

Controls ControlsThreadingModel

Controls Threading Model

Overview

What threading issues exist when using and writing controls?

Controls default to a single-threaded model – only one thread at a time will be executing code in a given control instance. This simplifies client and authoring logic, but may result in excessive contention and sub-optimal performance. Sophisticated control developers may choose to implement logic to handle multiple threads and concurrent execution.

Control Client

Client access to control instances (ie `C_.*ontrolBeans`) is always thread-safe. Generated code + infrastructure manage concurrency issues:

- Concurrent calls to operations may block (depending on control implementation and container)
- Concurrent calls to get/set properties block as necessary to maintain coherency
- Concurrent calls to other [ControlBean](#) generated methods and APIs are thread-safe.
- Event handlers are client code! Infrastructure may result in multiple event handlers being invoked concurrently, client's responsibility to ensure handlers are thread-safe.

Control Implementation

By default, control implementations delegate responsibility for thread-safety to the infrastructure, which provides a single-threaded environment for implementations. This is semantically equivalent to marking every operation and event handler method with "synchronized".

Implementations may choose to explicitly manage thread-safety issues themselves by annotating the implementation class with the `@ThreadingModel` annotation:

```
package org.apache.beehive.controls.api.bean;

public @interface ThreadingModel
{
    public enum Policy = { SINGLE_THREADED, MULTI_THREADED }
    public Policy value() default SINGLE_THREADED;
}
```

If an implementation specifies `@ThreadingModel(ThreadingModel.Policy.MULTI_THREADED)`, the infrastructure will permit multiple threads to execute concurrently on operations and event handlers. The implementation is expected to use standard Java concurrency mechanisms to guarantee data coherency.

Open Issues

- Is it necessary to talk about threading issues around client initializers / [ControlBean](#) instantiation? Seems straight-forward.