

# Spring Bean References in Third Party...

---



[Brad Handy](#) 33 posts since

Dec 2, 2009

I have what I think is the correct way to register a `SpringBeanNamesReferenceProvider` from within my reference contributor, but the references won't resolve.

```
XmlUtil.registerXmlAttributeValueReferenceProvider(registrar, attributeNamesArray,
xmlTagFilter(NamespaceFilters.COMMON_NAMESPACE_FILTER, tagClassFilter,
>tagName"), new SpringBeanNamesReferenceProvider());
```

```
private ElementFilter xmlTagFilter(NamespaceFilter namespaceFilter, ClassFilter
tagClassFilter,
    String... tagNames) {
    TextFilter tagNameFilter = new TextFilter(tagNames);
    AndFilter tagAndClassWithinNamespace = new AndFilter(namespaceFilter,
tagClassFilter, tagNameFilter);

    return new ParentElementFilter(tagAndClassWithinNamespace, 2);
}
```

I know the `xmlTagFilter` resolves element correctly, because I'm using it with my customer reference providers, and they resolve my customer references. I just can't get the Spring Beans to resolve. I checked my Spring filesets and they appear to be configured properly.

The only thing I can think of right now, is I don't have any custom `DomElement` interfaces for the document I'm viewing. Could this be it, or is there something else I'm missing?

## Spring Bean References in Third Party...



[Brad Handy](#) *33 posts since*

Dec 2, 2009 1. **Re: Spring Bean References in Third Party Plugins** Dec 29, 2009 9:20 PM

Nevermind. I realized the references provided by the reference providers take a PsiLiteralExpression. I just rolled my own reference and reference provider, then retrieved the model for the module from the SpringManager to perform the resolutions.



[Taras Tielkes](#) *1,016 posts since*

Oct 20, 2002 2. **Re: Spring Bean References in Third Party Plugins** Dec 29, 2009 9:44 PM

Hi Brad,

For some simple cases, you can add tooling annotations to your custom bean schema for declaring an attribute value to resolve to a bean of specific type.

Code completion in IDEA supports these annotations. But given that you've implemented specific support as a plugin, I assume you already know.

-tt