

# BIBFRAME 2.0 Specifications

## Notes

June 2, 2016  
Library of Congress

*This is one of a set of specifications for BIBFRAME 2.0.*

### Background: Notes in 1.0

In BIBFRAME 1.0, notes were supplied via property (or subproperty of) `bf:note`, as in this example.

```
bf:creditsNote "Produced by John Andrew Schreiner" .
```

`bf:creditsNote` is a subproperty of `bf:note`.

### Changes in 2.0

- All subproperties of `bf:note` (`bf:creditsNote`, `bf:contentNote`, etc.) are eliminated.
- The general note property, `bf:note` is retained, but redefined as an object property.
- Class `bf>Note` is defined. It is the range of `bf:note`. (Note content is supplied via `rdfs:label` as a property of `bf>Note`.)

### Expressing Notes and their Types in 2.0

There are five categories of notes in 2.0.

1. Informal note.
2. A note with no type expressed.
3. A note with type implied. (Special case of 2.)
4. A note whose type is expressed as a literal.
5. A note whose type is expressed by an external class.

### Informal Note

There are several new BIBFRAME 2.0 properties that are informally considered to be notes, but are not formal notes, that is, are not expressed via `bf:note` and the `Note` resource, and “note” is not part of the property name. For example the property `bf:credits` is introduced, and the above credits note would be expressed as:

*Example 1.*

```
bf:credits "Produced by John Andrew Schreiner" .
```

Although `bf:credits` in 2.0 essentially replaces 1.0 property `bf:creditsNote`, it is not a subproperty of `bf:note`. It is a simple datatype property expressing a note.

## Note with No Type Expressed

The following example expresses a note, but does not express a note type.

*Example 2.*

```
bf:note [ a          bf:Note ;
          rdfs:label  "Late 16th century blind-tooled binding, dark brown calf" ] .
```

In contrast to the above example this is a BIBFRAME note, expressed via property `bf:note` and class `bf:Note`.

## Note with Type Implied

The following example does not express an explicit note type. The type may be inferred from the context.

*Example 3.*

```
bf:baseMaterial [
  bf:code [ rdf:value      "o" ;
            bf:source [rdf:value  "marc007ng04" ] ];
  bf:note [ a          bf:Note ;
            rdfs:label  "Image printed on thick gold paper." ] ] .
```

The implied note type is "base material".

## Note Type Expressed as Literal

In the following example, the same note as in example 2 is expressed, and in addition, a note type, "binding".

*Example 4.*

```
bf:note [ a          bf:Note ;
          bf:noteType "binding" ;
          rdfs:label  "Late 16th century blind-tooled binding, dark brown calf" ] .
```

The property `bf:noteType` is defined specifically for, and should be used only, when there is no known class for the note type.

## Note Type Expressed by External Class

In the following example, the note type is expressed via external namespace "ex:" which defines note type `ex:FundingInformation`

*Example 5.*

```
bf:note [ a          ex:FundingInformation, bf:Note ;
          rdfs:label  "Sponsored by the U.S. Air Force" ] .
```

BIBFRAME itself does not define any note types (other than generic `bf:Note`).