

Table of Contents

With a dynamic object, we think of a dynamically created object that is available in the application without changing the application. Sometimes, it's useful to "inject" objects into your application and use such objects in your business logic. This use-case is already supported via [Object Injection](#). The missing part is that it's not automatically possible to access such objects in your application. Sure, if you put an [action](#) in your LCO and a [simple remote call](#) in your [custom application](#), it'll be easy. But it's not guaranteed that you have a custom application. It's some work to extend the standard ProjX application. So this is not something to be done last minute.

To support this use-case, we have dynamic objects available via standard ProjX API. And here is it:

```
public Object getDynamicContent(String pName);
```

The name is the important thing. By default, no dynamic objects exist, so the method will return null.

If want to use a custom dynamic object, write one, e.g.:

```
public class DynamicUserContent extends AbstractDynamicContent
{
    @Override
    public String getContentTypeName()
    {
        return "userDetails";
    }

    @Override
    public Object getDynamicContent()
    {
        String sUser = SessionContext.getCurrentSession().getUserName();

        Bean bean = new Bean();
        bean.put("FIRST_NAME", "John");
        bean.put("LAST_NAME", "Doe");
        bean.put("EMAIL", "email@domain.com");
        bean.put("PROFILE_IMAGE",
FileUtil.getContent(ResourceUtil.getResourceAsStream("/images/" +
sUser.toLowerCase() + ".png")));

        return bean;
    }
}
```

This object must be [injected](#) in your session LCO, for example via ServiceLoader. If it's injected, the API call

```
((ProjX)getApplication()).getDynamicContent("userDetails");
```

will return the Bean instance of your DynamicUserContent.

As you can see, it's super easy to inject an object and make it accessible in your application without changing your application code.

From:
<https://doc.sibvisions.com/> - **Documentation**

Permanent link:
https://doc.sibvisions.com/applications/user/dynamic_objects

Last update: **2020/07/01 17:19**

