

点击蓝字，立即关注

前言

随着快速迭代和业务复杂度的提升，传统的自动化测试脚本逐渐暴露出维护难、扩展性差、覆盖不全等问题。

如何用更科学、更系统的方法来建模和实现复杂业务流程的自动化测试，是每一位测试工程师必须思考的问题。

有限状态自动机（Finite State Machine, FSM）作为一种经典的建模工具，广泛应用于编译原理、协议设计、流程控制等领域。

近年来，FSM思想也逐渐被引入到自动化测试领域，成为提升测试系统性、可维护性和自动化程度的重要利器。

本文将以Python+Playwright+Pytest+BDD为技术栈，系统地讲解如何在自动化测试框架中引入FSM思想，希望你能从本文中获得有价值的参考和启发。

有限状态自动机（FSM）理论基础

2.1

FSM的基本概念

有限状态自动机（FSM）是一种用于描述系统在有限个状态之间转移的数学模型。

它由以下几个核心要素组成：

- **状态**：系统在某一时刻所处的情形。例如，用户未登录、已登录、已注册未验证等。
- **事件/输入**：触发状态转移的外部动作或条件。例如，点击登录按钮、输入验证码、点击退出等。
- **转移**：由事件触发的状态变化过程。例如，用户输入正确账号密码后，从“未登录”转移到“已登录”。
- **初始状态**：系统开始时的状态。
- **终止状态**：系统结束时的状态（可选）。

FSM通常用状态转移图或状态转移表来表示。状态转移图用节点表示状态，用有向边表示转移；状态转移表则用二维表格列出所有状态和事件的对应关系。

2.2

FSM的分类

FSM主要分为两类：

- **确定性有限状态自动机（DFA）**：在任何时刻，系统在某一状态下对某一输入只有唯一的转移。
- **非确定性有限状态自动机（NFA）**：在某一状态下对某一输入可能有多个转移。

在自动化测试领域，通常采用DFA进行流程建模，因为业务流程大多是确定性的。

2.3

在自动化测试中的优势

FSM在自动化测试中的应用，带来了如下显著优势：

1. 流程建模清晰：将复杂业务流程拆解为状态和转移，结构化表达系统行为，便于理解和沟通。
2. 测试用例自动生成：可根据FSM自动生成覆盖各种状态和转移的测试用例，提升覆盖率。
3. 异常路径检测：便于发现系统在异常输入下的行为，提升健壮性。
4. 可维护性和可扩展性：流程变更时只需调整FSM模型，测试代码易于维护和扩展。
5. 与自动化框架无缝集成：FSM思想可与主流自动化测试框架（如Selenium、Playwright、Appium等）无缝结合，提升整体测试能力。

应用场景

3.1

典型应用场景

1. 用户注册与登录流程：如注册、邮箱验证、登录、退出等多状态流程。
2. 订单处理流程：如下单、支付、发货、收货、退货等。
3. 权限与审批流程：如申请、审批、驳回、通过等。
4. 多页面、多分支业务流程：如电商购物、内容发布、数据分析等。

3.2

适用性分析

FSM适用于以下场景：

- 业务流程复杂、状态众多、分支多。
- 流程经常变更、需频繁扩展。
- 需自动生成测试用例、提升覆盖率。
- 需对异常路径、边界条件进行系统性测试。

对于简单的线性流程，FSM虽可用，但未必必要。对于复杂流程，FSM则能极大提升测试的系统性和可维护性。

项目结构设计

4.1

推荐项目结构

良好的目录结构是高效开发和维护的基础。结合FSM思想，pytest-playwright框架项目结构设计如下：

```
1 pytest-playwright-demo/
2   ├── fsm/
3   |   └── user_fsm.py
4   ├── pages/
5   |   ├── base_page.py
6   |   ├── login_page.py
7   |   ├── register_page.py
8   |   └── dashboard_page.py
9   ├── tests/
10  |   └── test_user_flow_fsm.py
11  ├── conftest.py
12  ├── requirements.txt
13  └── README.md
```

- **fsm/**：存放FSM模型相关代码。
- **pages/**：页面对象模型（POM），每个页面一个类，便于维护。
- **tests/**：测试用例目录。
- **conftest.py**：Pytest的全局fixture配置。
- **requirements.txt**：依赖库清单。
- **README.md**：项目说明文档。

4.2

页面对象模型（POM）设计

页面对象模型（POM）是一种将页面操作与测试逻辑分离的设计模式。每个页面封装为一个类，包含页面元素和操作方法，便于维护和复用。

4.3

FSM与POM的结合

- FSM负责流程建模，POM负责页面操作。
- 测试用例只需关注流程和断言，页面细节交由POM处理。
- FSM与POM解耦，提升代码复用性和可维护性。

.....

本文节选自[第八十六期《51测试天地》](#)

原创文章

[《【测试干货】有限状态自动机（FSM）在自动化测试框架中的实战》](#)

文章后续将继续为大家介绍

[“复杂业务流程FSM建模与实现”](#)

[及FSM在自动化测试中的进阶应用”等内容](#)

想继续阅读全文或查看更多[《51测试天地》的原创文章](#)

请点击下方[阅读原文或扫描二维码](#)查看



公众号：51Testing软件测试网

声明：本文为51Testing软件测试网blues_C用户投稿内容，该用户投稿时已经承诺独立承担涉及知识产权的相关法律责任，并且已经向51Testing承诺此文并无抄袭内容。发布本文的用途仅为学习交流，不做任何商用，未经授权请勿转载，否则作者和51Testing有权追究责任。如果您发现本公众号中有涉嫌抄袭的内容，欢迎发送邮件至：editor@51testing.com进行举报，并提供相关证据，一经查实，将立刻删除涉嫌侵权内容。

每日有奖互动

你在测试复杂业务流程时，通常采用什么方法进行建模？

是否尝试过文中介绍的FSM？

