26

26.2 Realm Objects

26.2.1 The Reflect.Realm Constructor

The initial value of **`Reflect.Realm`** is the %Realm% intrinsic object. **`Reflect.Realm`** is the constructor for Realm objects. When **`Reflect.Realm`** is called as a function rather than as a constructor, it initializes its **this** value with the internal state necessary to support the **`Reflect.Realm.prototype`** built-in methods.

The **`Reflect.Realm`** constructor is designed to be subclassable. It may be used as the value in an **`extends`** clause of a class definition. Subclass constructors that intend to inherit the specified Realm behaviour must include a **`super`** call to **`Reflect.Realm`**.

26.2.1.1 Reflect.Realm ([target , handler])

When the **`Reflect.Realm`** function is called with optional arguments *target* and *handler* the following steps are taken:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. If the value of realmObject's [[RealmRecord]] internal slot is not undefined, throw a TypeError
   exception.
4. If any arguments were passed to this function, then
   a. Let newGlobal be ProxyCreate(target, handler).
   b. ReturnIfAbrupt(newGlobal)
5. Else,
   a. Let newGlobal be undefined.
6. Let realmRec be CreateRealm().
7. Perform SetRealmGlobalObj(realmRec, newGlobal).
8. Let translate be GetMethod(realmObject, "directEval").
9. ReturnIfAbrupt(translate).
10. Let fallback be GetMethod(realmObject, "nonEval").
11. ReturnIfAbrupt(fallback).
12. Let indirectEval be GetMethod(realmObject, "indirectEval").
13. ReturnIfAbrupt(indirectEval).
14. Set realmRec.[[directEvalTranslate]] to translate.
15. Set realmRec.[[nonEvalFallback]] to fallback.
16. Set realmRec.[[indirectEval]] to indirectEval.
17. NOTE the following step ensures that this function was not reentrantly applied to realmObject during
    the above steps.
18. If the value of realmObject's [[RealmRecord]] internal slot is not undefined, throw a TypeError
    exception.
19. Set realmObject's [[RealmRecord]] internal slot to realmRec.
20. Let initGlobal be GetMethod(realmObject, "initGlobal").
21. ReturnIfAbrupt(initGlobal).
22. If initGlobal is not undefined, then
    a. Let status be the result of calling the [[Call]] internal method of initGlobal, passing
       realmObject as the this value and no arguments.
    b. ReturnIfAbrupt(status).
23. Else,
    a. Let *status* be SetDefaultGlobalBindings(*realmRec*).
    b. ReturnIfAbrupt(*status*).
24. Return *realmObject*.
```

26.2.1.2 new Reflect.Realm (...argumentsList)

When `Reflect.Realm` is called as part of a `new` expression it is a constructor: it initializes a newly created object. It performs the following steps:

```
1. Let F be the %Realm% function object on which the new operator was applied.
2. Let argumentsList be the argumentsList argument of the [[Construct]] internal method that was invoked by the new operator.
3. Return the result of Construct(F, argumentsList).
```

If `Reflect.Realm` is implemented as an ECMAScript function object, its [[Construct]] internal method will perform the above steps.

26.2.2 Properties of the Reflect.Realm Constructor

The value of the [[Prototype]] internal slot of the `Reflect.Realm` constructor is the Function prototype object (**Error! Reference source not found.**).

Besides the `length` property (whose value is **0**), the `Reflect.Realm` constructor has the following properties:

26.2.2.1 Reflect.Realm.prototype

The initial value of `Reflect.Realm.prototype` is the intrinsic %RealmPrototype% object (26.2.3).

This property has the attributes { [[Writable]]: **false**, [[Enumerable]]: **false**, [[Configurable]]: **false** }.

26.2.2.2 Reflect.Realm [@@create] ( )

The @@create method of a `Reflect.Realm` function object *F* performs the following steps:

```
1. Let F be the this value.
2. Let obj be the result of calling OrdinaryCreateFromConstructor(F, "%RealmPrototype%", ( [[RealmRecord]])).
3. Return obj.
```

The value of the `name` property of this function is `"[Symbol.create]"`.

This property has the attributes { [[Writable]]: **false**, [[Enumerable]]: **false**, [[Configurable]]: **true** }.

26.2.3 Properties of the Reflect.Realm Prototype Object

The value of the [[Prototype]] internal slot of the `Reflect.Realm` prototype object is the standard built-in Object prototype object (**Error! Reference source not found.**). The `Reflect.Realm` prototype object is an ordinary object. It does not have a [[RealmRecord]] internal slot.

26.2.3.1 Reflect.Realm.prototype.constructor

The initial value of `Reflect.Realm.prototype.constructor` is the built-in %Realm% constructor.

26.2.3.2 Reflect.Realm.prototype.eval (source)

When `Reflect.Realm.prototype.eval` is called with argument *source* it performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realm be the value of realmObject's [[RealmRecord]] internal slot.
4. If realm is undefined, then throw a TypeError exception.
5. Return the result of IndirectEval(realm, source).
```

Rev 25 Allen Wirfs-B…, 8/9/2014 9:25 AM
Comment [1]: TODO

26.2.3.3 get Reflect.Realm.prototype.global

`Reflect.Realm.prototype.global` is an accessor property whose set accessor function is **undefined**. Its get accessor function performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realm be the value of realmObject's [[RealmRecord]] internal slot.
4. If realm is undefined, then throw a TypeError exception.
5. Return realm.[[globalThis]].
```

26.2.3.4 get Reflect.Realm.prototype.intrinsics

`Reflect.Realm.prototype.intrinsics` is an accessor property whose set accessor function is **undefined**. Its get accessor function performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realm be the value of realmObject's [[RealmRecord]] internal slot.
4. If realm is undefined, then throw a TypeError exception.
5. Let table be ObjectCreate(%ObjectPrototype%).
6. Let intrinsics be realm's [[intrinsics]] internal slot.
7. For each name in the "Intrinsic Key" column of  Error! Reference source not found., in row order
   do
    a. Let object be the value of the field of intrinsics whose name is name.
    b. Perform CreateDataProperty(table, key, object).
8. Return table.
```

26.2.3.5 get Reflect.Realm.prototype.stdlib

`Reflect.Realm.prototype.stdlib` is an accessor property whose set accessor function is **undefined**. Its get accessor function performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realm be the value of realmObject's [[RealmRecord]] internal slot.
4. If realm is undefined, then throw a TypeError exception.
5. Let props be ObjectCreate(%ObjectPrototype%).
6. For each property of the Global Object specified in clause Error! Reference source not found., do
    a. Let name be the string value of the property name.
    b. Let desc be the fully populated data property descriptor for the property containing the
       specified attributes for the property. For properties whose values are functions, the value of the
       [[Value]] attribute is the corresponding intrinsic function object for realm.
    c. Let status be DefinePropertyOrThrow(props, name, desc).
    d. ReturnIfAbrupt(status).
7. Return props.
```

NOTE The object returned is suitable for use as the second argument to Object.defineProperties. A Realm's global object can be initialized with its clause **Error! Reference source not found.** standard values using an expression such as:

```
Object.defineProperties(newRealm.global, newRealm.stdlib);
```

26.2.3.6 Reflect.Realm.prototype [@@toStringTag]

The initial value of the @@toStringTag property is the string value `"Reflect.Realm"`.

This property has the attributes { [[Writable]]: **false**, [[Enumerable]]: **false**, [[Configurable]]: **true** }.

26.2.3.7 Realm Subclass Extension Properties

The following properties are intended to be over-ridden by subclasses of `Reflect.Realm`.

26.2.3.7.1 Reflect.Realm.prototype.directEval (source)

When `Reflect.Realm.prototype.directEval` is called with argument *source* it performs the following steps:

```
1. Return source.
```

NOTE If an apparent direct eval call had multiple arguments, those arguments are all passed to this function.

26.2.3.7.2 Reflect.Realm.prototype.indirectEval (source)

When `Reflect.Realm.prototype.indirectEval` is called with argument *source* it performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realm be the value of realmObject's [[RealmRecord]] internal slot.
4. If realm is undefined, then throw a TypeError exception.
5. Return IndirectEval(realm, source).
```

Rev 23 Allen Wirfs-B…, 8/9/2014 9:25 AM
Comment [2]: TODO

26.2.3.7.3 Reflect.Realm.prototype.initGlobal ( )

When `Reflect.Realm.prototype.initGlobal` is called it performs the following steps:

```
1. Let realmObject be the this value.
2. If Type(realmObject) is not Object or realmObject does not have a [[RealmRecord]] internal slot,
   throw a TypeError exception.
3. Let realmRec be the value of realmObject's [[RealmRecord]] internal slot.
4. If realmRec is undefined, then throw a TypeError exception.
5. Return  SetDefaultGlobalBindings(realmRec).
```

26.2.3.7.4 Reflect.Realm.prototype.nonEval (function, thisValue, argumentsList)

When **`Reflect.Realm.prototype.nonEval`** is called with arguments *function*, *thisValue*, and *argumentsList* it performs the following steps:

1. If IsCallable(*function*) is **false**, then throw a **TypeError** exception.
2. Let *args* be CreateListFromArrayLike(*argumentsList*).
3. ReturnIfAbrupt(*args*).
4. Perform PrepareForTailCall( ).
5. Return the result of calling the [[Call]] internal method of *function*with arguments *thisValue* and *args*.

26.2.4 Properties of Reflect.Realm Instances

`Reflect.Realm` instances are ordinary objects that inherit properties from the **`Reflect.Realm`** prototype object. **`Reflect.Realm`** instances each have a [[RealmRecord]] internal slot.

26.3 Loader Object

26.3.3 Properties of the Reflect.Loader Prototype Object

26.3.3.13 get Reflect.Loader.prototype.realm

`Reflect.Loader.prototype.realm` is an accessor property whose set accessor function is **undefined**. Its get accessor function performs the following steps:

1. Let *loader* be this Loader.
2. ReturnIfAbrupt(*loader*).
3. Let *loaderRecord* be *loader's* [[LoaderRecord]] internal slot.
4. Return RealmObjectFor(*loaderRecord*.[[Realm]]).

Rev 22 Allen Wirfs-…, 8/30/2014 4:53 PM
Comment [3]: TODO: need to define. Lazily create Realm objects?