

Using BundleTracker to support the OSGi Extender Pattern

BJ Hargrave

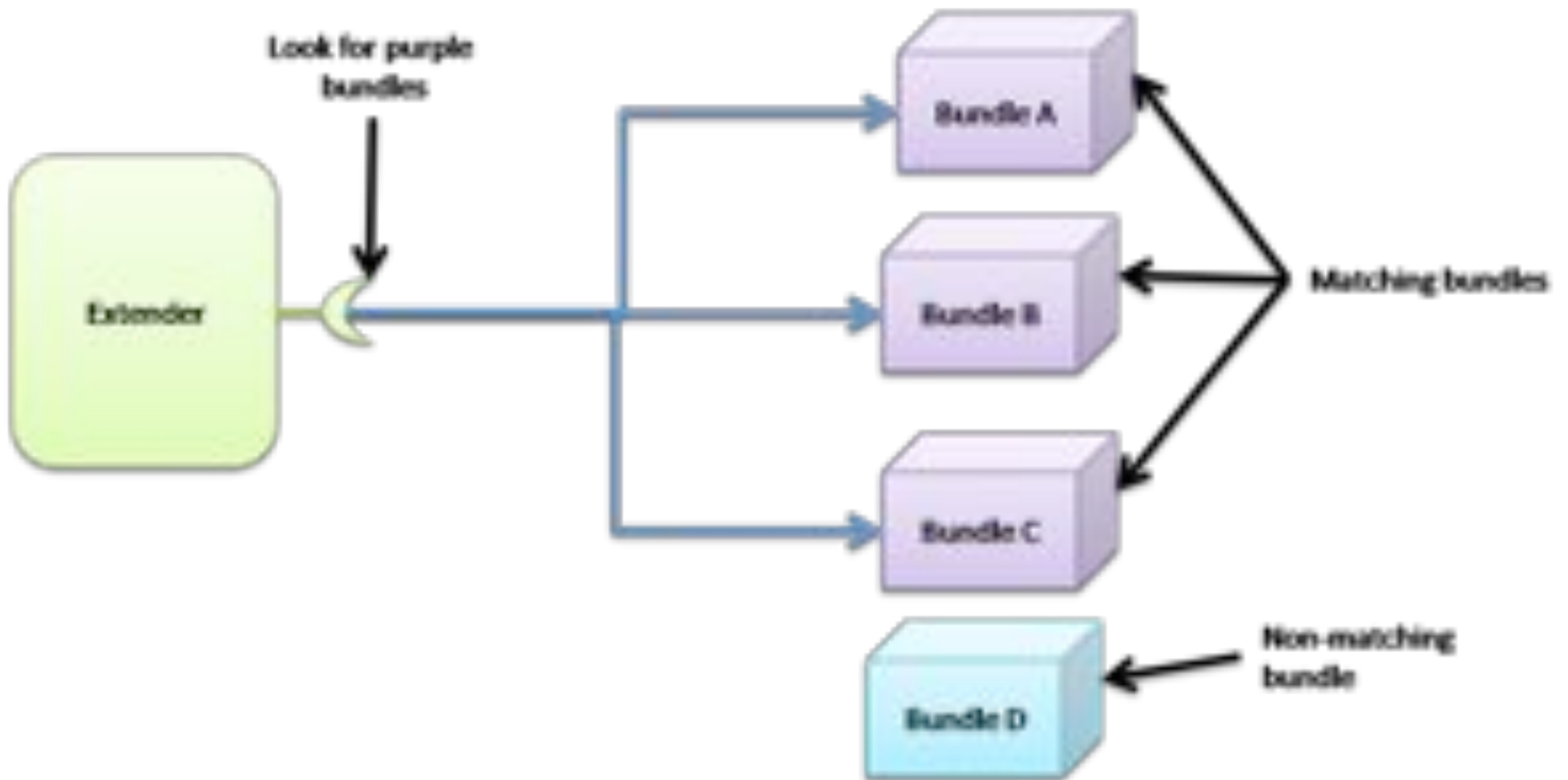
IBM Lotus

Senior Technical Staff Member





Extender Pattern





But Tracking Dynamic Things is Hard!



ServiceTracker



BundleTracker



BundleTracker(BundleContext context, int stateMask, BundleTrackerCustomizer customizer)

Create a `BundleTracker` for bundles whose state is present in the specified state mask.

Bundles whose state is present on the specified state mask will be tracked by this `BundleTracker`.

See Also:

[Bundle.getState\(\)](#)

Parameters:

context The `BundleContext` against which the tracking is done.

stateMask The bit mask of the ORing of the bundle states to be tracked.

customizer The customizer object to call when bundles are added, modified, or removed in this `BundleTracker`. If `customizer` is `null`, then this `BundleTracker` will be used as the `BundleTrackerCustomizer` and this `BundleTracker` will call the `BundleTrackerCustomizer` methods on itself.



```
public class MyBundleTrackerCustomizer implements
BundleTrackerCustomizer {
    public Object addingBundle(Bundle bundle, BundleEvent event) {
        // ...
        return someObjectToTrack;
    }
    public void modifiedBundle(Bundle bundle, BundleEvent event,
Object object) {
        // ...
    }
    public void removedBundle(Bundle bundle, BundleEvent event,
Object object) {
        // ...
    }
}
```