

z2-Environment - Bug #2195

Component Resource Wrapper Resource may be made available without a handle

06.05.2025 17:19 - Henning Blohm

Status:	Feedback	Start date:	06.05.2025
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	2.10.2		
origin:			

**Description**

As observed by tracekey on May 5th.

The root cause is that the component factory provided resource is bound to the holder in ComponentResourceWrapper before setting the handle:

```
...
this.exclusiveSet(this.wrapped, ()->cf.createComponentResource(name));
Resource r = this.wrapped.get();
if (r == null) {
    throw new IllegalStateException("Component factory \"" + cfn +
    "\" did not provide component resource: " + name);
}
// NOTE: via the context implementation indirection, wrapped keeps a ref on this wrapper - which i
s
// crucial as it binds this wrappers life cycle to the wrapped resource
r.init(handle());
...
```

while in

```
private void check() {
    if (this.wrapped.get()==null && this.delegate.get()==null) {
        int i = this.invalidationCount.get();
        _loadDependencies();
        _load();
    }
    ...
}
```

we only check that the resource is put, so that the resource may be used already before the handle is assigned.

Acceptance Criteria

- The handle is set before handing out the resource.
- NOTE: There may be multiple calls to init under high concurrency and it is up the the resource implementation to protect its state from concurrency induced inconsistencies.

Implementation Hint

Change Resource like this:

```
public abstract class Resource {
    private IResourceHandle handle;

    /**
     * Called at initialization time with the resource management. At the point in time this
     */
    public final Resource withHandle(IResourceHandle c) {
        this.handle = c;
        return this;
    }
}
```

```
/**
 * Called at initialization time with the resource management. At the point in time this
 * method gets called, a handle has been associated.
 * While there will be only one instance at a time, in highly concurrent situations, this meth
od may be called more than once
 */
public void init() {}
```

and Component Resource Wrapper:

```
this.exclusiveSet(this.wrapped, ()->cf.createComponentResource(name).withHandle(handle()));
Resource r = this.wrapped.get();
if (r == null) {
    throw new IllegalStateException("Component factory \"" + cfn +
    "\" did not provide component resource: " + name);
}
r.init();
```

and similarly for other resource provisioning as to make sure that the handle is bound before the wrapper is set.

## History

### #1 - 06.05.2025 17:27 - Henning Blohm

- Description updated
- Status changed from New to In Progress
- Target version set to 2.10.2

### #2 - 07.05.2025 11:27 - Henning Blohm

- Status changed from In Progress to Feedback