Plugin file format - Optional element reference

Table of contents

1 Optional element reference....3
2 Element export....3
2.1 Attribute id....3
2.2 Attribute library....3
2.3 Attribute symbol....3
3 Element import....3
3.1 Attribute id....3
3.2 Attribute library....4
3.3 Attribute symbol....4
4 Element extension-point....4
4.1 Attribute id....4
4.2 Attribute schema....4
5 Element extend....4
5.1 Attribute library....4
5.2 Attribute symbol....4
6 Element extension....4
6.1 Attribute point....5
7 Element event....5
7.1 Attribute id....5
7.2 Attribute signature....5
8 Element event-handler....5
8.1 Attribute event....5

Copyright © 2008 Hugh Leather All rights reserved.

8.2 Attribute listener....5
9 Element around....6
9.1 Attribute id....6
9.2 Attribute signature....6
10 Element join-point....6
10.1 Attribute id....6
10.2 Attribute signature....6
11 Element call....6
11.1 Attribute library....6
11.2 Attribute symbol....7
12 Element callback....7
12.1 Attribute library....7
12.2 Attribute symbol....7
12.3 Attribute import....7
12.4 Attribute type....7

Copyright © 2008 Hugh Leather All rights reserved.

1. Optional element reference

Plugin applications can decide which elements can be allowed beneath a plugin element (except for the fixed set which are always included). The plugin system provides a variety of optional elements, found below. It is expected that most applications will provide all of these elements. Plugin applications may also add their own, new elements.

Check with the plugin application documentation to see which elements it allows.

2. Element export

```
<export
    id        = extension point id
    library?  = library id
    symbol    = symbol name>
    <!-- Content: EMPTY -->
</export>
<!-- Contained by: plugin -->
```

2.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

2.2. Attribute library

This attribute is variable expanded. Variables are resolved in the context of the plugin.

2.3. Attribute symbol

This attribute is variable expanded. Variables are resolved in the context of the plugin.

3. Element import

```
<import
    id        = extension point id
    library?  = library id
    symbol    = symbol name>
    <!-- Content: EMPTY -->
</import>
<!-- Contained by: plugin -->
```

3.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

Copyright © 2008 Hugh Leather All rights reserved.

3.2. Attribute library

This attribute is variable expanded. Variables are resolved in the context of the plugin.

3.3. Attribute symbol

This attribute is variable expanded. Variables are resolved in the context of the plugin.

4. Element extension-point

```
<extension-point
    id        = extension point id
    schema?   = xsd file name>
    <!-- Content: extend? -->
</extension-point>
<!-- Contained by: plugin -->
```

4.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

4.2. Attribute schema

This attribute is variable expanded. Variables are resolved in the context of the plugin.

5. Element extend

```
<extend
    library?  = library id
    symbol    = symbol name>
    <!-- Content: EMPTY -->
</extend>
<!-- Contained by: extension-point -->
```

5.1. Attribute library

This attribute is variable expanded. Variables are resolved in the context of the plugin.

5.2. Attribute symbol

This attribute is variable expanded. Variables are resolved in the context of the plugin.

6. Element extension

Copyright © 2008 Hugh Leather All rights reserved.

```
<extension
    point     = extension point id
    *         = any attributes allowed>
    <!-- Content: ANY -->
</extension>
<!-- Contained by: plugin -->
```

6.1. Attribute point

This attribute is variable expanded. Variables are resolved in the context of the plugin.

7. Element event

```
<event
    id        = extension point id
    signature = signature>
    <!-- Content: call -->
</event>
<!-- Contained by: plugin -->
```

7.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

7.2. Attribute signature

This attribute is variable expanded. Variables are resolved in the context of the plugin.

8. Element event-handler

```
<event-handler
    event     = extension point id
    listener  = extension point id
    *         = any attributes allowed>
    <!-- Content: ANY -->
</event-handler>
<!-- Contained by: plugin -->
```

8.1. Attribute event

This attribute is variable expanded. Variables are resolved in the context of the plugin.

8.2. Attribute listener

This attribute is variable expanded. Variables are resolved in the context of the plugin.

Copyright © 2008 Hugh Leather All rights reserved.

9. Element around

```
<around
    id        = extension point id
    signature = signature>
    <!-- Content: ( call | callback ) -->
</around>
<!-- Contained by: plugin -->
```

9.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

9.2. Attribute signature

This attribute is variable expanded. Variables are resolved in the context of the plugin.

10. Element join-point

```
<join-point
    id        = extension point id
    signature = signature>
    <!-- Content: ( call | callback ) -->
</join-point>
<!-- Contained by: plugin -->
```

10.1. Attribute id

This attribute is variable expanded. Variables are resolved in the context of the plugin.

10.2. Attribute signature

This attribute is variable expanded. Variables are resolved in the context of the plugin.

11. Element call

```
<call
    library?  = library id
    symbol    = symbol name>
    <!-- Content: EMPTY -->
</call>
<!-- Contained by: event, around, join-point -->
```

11.1. Attribute library

Copyright © 2008 Hugh Leather All rights reserved.

This attribute is variable expanded. Variables are resolved in the context of the plugin.

11.2. Attribute symbol

This attribute is variable expanded. Variables are resolved in the context of the plugin.

12. Element callback

```
<callback
    library?  = library id
    symbol    = symbol name
    import?   = extension point id
    type?     = { "static" | "non-static" | "ffi" } default "static or
ffi">
    <!-- Content: EMPTY -->
</callback>
<!-- Contained by: around, join-point -->
```

12.1. Attribute library

This attribute is variable expanded. Variables are resolved in the context of the plugin.

12.2. Attribute symbol

This attribute is variable expanded. Variables are resolved in the context of the plugin.

12.3. Attribute import

This attribute is variable expanded. Variables are resolved in the context of the plugin.

12.4. Attribute type

This attribute is variable expanded. Variables are resolved in the context of the plugin.

Copyright © 2008 Hugh Leather All rights reserved.