

# Study on Anchor Point Positioning Element Method for RPA System

Changping Chen Yansheng Li Wenjie Yang Yajun Zhou Ziming Wei

Sichuan CLP Qiming Information Technology Co., LTD., Chengdu, Sichuan, 610000, China

**【Abstract】** Robot process automation (RPA) is a kind of predefined business processing process based, using software to simulate human operations, used to replace or assist employees to complete the repetitive, fixed process work in computer equipment. When using RPA software to automate browser page elements, it is often necessary to operate some elements without unique feature attributes and location dynamic change. If positioning such elements is only through the basic positioning technology, it is necessary to add index values to obtain specific elements. However, when the target element position changes in the page, the element that may not be acquired when getting the element using the base location plus index value is not the desired target element. To solve the above problems, the paper proposes a —— anchor point positioning method suitable for the RPA system. Experimental results show that the use of anchor point positioning method can solve the problem of dynamic position change.

**【Keywords】** anchor point positioning; RPA; robot process automation DOM element search; web page automation

## 适用于 RPA 系统的锚点定位元素方法研究

陈昌平 李彦生 杨文杰 周亚俊 魏子明

四川中电启明星信息技术有限公司, 中国·四川 成都 610000

**【摘要】** 机器人流程自动化 (RPA) 是一种基于预定义业务处理流程, 采用软件模拟人类操作, 用于代替或辅助员工在计算机设备中完成重复性强、流程固定的工作。在使用 RPA 软件对浏览器页面元素进行自动化操作时, 经常需要操作一些自身无唯一特征属性并且位置动态变化元素, 定位这类元素如果只是通过基础定位技术, 必要加上索引值才能获取到具体元素。然而当目标元素在页面中位置发生变化后, 使用基础定位加索引值获取元素时可能获取不到或获取到的元素不是期望的目标元素。为解决上述问题, 论文提出一种适用于 RPA 系统的用于解决动态位置变化的元素定位方法——锚点定位法。实验结果表明, 使用锚点定位法对元素进行定位, 可以解决动态位置变化的元素定位问题。

**【关键词】** 锚点定位; RPA; 机器人流程自动化; DOM 元素搜索; 网页自动化

DOI: 10.26549/bdai.v3i3.11789

### 1 引言

机器人流程自动化 (Robotics Process Automation) 简称 RPA, 也时常被称为数字化员工。RPA 软件在技术上类似图形用户界面测试工具<sup>[1]</sup>, 是一种基于预定义业务处理流程, 采用软件模拟人类操作, 用于代替或辅助员工在计算机设备中完成重复性强、流程固定的工作。在使用 RPA 软件对浏览器页面元素进行自动化操作时, 经常需要对一些自身没有唯一特征属性, 只能靠 class 和 tagName 等非唯一结果基础定位技术实现定位。比如表格中由于查询条件、排序方式、数据权限等不同, 经常会发生的是同一条数据, 在不同情况下它在表格中的位置是不同的, 如何在动态变化的情况下准确找到目标数据所对应的元素是完成流程自动化的关键技术难点。通过 class, tagName 获取到的结果并不一定是唯一的, 所以想要获取到最终目标元素都必须借助索引才行, 但借助索引之后就必须要保证元素在网页中的位置不能发生变化, 否则会导致获取元素失败或获取到错误元素问题。

为解决上述问题, 论文提出一种适用于 RPA 系统的用于解决动态位置变化的元素定位方法——锚点定位法。其核心思想是通过一个固定的“锚点”, 利用锚点构建一个与目标拥有关系的中间元素, 通过缩小查找范围, 使其目标元素即使没有唯一的特征属性, 也能通过基础定位获取到唯一一个元素——目标元素。实验结果表明, 使用锚点定位法对元素进行定位, 可以解决动态元素定位失败、定位错误等问题。

### 2 常用基础元素定位技术及局限

所谓定位技术, 表示的是一种可以获取到网页元素的技术手段。本章节首先介绍现有元素定位的技术和每一种在定位技术在 RPA 系统用的使用局限, 便于更好地理解为什么需要锚点定位法。

#### 2.1 ID 属性定位及局限

W3C 规定每一个元素的 ID 应该是唯一的, 所以可以使用元素 ID 对元素进行唯一定位, 因此使用 ID 定位元素可以忽略元素位置, 只要元素拥有 ID 属性并保证了它的合法性, 就能通过 ID 定位到目标元素。但是在实际网页中很多元素没有 ID, 而且在一些老旧特殊的业务系统, 也可能存在一些 ID 值相同的元素; 另外还有一些情况是元素拥有 ID, 但 ID 值是动态生成的, 每一次刷新页面都会随机生成一个新的

**【作者简介】** 陈昌平 (1986-), 男, 中国四川自贡人, 工程师, 从事 RPA、搜索引擎、大数据研究。

ID, 而这样的 ID 在没有掌握随机生成规律之前是不能用作定位元素的, 所以如果仅仅通过 ID 定位元素只能解决一小部分元素定位问题。

### 2.2 class 和 tagName 定位及局限

class 和 tagName 应该是出 ID 之外, 使用频率最高的定位手段, 但由于 class 和 tagName 在网页并不是唯一的, 通过它们定位元素时, 得到的结果通常是一个数组, 而我们需要目标元素只是包含在这个数组中的一个元素而已。所以通过 class 和 tagName 定位元素时, 对元素位置是敏感的, 在采用 class 和 tagName 定位元素时, 如果没有其他属性进行配合实现唯一元素定位, 那么只有在保证元素位置固定情况下, 才能准确定位元素。

### 2.3 XPath 定位及局限

XPath 可以理解为是从一个起点到目标元素的路径, 从 HTML 根到目标元素的 Xpath 称为全路径: 除了全路径之外, 还可以从一个元素到目标元素, 这种称为相对路径。XPath 可以通过索引进行精确定位, 避免出现同时定位多元素的问题<sup>[2]</sup>。但是针对页面结构复杂的网站, 使用全路径的 XPath 对元素进行定位会出现元素路径过长的问题, 导致定位速度慢<sup>[3]</sup>; 并且由于 XPath 也使用了所以进行精确定位, 因此当元素位置发生变化后, xpath 也会随之变化。

#### 2.3.1 基于基础定位技术实现的 RPA 选择器和局限

RPA 系统本身是模拟员工进行自动化操作, 而它操作的目标就是界面上看得见的一个个按钮或其他 UI 元素。人靠眼睛识别并定位这些看得见的元素, 而 RPA 系统就是靠“选择器”来实现元素定位, 所以选择器是一个 RPA 系统的核心功能之一, 从一定程度将, 选择器也决定了一个 RPA 系统的上下限能力。

对于 Web 元素的选择器, 基本是基于 id、class 和 tagName 三种基础定位技术加上元素其他属性实现定位的, 在上述三种手段都失效的情况下可能还会使用到 Xpath 和 Css-Selector。通常情况下通过这样的实现方式可以定位大部分的元素, 但是由于业务系统的复杂性, 元素本身没有更多的元素属性可以给选择器使用, 那么基础定位加上元素属性能够获取到的

就不是唯一元素, 而是一个元素数组, 需要的目标元素也只是这个数组中的一个元素而已。因此想要根据一个选择器获取到唯一一个元素, 就必须加上目标元素在数组中的索引值, 所以 RPA 系统的元素选择器需要有三部分内容, 即: 基础选择器+[属性过滤]+[索引值]。

例如: “.edit-btn[value=编辑]:2” 这个选择器表示的是获取 class 中有样式 “edit-btn” 并且 value 属性是 “编辑” 的所有元素, 然后获取结果中的第 3 个元素返回。其中索引值 “2” 仅仅表示了当前这个元素实在结果数组中的位置, 但是由于业务变化或渲染逻辑发生变化, 我们的目标元素在结果数组中的索引可能大于 2, 也有可能小于 2。

动态元素定位的难点是由于元素对应的 DOM 在界面上的位置发生了变化, 导致获取到的元素不是预期元素。而造成这一问题的原因就是原因的 “索引值” 随着 DOM 变化而失效。

例如需要通过 RPA 来自动编辑员工的个人信息, 通过分析发现每条数据的编辑按钮上只有一个相对比较特定的样式类(class) “edit-btn”, 再无其他信息可以参与到元素筛选, 如图 3 所示。因此只能构造成一个选择器 “.edit-btn” 来进行定位, 但是当前情况下包含 “edit-btn” 这个样式的元素有 4 个, 因此通过选择器获取到的元素数组长度为 4, 第一个元素表示第一条数据的编辑按钮, 其他以此类推。如果需要编辑职员刘明生的信息, 数据变化之前如图 1 所示, 刘明生在表格的第三行, 因此如果要编辑刘明生的信息, 需要触发的编辑按钮选择器为 “.edit-btn:2”, 表示样式类(class) 中包含 “edit-btn” 样式的第 3 个元素。如果在数据永远不变化的情况下, 通过 “.edit-btn:2” 选择器可以始终定位到刘明生数据对应的编辑按钮; 但是由于数据是变化的, 当数据变化为如图 2 所示时, “.edit-btn:2” 这个选择器定位到的编辑按钮变成了黄晓霞这个职员的编辑按钮, 这时就发生了 “定位错误现象”。元素定位错误后, 轻则影响流程执行失败, 重则导致数据修改错误等严重问题。

序号	职员	入职日期	岗位	操作
1	王齐	2016-05-03	工程师	<input type="button" value="编辑"/> <input type="button" value="删除"/>
2	张凯峰	2016-05-02	工程师	<input type="button" value="编辑"/> <input type="button" value="删除"/>
3	刘明生	2016-05-04	项目总监	<input type="button" value="编辑"/> <input type="button" value="删除"/>
4	黄晓霞	2016-05-01	财务会计	<input type="button" value="编辑"/> <input type="button" value="删除"/>

图 1 职员数据表格 1

序号	职员	入职日期	岗位	操作
1	张凯峰	2016-05-02	工程师	<span>编辑</span> <span>删除</span>
2	刘明生	2016-05-04	项目总监	<span>编辑</span> <span>删除</span>
3	黄晓霞	2016-05-01	财务会计	<span>编辑</span> <span>删除</span>
4	张小小	2016-05-01	人事专员	<span>编辑</span> <span>删除</span>

图 2 职员数据表格 2

序号	职员	入职日期	岗位	操作
1	刘明生	2016-05-04	项目总监	<span>编辑</span> <span>删除</span>
2	王齐	2016-05-03	工程师	<span>编辑</span> <span>删除</span>
3	张凯峰	2016-05-02	工程师	<span>编辑</span> <span>删除</span>
4	黄晓霞	2016-05-01	财务会计	<span>编辑</span> <span>删除</span>

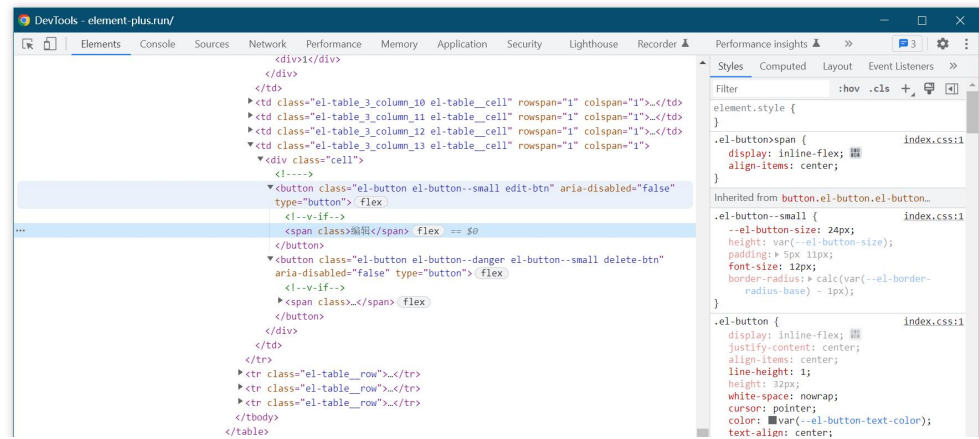


图 3 职员数据表格元素分析图

### 2.3.2 锚点定位的实现原理

#### (1) 锚点定位元素

锚点定位原理是基于基础定位技术,首先是获取目标元素的 `xpath`,然后在目标元素(`target`)周围寻找一个不受 DOM 位置影响,可以通过基础选择器加属性过滤进行唯一定位的元素作为锚点 (`anchor`),然后获取锚点的 `xpath` 并与目标元素的 `xpath` 进行比较计算得到与目标元素共同的父级元素(`P`),再计算出锚点到父级的层级为 `h`;在 `P` 范围内检查是否能够通过 `class`、`tagName` 属性唯一定个位元素,如果无法通过简单选择器定位,则计算 `P` 到目标元素的相对 `Xpath`,最终可以得到一个目标元素的锚点定位信息:

```
{anchor: {tagName:span,text:"xxx"},target: {xpath:"xxx",tagName:"xxx",class:"xxx", ..., up:h}}
```

锚点定位流程如图 4 所示。

在锚点定位信息中, `anchor` 表示锚点元素的定位信息; `target` 表示需要定位的目标元素,其中如果出现了 `xpath` 将不会再出现其他属性,否则将不会包含 `xpath` 属性; `up` 表示从锚点元素开始向上通过几层能找到锚点和目标元素的共同父类。锚点定位的关键在于锚点元素的选择,锚点元素一定是一个可以通过基

础选择器加属性筛选能够唯一定位的元素。

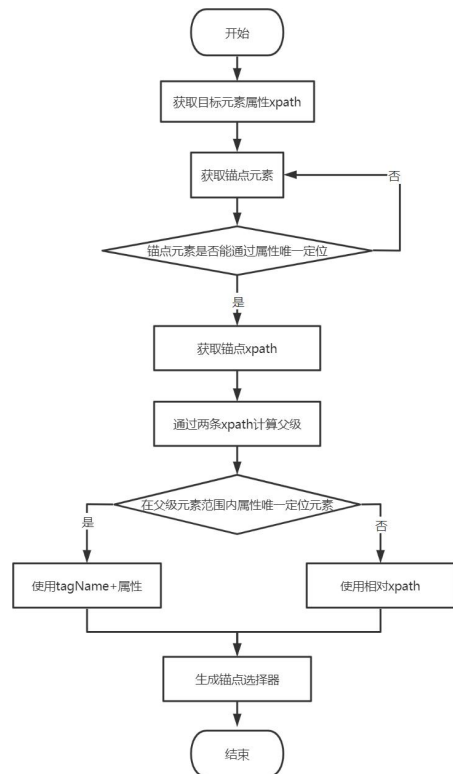


图 4 锚点定位元素流程图

(2) 通过锚点查找元素

基于锚点查找目标元素原理是通过 anchor 包含的定位信息找到锚点元素,然后从获取锚点元素的第 h 层父级;最后根据获取到的父级元素,通过 tagName+属性(此时 class 将作为普通属性使用)或 xpath 定位到最终目标元素。具体锚点查找元素定位流程如图 5 所示。

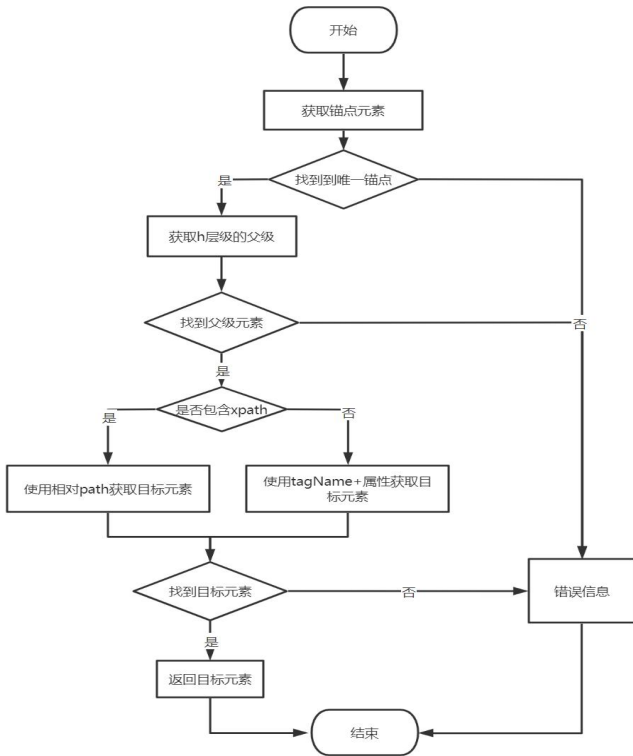


图 5 通过锚点查找元素流程图

(3) 实际应用效果

图 6 给出了 RPA 软件的部分界面,主要功能是创建一个“编辑岗位”的 RPA 流程,以实现自动打开 Chrome 浏览器,对中网页元素的一系列自动化操作。具体操作步骤为:

(1) 首先流程定义了两个关键变量 employee 和 job, employee 表示当前需要编辑的雇员名字, job 表示需要修改为的新职位;

(2) 第 1 步打开 Chrome 浏览器并访问指定 URL 打开业务系统;

(3) 第 2-4 步是自动输入系统账号/密码并点击登录按钮,第 5 步登录系统后直接点击按钮进入雇员信息维护界面;

(4) 第 6 步点击需要编辑雇员的编辑按钮,按钮的选择器如图 7 所示。

由于在选择器中包含了索引值 2,因此当数据表情况如图 1 所示时,执行第 6 步时可以正常触发雇员“刘明生”的编辑按钮。但是当数据如图 2 所示时,执行后雇员“黄晓霞”的岗位将被错误地修改为了“产品总监”。

图 8 是添加了锚点后的选择器信息,第一个 web 表示的就是锚点元素,其中 aaname (表示元素内可见的文本)使用了 employee 变量,表示锚点元素的 innerText 等于 employee 变量;在 anstor 中 upIndex 等于 4,表明锚点元素向上 4 层找到与目标元素的共同父级,然后在父级范围通过 tagName+aaname 可以定位到唯一元素。

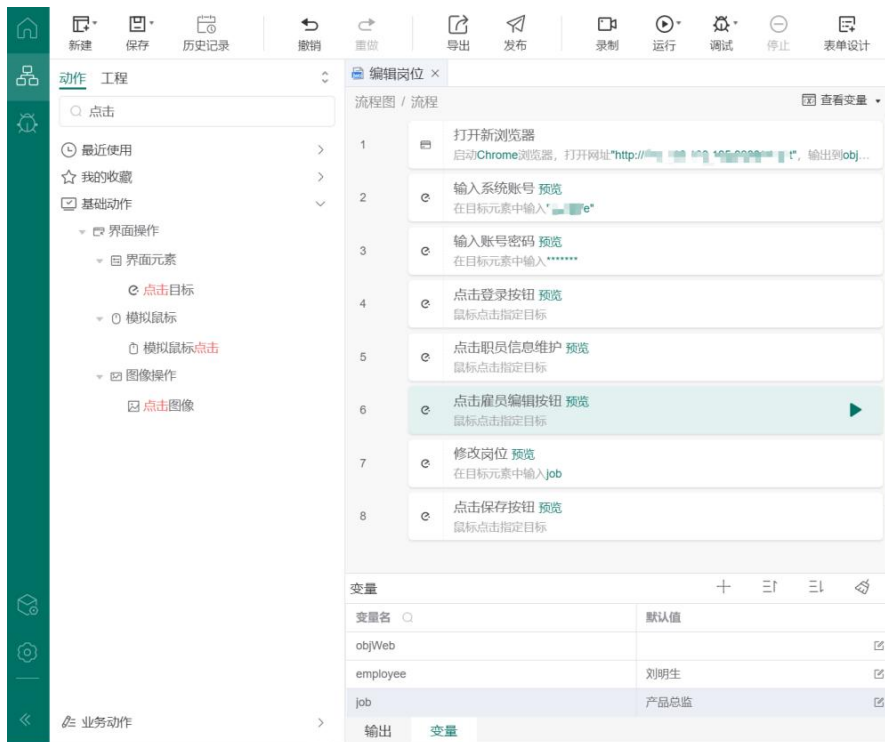


图 6 自动编辑刘明生岗位信息

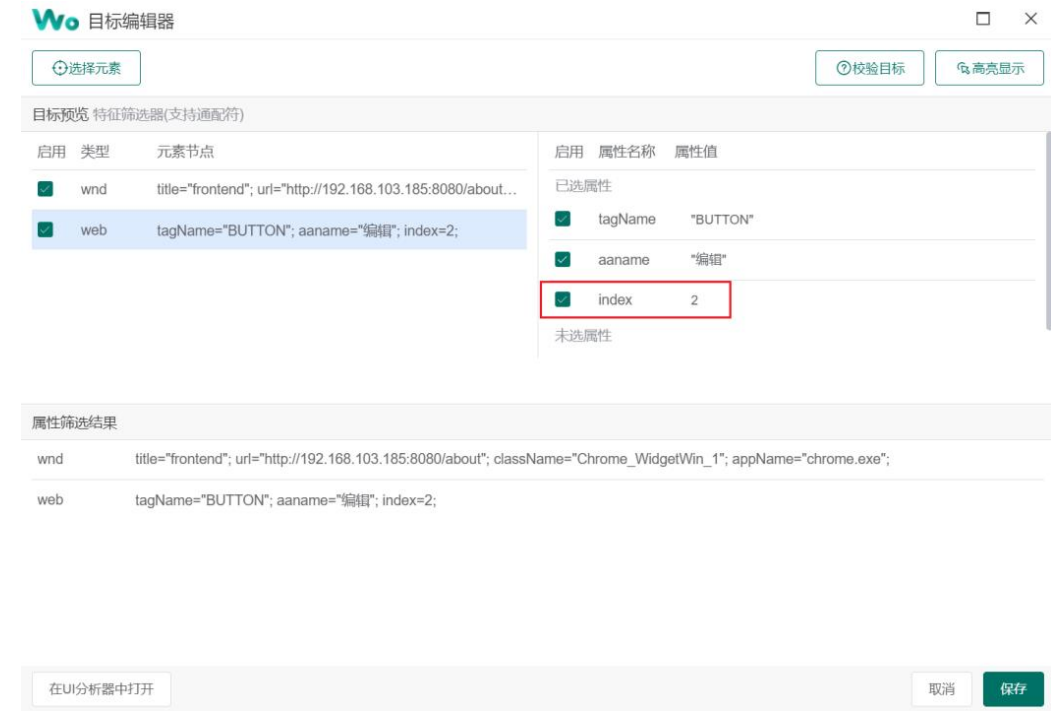


图 7 职员编辑按钮的选择器信息

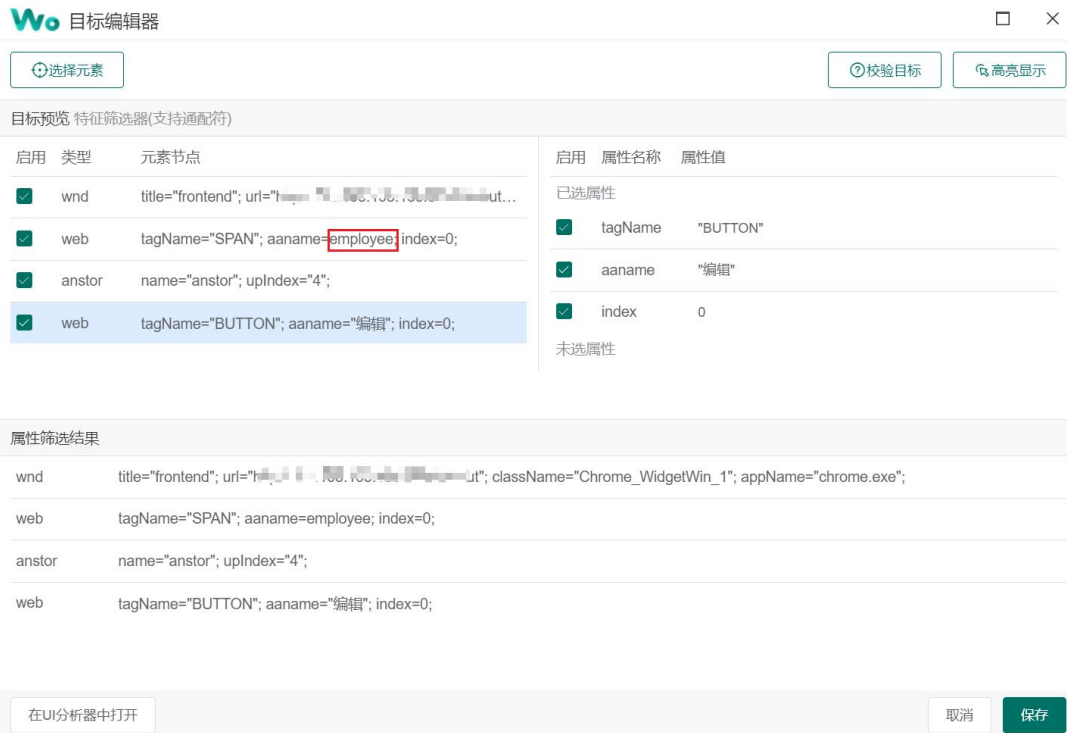


图 8 职员编辑按钮的锚点选择器信息

通过使用锚点元素，可以很好地解决数据变化引起的元素位置变化而导致的定位错误问题。在本例中加入锚点后，无论需要编辑的数据位置怎么变化，都能通过锚点元素向上 4 层的父级元素获取到预期的编辑按钮。

### 3 结语

针对使用 RPA 系统对浏览器中 DOM 元素操作时，由于数据变化引起的 DOM 元素位置发生变化，

导致 RPA 系统定位元素不准或找不到元素的情况，论文提出了一种基于锚点定位元素的方法。利用在目标元素周围寻找一个不受 DOM 位置变化影响的元素作为锚点，然后通过寻找锚点和目标元素的共同父级，最后父级元素使用相对 xpath 找到目标元素。在面对复杂变化的业务系统网页时，锚点定位法可以很好地解决大部分因为元素位置发生变化而引起的 RPA 执行失败问题。但是锚点定位方法仍然存在不

足,首先是锚点的选择,锚点元素的选择器只能是基础定位+属性过滤后能得到的唯一元素,不能包含索引信息;其次锚点定位需要依赖与目标元素的共同父级,如果选定的锚点与目标元素的“距离”太远,会造成目标元素的相对 xpath 过长甚至是全路径 xpath,从而高概率引起元素定位失败问题。

#### 参考文献

- [1] CHUONG L V,HUNG P D,DIEP V T, etal. Robotic Process Automation and Opportunities for Vietnamese Market[C].Proceedings of The 7international Conference on Computer and Communications Management. 2019:94-98.
- [2] THACKSTON R. Exploring the Use of XPath Queries for Automated Assessment of Student Web Development Projects[C].SIGITE 20: The 21st Annual Conference on Information Technology Education. 2020:255-259.
- [3] 王岩松,秦云川,蔡宇辉,等.一种基于 UIA 接口的 RPA 系统设计方法[J].计算机科学,2022(8):5.