
	2	Resolution	1920*1080	size*100%
Application program	3	Chrome	76.0.3809.100	
	4	IE	11 or above	
	5	Java	JDK/JRE 1.8.1	
	6	Python	3.7.4 or above	
	7	MySQL	8.0.15 or above	
	8	Microsoft .net framework	4.6.1 or above	
	9	Microsoft Office	2016	
	10	Adobe Acrobat Reader	2017	
	11	7-zip	18	
	12	WeChat	2.6.5 or above	
Business system	13	Wind	18.4.1	
	14	Ukey control system	V1.1	
	15	SAP	750	
	16	Yonyou NC	V6.5	
	17	Kingdee EAS	V1.0	

图六 出厂设置示例

4. 资源管理

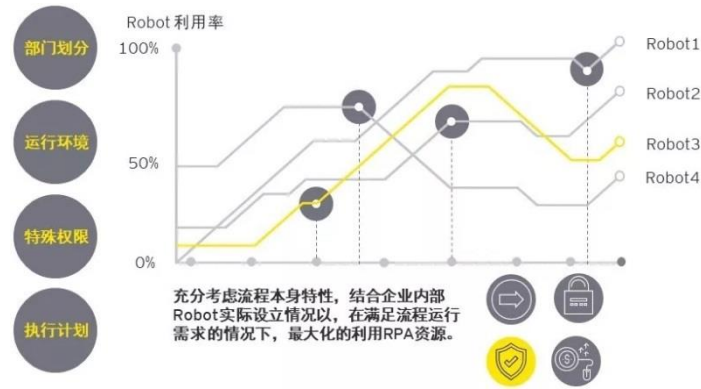
机器人的“抗压”能力超乎你的想象，它可以 24 小时地不停工作，但就目前而言几乎没有企业能充分利用自己的机器人能力，往往是哪里有需要，哪里就新增机器人。这就会导致企业内部机器人空闲时间过多，机器人无法得到有效的利用，大幅降低了机器人项目的收益回报。

本着“集中管控”的原则，安永认为，当我们充分考虑“出厂”设置后，就可以从内部和外部全面对机器人进行管理。有了标准统一的“出厂”设置，机器人就成为了企业可随意利用的资源，按照业务流程的执行计划，就能够对这些机器人进行统一的调度运行，最大化利用 RPA 的能力，并且降低了 License 成本投入。

在企业中，不同的部门有着不同的权限与职能，部门之间往往有着各种边界，同样不同的流程对机器人的运行环境及执行计划也有着不同的要求，有些部门或者特殊的流程还需要特殊配置的机器人。

例如，完成报税工作需要税盘的连接，而企业的税盘通常由税务部门管理；网银相关的业务流程不仅需要搭配能连接 UKey 的环境，

而且需要在特殊的时间段执行。因此，在 RPA 的设计过程中，应该充分考虑到自动化流程的部门划分，运行环境，执行权限与执行计划，在满足企业规章制度的情况下，进行机器人的资源管理。



图七 RPA 资源管理

5. 通用代码库

相信大家对“中国速度，一天三层楼”的新闻并不陌生，基建行业的快速施工的能力来自建筑理念的转变和技术的进步，不再是一砖一瓦浇灌，而是采用成型的工业化模块，通过拼接已组装好的各种材料和模块，省去大量施工时间，显著提升施工效率。同样，在 RPA 部署中我们也可以应用这种理念。

在大量 RPA 实施项目中，安永发现同一企业的不同业务流程，往往会有一些同样的子流程或操作，比如登录 SAP 系统、访问内部的网站或业务系统、验证码的识别等。如果对所有流程都进行独立的开发，会导致开发工作的重复和冗余，且后期维护和更新都需要重复修改多处。但是，如果将这些业务流程中常见的操作和动作封装成一个个组件，并建立企业的 RPA 代码库进行统一管理，就可以形成一个“随手可用，安全可靠，高效便捷”的资源库。通过代码库的建设可以减少开发成本（因为可以避免重复开发），规范开发模式（因为大家用

的都是同一套代码)，方便业务人员自行开发（因为业务人员可以直接调用）等等。



图八 RPA 代码库

安永始终致力于 RPA 代码库的建设。截至目前，安永内部的通用代码库涉及 20 余种应用系统，拥有超过 100 份通用代码。同时，安永拥有一套严格完善的管理体系和方法，包括代码入库审查、代码上架、代码更新、代码维护、代码借用审查、代码退役等等。借助这一整套管理方式，安永可以帮助企业快速建立专属代码库，从而允许企业内部团队基于代码库快速完成业务流程的开发，让熟悉业务流程的业务团队自行发挥，打造属于自己的机器人。

EY Code Library			
File&Data	System	API	Web
File_M_001_Archive	System_SAP_V_001_LoginSAP	API_M_001_VerificationCodeRecognition	Web_V_001_SaveAs
File_M_002_Move File	System_SAP_V_002_TcodeSelection	API_M_002_Database MappingTable-EBS	Web_V_002-001_LoginE-bank_BOC
File_M_003_CopyFile	System_SAP_V_003-001_LayoutForFBL1N	API_M_002_USBDriverInterfaceSwitch over	Web_V_002-002_LoginE-bank_ICBC
File_V_004-Decompress-ZIP	System_ORACLE_V_001_LoginEBS	API_V_001_QiChaCha_Person	Web_V_003-001_Journal_ICBC
Data_M_001_Vlookup	System_JINDIE_V_001_LoginEAS	API_V_002_QiChaCha_Entity	Web_V_003-002_Receipt_BOC
Other_V_001_Screenshot	System_YONGYOU_V_001_LoginNC6	API_M_001_HeHe_CardInformation	Web_V_004-001_Journal_ICBC
Other_V_002_StartWithRun	System_LANGCHAO_V_001_LoginGS	API_M_001_HeHe_CardForDriverCard	Web_V_004-002_Journal_BOC
...

图九 安永 RPA 代码库示例

6. 质检程序

在当今自动化的制造生产线上，整个生产环节会设置多个自检点以移除次品、残品或不合规的产品。这些自检点有物理方式，如称重、震动、翻转等；也有化学方式，如溶解、溶剂反应等各种各样的方式，其目的就是保证生产的最终产品是符合质量要求的。

类似的，安永认为，作为商业中的自动化流程，RPA 需要具备自我检查的质检程序。RPA 通常可以捕获异常情况并作出反应，但有时会出现这种情况：整个流程没有任何异常，但最终结果却是错误的，这往往是因为交互对象的改变而导致的。但自动化流程“无声”的错误可能会带来严重的后果。

例如，某流程需要从特定网站的右上角获取一个金额数据，并将此数据作为输入源流转至后续流程以得出最终结果。但此网站在不知情的情况下进行了更新，原位置上的信息由金额变成了交易量，而 RPA 仍从原定页面元素获取数据，从而导致获取错误的数据并继续后续的流程，进而可能会对最终流程的执行带来严重的后果。

为了尽量规避此类问题，安永认为应该在设计时考虑如何添加质检程序，此功能可从两个角度实现。

自我历史参照：

通过对关键节点数据源的纵向分析，得出此数据的可能范围，并设立预警系统，一旦 RPA 得出的数据超出预定的范围，RPA 会停止工作并通知业务人员。

多来源对比：

对来源不止一处的信息源，可以设置多来源获取数据并进行对比，如果出现无法匹配的情况，则停止工作并通知业务人员。



图十 RPA 需要具备质检功能

7. 安全策略

“道路千万条，安全第一条”，安全无论在哪个领域都异常重要。同样的，安全作为 RPA 开发的重中之重，需要贯穿于整个设计与开发环节。安永认为，为了保证 RPA 实际运行过程中的安全性，需要从下述八个安全维度考虑 RPA 的设计和开发工作。

参数配置安全：

当业务信息输入全部参数化后，RPA 运行所依赖的业务配置表需要加密处理，或放置于有访问级别限制的共享盘中，置于公司防火墙保护下，保障配置信息安全。

信息存储安全：

对于所有的输入、过程和结果文件，需要按照统一标准的方式进行备份，为事后核查及审计要求提供支持。

信息传输安全：

在使用 RPA 发送、传输、流转信息过程中，应尽量避免使用非企业认可的即时沟通软件、外部软件或第三方软件，防止信息泄露的风险。

网络端口与访问安全：

对于需要向外部发送和接受数据的流程，需要对发出和接收的数据进行加密处理，并使用可信任的证书加载，以保证业务数据无法破

译或盗取。

物理环境安全：

若使用实体电脑作为 RPA 载体，则需要确保存放这些电脑或相应如 Ukey 集中器等物理设备的安全，保障通顺的网络环境和供电。同时，应限制存放环境的人员出入，减少与不相干人员的接触。

日志安全：

在较长、较复杂的流程中，应在流程的关键节点添加状态日志，说明当前流程节点，为后续勘查和纠错提供支持。

代码安全：

在庞大复杂的 RPA 项目中，应使用 SVN 等工具进行代码管理，统一管理监控代码开发，防止代码版本错乱，代码被意外修改，代码丢失等问题。

组件安全：

企业自建的 RPA 代码库应由专人管理，并定期维护和更新已开发的组件，保证后续使用的安全和稳定。



图十一 RPA 部署时需要考虑的安全因素

结尾

相信很多与 RPA 打过交道的人都会有一点体会，那就是 RPA 的报

错和异常中断让人非常头疼。安永认为，打造一个机器人很容易，但打造一个健壮、稳定、安全、灵活的机器人却很难。好比一栋高耸的摩天大楼需要坚固的地基和错综复杂但承重均匀的钢筋水泥，RPA也需要类似的架构和代码来保证其稳定、可靠和安全。

所以，在 RPA 开发和部署过程中，应当全面考虑上述七个维度，在实施过程中切实地贯彻这七大理念，力争所开发出来的机器人可以应对异常并妥善处理。

最后需要强调的是，上述七个方面所讨论的是一种指引和标准，但更为重要的是对业务需求的理解程度、对业务流程的熟悉程度、对实际情况的判断分析。如果对业务流程本身就理解不具体，不透彻，不清晰，那再怎么套用上述设计理念都是徒劳低效的。

原文链接：<https://mp.weixin.qq.com/s/VEV3CXLC7UMApR4zMUQhbA>，
转载请注明。