DRef is a special well known type of Ref that is created by the DREF op-code, which:

* points (as a “referent”) to an actual Ref instance (e.g. FutureRef, LazyRef, a combination of them, etc.),
* delegates both “get” and “set” operations to the referent
* can only exist in a frame register, and
* cannot ever be actually visible to any other op-codes.

The rules for operations on the DRef registers are:

Assigning a value to a register “r” (invoke\*, call\* op-codes)

|  |  |  |
| --- | --- | --- |
| *Value type* | *v = ObjectHandle<T>* | *v = ProxyHandle<T>* |
| *Register Info* |  |  |
| *T* | r = v | r = v.get() |
| *DRef<T>* | ((DRef) r).set(v) | ((DRef) r).set (v) |

Getting a value from a register “r” (any op-code except REF)

|  |  |  |
| --- | --- | --- |
| *Register Info* | *T* | *DRef<T>* |
|  | v = r | v = r.get() |

REF op-code from a register “r “

|  |  |  |
| --- | --- | --- |
| *Register Info* | T | DRef<T> |
|  | v = new Ref(r) | v = r.getReferent() |