

# Cougaar

Cougaar Agent Architecture Open-Source site

## Generics

*Enhanced in Cougaar 12.6 are use of generics.*

We have added generics support for black board subscriptions and predicates. This reduces runtime checking of instanceof and subsequent casting.

The new API looks like:

```
public interface UnaryPredicate<T> {
    boolean execute(Object o); // can't be "(T o)!"
}
public interface BlackboardService { ...
    <T> IncrementalSubscription<T>
        subscribe(UnaryPredicate<T> u);
}
public interface Subscription<T> { ...
}
public interface IncrementalSubscription<T>
    extends Subscription<T> {
    ...
    Collection<T> getAddedCollection();
}

... MyExamplePlugin ... { ...
private IncrementalSubscription<Task> taskSub;
...
taskSub = blackboard.subscribe(new UnaryPredicate<Task>() {
    public boolean execute(Object o) {
        return (o instanceof Task);
    }
});
...
for (Task t : taskSub.getAddedCollection()) { ... }
}
```

Contrast that with the previous APIs (note the extra casts using the previous APIs):

```
public interface UnaryPredicate {
    boolean execute(Object o);
}
public interface BlackboardService { ...
    Subscription subscribe(UnaryPredicate u);
}
public interface Subscription { ...
}
public interface IncrementalSubscription extends Subscription {
    ...
    Collection getAddedCollection();
}

... MyExamplePlugin ... { ...
private IncrementalSubscription taskSub;
    ...
    taskSub = (IncrementalSubscription) blackboard.subscribe(new
UnaryPredicate() {
    public boolean execute(Object o) {
        return (o instanceof Task);
    }
});
    ...
    for (Object o : taskSub.getAddedCollection()) {
        Task t = (Task) o; ...
    }
}
```

---