

1 5.1. Element <SubjectMatch>

2 The <SubjectMatch> element SHALL identify a set of *subject*-related entities by matching
3 *attribute* values in the <xacml-context:Subject> element of the *context* with the embedded
4 *attribute* value.

5

```
6        <xs:element name="SubjectMatch" type="xacml:SubjectMatchType"/>  
7        <xs:complexType name="SubjectMatchType">  
8           <xs:sequence>  
9             <xs:choice>  
10               <xs:element ref="xacml:QualifiedSubjectAttributeDesignator"/>  
11               <xs:element ref="xacml:AttributeSelector"/>  
12             </xs:choice>  
13             <xs:element ref="xacml:AttributeValue"/>  
14           </xs:sequence>  
15           <xs:attribute name="MatchId" type="xs:anyURI" use="required"/>  
16        </xs:complexType>
```

17

18 The <SubjectMatch> element is of **SubjectMatchType** complex type.

19 The <SubjectMatch> element contains the following attributes and elements:

20 MatchId [Required]

21 This attribute specifies a matching function. The value of this attribute MUST be of type
22 xs:anyURI, with legal values documented in Appendix A.

23 <QualifiedSubjectAttributeDesignator> [Required choice]

24 This element MAY be used to identify one or more *attribute* values in the <xacml-
25 context:Subject> child elements of the <xacml-context:Request> element.

26 <AttributeSelector> [Required choice]

27 This element MAY be used to identify one or more attribute values in the <xacml-
28 context:Subject> child elements of the <xacml-context:Request> element.

29 <AttributeValue> [Required]

30 This element contains the embedded *attribute* value.

31 5.2. Complex type AttributeDesignatorType

32 The **AttributeDesignatorType** complex type is the base type for elements and extensions that
33 refer to *named attributes*. A *named attribute* has specific criteria with which to match attributes
34 within a specific part of the <xacml-context:Request> element. The **AttributeDesignatorType**
35 complex type specifies the attributes used for the match criteria that are common to all *named*
36 *attributes*. Elements and extensions of the **AttributeDesignatorType** complex type MAY
37 determine the presence of *named attributes* or select *attribute values* associated with *attributes*
38 that match *named attributes*. Elements and extensions of the **AttributeDesignatorType** SHALL
39 only enhance the match semantics.

40

```

41 <xs:complexType name="AttributeDesignatorType">
42   <xs:attribute name="AttributeId" type="xs:anyURI" use="required" />
43   <xs:attribute name="DataType" type="xs:anyURI" use="required" />
44   <xs:attribute name="Issuer" type="xs:anyURI" use="optional" />
45   <xs:attribute name="MustBePresent" type="xs:boolean" use="optional" />
46 </xs:complexType>

```

47

48 A **named attribute** SHALL match an **attribute** if the values of their respective `AttributeId`,
49 `DataType`, and `Issuer` attributes match. The `AttributeId` attribute MUST match, by URI
50 equality, that of the `AttributeId` attribute of the **attribute**. The `DataType` attribute MUST match,
51 by URI equality, that of the `DataType` attribute of the same **attribute**. If the `Issuer` attribute is
52 supplied, it MUST match, by URI equality, the `Issuer` attribute of the same **attribute**. If the
53 `Issuer` attribute is not supplied, the matching of the **named attribute** SHALL be governed by
54 `AttributeId` and `DataType` attributes alone, regardless of the presence, absence, or actual
55 value of the `Issuer` attribute of the otherwise matching **attribute**.

56 The `<AttributeDesignatorType>` contains the following attributes:

57 `AttributeId` [Required]

58 This attribute SHALL specify the `AttributeId` with which to match the **attribute**.

59 `DataType` [Required]

60 This attribute SHALL specify the `DataType` with which to match the **attribute**.

61 `Issuer` [Optional]

62 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **attribute**.

63 `MustBePresent` [Optional]

64 This attribute governs the whether the element returns **indeterminate** in the case of the
65 absence of the **named attribute**. If the **named attribute** is absent and `MustBePresent` is
66 set to `true`, this element SHALL result in **indeterminate**. If `MustBePresent` is not
67 supplied, its default value SHALL be `false`

68 5.3. Complex type SubjectQualifierType

69 The **SubjectQualifierType** complex type that extends the **AttributeDesignatorType** complex type.
70 It is the base type for elements and extensions that specify **qualified subjects**. A **subject** is a
71 `<Subject>` element of the `<Subjects>` element of the `<xacml-context:Request>` element. A
72 **subject** contains a grouping of **subject attributes**. A **named subject attribute** is a **named**
73 **attribute** from any and all **subjects**. The **SubjectQualifierType** complex type extends the
74 **AttributeDesignatorType** with a `MatchId` attribute and an `<AttributeValue>` element. The
75 **SubjectQualifierType** extends the match semantics of the **AttributeDesignatorType** such that it
76 SHALL select only the **subjects**, i.e. the **qualified subjects**, for which the **subject attributes** for
77 the **named subject attribute** are present and their **attribute values** match the value contained in
78 the `<AttributeValue>` element. Elements and extensions of the **SubjectQualifierType** complex
79 type MAY select or determine the presence of **subjects** that have **subject attributes** that match
80 the **attribute values** of **named subject attributes**. Elements and extensions of the
81 **SubjectQualifierType** SHALL only enhance the match semantics.

82

```

83 <xs:complexType name="SubjectQualifierType">

```

```

84 <xs:complexContent>
85 <xs:extension base="xacml:AttributeDesignatorType">
86 <xs:attribute name="MatchId"
87             type="xs:anyURI"
88             use="required" />
89 <xs:element ref="xacml:AttributeValue" />
90 </xs:extension>
91 </xs:complexContent>
92 </xs:complexType>

```

93

94 The <SubjectQualifierType> complex type contains the following attributes and elements in
95 addition to the attributes of the **AttributeDesignatorType** complex type:

96 MatchId [Required]

97 This attribute SHALL specify the function with which to compare *attribute values*.

98 AttributeValue [Required]

99 This element SHALL specify the attribute value with which to *match named subject*
100 *attributes*.

101 5.4. Element <SubjectQualifier>

102 The <SubjectQualifier> element of the **SubjectQualifierType** and follows it semantics which
103 are specified in Section 5.3.

104 5.5. Complex type SubjectAttributeDesignatorType

105 The **SubjectAttributeDesignatorType** complex type that extends the **AttributeDesignatorType**
106 complex type. It is the base type for elements and extensions that refer to *named subject*
107 *attributes*. A *subject attribute* is an *attribute* that SHALL only be located within the <Subject>
108 elements of the <Subjects> element the <xacml-context:Request> element. A *named subject*
109 *attribute* is a *named attribute* from any and all <Subject> elements. The
110 **SubjectAttributeDesignatorType** complex type extends the **AttributeDesignatorType** with a
111 **SubjectQualifier** elements. The **SubjectAttributeDesignatorType** extends the match
112 semantics of the **AttributeDesignatorType** such that the attribute search space SHALL be
113 restricted *qualified subjects*. A *qualified subject* is a <Subject> element that match all of the
114 contained **SubjectQualifier** elements. Elements and extensions of the
115 **SubjectAttributeDesignatorType** complex type MAY determine the presence of *named subject*
116 *attributes* or select *attribute values* associated with *subject attributes* that match *named*
117 *subject attributes*. Elements and extensions of the **SubjectAttributeDesignatorType** SHALL only
118 enhance the match semantics.

119

```

120 <xs:complexType name="SubjectAttributeDesignatorType">
121 <xs:complexContent>
122 <xs:extension base="xacml:AttributeDesignatorType">
123 <xs:element ref="SubjectQualifier"
124             minOccurs="0"
125             maxOccurs="unbounded" />
126 </xs:extension>
127 </xs:complexContent>
128 </xs:complexType>

```

129

130 The `<SubjectAttributeDesignatorType>` complex type contains the following elements in
131 addition to the attributes of the **AttributeDesignatorType** complex type:

132 `SubjectQualifier` [Optional]

133 These elements SHALL restrict the attribute search space to *qualified subjects*.

134 5.6. Complex type **QualifiedSubjectAttributeDesignatorType**

135 The **QualifiedSubjectAttributeDesignatorType** complex type that extends the
136 **SubjectAttributeDesignatorType** complex type. The **QualifiedSubjectAttributeDesignatorType**
137 complex type extends the **SubjectAttributeDesignatorType** with a `FromSubjects` attribute. The
138 **QualifiedSubjectAttributeDesignatorType** extends the match semantics of the
139 **SubjectAttributeDesignatorType** such that it matches all *subject attribute values* from all
140 *qualified subjects*. If the `FromSubjects` attribute is set to `single-subject`, (its default value) and
141 there are more than one *qualified subjects*, this element results in indeterminate. Elements and
142 extensions of the **SubjectAttributeDesignatorType** complex type select *attribute values*
143 associated with *subject attributes* that match *named subject attributes* from *qualified subjects*.
144 Elements and extensions of the **QualifiedSubjectAttributeDesignatorType** SHALL only enhance
145 the match semantics.

146

```
147 <xs:complexType name="QualifiedSubjectAttributeDesignatorType">  
148   <xs:complexContent>  
149     <xs:extension base="xacml:SubjectAttributeDesignatorType">  
150       <xs:attribute name="FromSubjects"  
151                   type="xs:string"  
152                   use="optional"  
153                   default="single-subject" />  
154     </xs:extension>  
155   </xs:complexContent>  
156 </xs:complexType>
```

157

158 The `<QualifiedSubjectAttributeDesignatorType>` complex type contains the following
159 attribute in addition to the attributes of the **SubjectAttributeDesignatorType** complex type:

160 `FromSubjects` [Optional]

161 This attribute SHALL specify whether named attributes match subject attributes from
162 multiple *qualified subjects*. If `FromSubjects` is not supplied, its default value SHALL be
163 `single-subject`. Its alternate valid value SHALL be `multiple-subjects`.

164

165 5.7. Complex type **QualifiedSubjectAttributesPresentType**

166 The **QualifiedSubjectAttributesPresentType** complex type that extends the
167 **SubjectAttributeDesignatorType** complex type. The **QualifiedSubjectAttributesPresentType**
168 complex type extends the **SubjectAttributeDesignatorType** with a `MustBeSingleSubject`
169 attribute. The **QualifiedSubjectAttributesPresentType** extends the match semantics of the
170 **SubjectAttributeDesignatorType** such that it matches *subject attribute values* from all the
171 *qualified subjects*. If the `MustBeSingleSubject` attribute is set to `true`, (its default value) and there
172 are more than one *qualified subjects*, this element results in *indeterminate*. Elements and
173 extensions of the **SubjectAttributesPresentType** complex type determine the presence of
174 *attribute values* associated with *subject attributes* that match *named subject attributes* from

175 **qualified subjects.** Elements and extensions of the **QualifiedSubjectAttributeIsPresentType**
176 SHALL only enhance the match semantics.

177

```
178 <xs:complexType name="QualifiedSubjectAttributeIsPresentType">  
179   <xs:complexContent>  
180     <xs:extension base="xacml:SubjectAttributeDesignatorType">  
181       <xs:attribute name="MustBeSingleSubject "  
182                 type="xs:boolean"  
183                 use="optional"  
184                 default="true" />  
185     </xs:extension>  
186   </xs:complexContent>  
187 </xs:complexType>
```

188

189 The <QualifiedSubjectAttributeIsPresentType> complex type contains the following
190 attribute in addition to the attributes of the **SubjectAttributeDesignatorType** complex type:

191 MustBeSingleSubject [Optional]

192 This attribute SHALL specify whether **named subject attributes** match **subject attributes**
193 from a single **qualified subject**. If MustBeSingleSubject is not supplied, its default
194 value SHALL be true.

195 **5.8. Element <QualifiedSubjectAttributeIsPresent>**

196 The <QualifiedSubjectAttributeIsPresent> element determines whether a **named subject attribute**
197 is present or not. Its match semantics are that of the **QualifiedSubjectAttributeIsPresentType**.

198 The <QualifiedSubjectAttributeIsPresent> element SHALL result in **true** if its **named**
199 **subject attribute** is present. If the MustBeSingleSubject attribute is set to true (its default
200 value), a result of **true** SHALL mean that a <QualifiedSubjectAttributeDesignator>
201 element with its FromSubject attribute set to single-subject (its default value) for this **named**
202 **subject attribute** SHALL return a bag consisting of at least one **attribute value**. The
203 MustBePresent attribute governs whether this element returns **false** or **indeterminate** in the case
204 of the absence of the **named subject attribute**. If the **named subject attribute** is not present and
205 the MustBePresent attribute is set to false (its default value) this element SHALL result in **false**.
206 If the **named resource attribute** is not present and the MustBePresent attribute is set to true,
207 this element SHALL result in **indeterminate**. Regardless of the MustBePresent attribute, if it
208 cannot be determined whether the **named subject attribute** is present or not present in the
209 request context, or the value of the **named subject attribute** is unavailable, then the expression
210 SHALL result in **indeterminate**.

211 A **named subject attribute** SHALL be considered present if at least one **subject attribute** exists
212 that matches the values of its corresponding AttributeId, DataType, and Issuer attributes as
213 well as the **SubjectQualifier** elements as per the match semantics of the
214 **QualifiedSubjectAttributeIsPresentType** specification [Section 5.7].

215 The <QualifiedSubjectAttributeIsPresent> MAY be passed to the <Apply> element as an
216 argument.

217

```
218 <xs:element name="QualifiedSubjectAttributeIsPresent "  
219           type="xacml:SubjectAttributeDesignatorType" />
```

220

221 The <QualifiedAttributeIsPresent> element has the following attributes:

222 AttributeId [Required]

223 This attribute SHALL specify the AttributeId with which to match the **subject attribute**
 224 of **qualified subjects**.

225 DataType [Required]

226 This attribute SHALL specify the DataType with which to match the **subject attribute** of
 227 **qualified subjects**.

228 Issuer [Optional]

229 This attribute, if supplied, SHALL specify the Issuer with which to match the **subject**
 230 **attribute** of **qualified subjects**.

231 MustBePresent [Optional]

232 This attribute governs the whether the <QualifiedSubjectAttributesPresent> element
 233 returns **false** or **indeterminate** in the case of the absence of the **named subject attribute**.
 234 If the **named subject attribute** is absent and MustBePresent is set to **false**, this
 235 element SHALL result in **false**. If the **named subject attribute** is absent and
 236 MustBePresent is set to **true**, this element SHALL result in **indeterminate**. If
 237 MustBePresent is not supplied, its default value SHALL be **false**.

238 MustBeSingleSubject [Optional]

239 This attribute governs the whether the <QualifiedSubjectAttributesPresent> element
 240 returns **true** or **false** in the case of the presence of the **named subject attribute** within
 241 multiple **qualified subjects**. If MustBeSingleSubject is set to **true** and there are zero
 242 or more than one **qualified subject**, this element SHALL result in **false**. Otherwise, this
 243 element SHALL result in the presence of the **named subject attribute** in the **qualified**
 244 **subjects**. If MustBePresent is not supplied, its default value SHALL be **false**.

245 **5.9. Element <QualifiedSubjectAttributeDesignator>**

246 The <QualifiedSubjectAttributeDesignator> element selects attribute values that match **named**
 247 **subject attribute**. Its match semantics are that of the **QualifiedSubjectAttributeDesignatorType**.

248 The <QualifiedSubjectAttributeDesignator> element SHALL return a bag of all the
 249 **subject attribute values** that are matched by the **named subject attribute** from **qualified**
 250 **subjects**. The FromSubjects attribute governs whether the matched attributes come are merged
 251 from multiple **qualified subjects** or are required to come from a single **qualified subject**. If the
 252 FromSubjects attribute is set to **single-subject** (its default value) and there are multiple
 253 **qualified subjects**, this element results in **indeterminate**. The MustBePresent attribute governs
 254 whether this element returns an empty bag or **indeterminate** in the case of the absence of the
 255 **named subject attribute**. If the **named subject attribute** is not present and the MustBePresent
 256 attribute is set to **false** (its default value) this element SHALL result in an empty bag. If the **named**
 257 **subject attribute** is not present and the MustBePresent attribute is set to **true**, this element
 258 SHALL result in **indeterminate**. Regardless of the MustBePresent attribute, if it cannot be
 259 determined whether the **named subject attribute** is present or not present in the request context,
 260 or the value of the **named subject attribute** is unavailable, then the expression SHALL result in
 261 **indeterminate**.

262 A **named subject attribute** SHALL match a **subject attribute** as per the match semantics
263 specified in the **QualifiedSubjectAttributeDesignatorType** complex type [Section 5.6].

264 The <QualifiedSubjectAttributeDesignator> MAY appear in a <SubjectMatch>
265 element and MAY be passed to the <Apply> element as an argument.

266

```
267 <xs:element name="QualifiedSubjectAttributeIsPresent "  
268 type="xacml:SubjectAttributeDesignatorType" />
```

269

270 The <QualifiedSubjectAttributeDesignator> element is of the
271 **QualifiedSubjectAttributeDesignatorType** complex type.

272 The <QualifiedSubjectAttributeDesignator> element has the following attributes:

273 AttributeId [Required]

274 This attribute SHALL specify the AttributeId with which to match the **subject attribute**
275 within **qualified subjects**.

276 DataType [Required]

277 This attribute SHALL specify the DataType with which to match the **subject attribute** within
278 **qualified subjects**.

279 Issuer [Optional]

280 This attribute, if supplied, SHALL specify the Issuer with which to match the **subject**
281 **attribute** within **qualified subjects**.

282 MustBePresent [Optional]

283 This attribute governs the whether the <QualifiedSubjectAttributeDesignator>
284 element returns an empty bag or **indeterminate** in the case of the absence of the **named**
285 **subject attribute**. If the **named subject attribute** is absent and MustBePresent is set to
286 false, this element SHALL result in an empty bag. If the **named subject attribute** is
287 absent and MustBePresent is set to true, this element SHALL result in **indeterminate**.
288 If MustBePresent is not supplied, its default value SHALL be false.

289 FromSubjects [Optional]

290 This attribute governs the whether the <QualifiedSubjectAttributeDesignator>
291 element returns subject attribute values from a single or multiple **qualified subjects**. If
292 FromSubjects is set to single-subject and there are more than one **qualified**
293 **subject**, then this element SHALL result in **indeterminate**. Otherwise, this element returns
294 a bag consisting of the **subject attribute values** matching the **named subject attributes**
295 of all the **qualified subjects**. If FromSubjects is not supplied, its default value SHALL be
296 single-subject. Its alternate valid value is multiple-subjects.

297 **5.10. Element <ResourceAttributeIsPresent>**

298

299 The <ResourceAttributeIsPresent> element determines whether a **named resource**
300 **attribute** is present. A **resource attribute** is an **attribute** that SHALL only be located within the
301 <Resource> element of the <xacml-context:Request> element. A **named resource**
302 **attribute** has specific criteria (described below) with which to match a **resource attribute**. A

303 **named resource attribute** SHALL be considered *present*, i.e. *not absent*, if there is at least one
304 **resource attribute** that matches the criteria.

305 The <ResourceAttributeIsPresent> element SHALL result in **true** if its **named resource**
306 **attribute** is present. A result of **true** SHALL mean that a <ResourceAttributeDesignator>
307 element for this **named resource attribute** SHALL return a bag consisting of at least one **attribute**
308 **value**. The `MustBePresent` attribute governs whether this element returns **false** or **indeterminate**
309 in the case of the absence of the **named resource attribute**. If the **named resource attribute** is
310 not present and the `MustBePresent` attribute is set to `false` (its default value) this element
311 SHALL result in **false**. If the **named resource attribute** is not present and the `MustBePresent`
312 attribute is set to `true`, this element SHALL result in **indeterminate**. Regardless of the
313 `MustBePresent` attribute, if it cannot be determined whether the **named resource attribute** is
314 present or not present in the request context, or the value of the **named resource attribute** is
315 unavailable, then the expression SHALL result in **indeterminate**.

316 A **named resource attribute** SHALL be considered present if at least one **resource attribute**
317 exists that matches the values of its corresponding `AttributeId`, `DataType`, and `Issuer`
318 attributes as per the match semantics of the **AttributeDesignatorType** specification [Section 5.1].

319 The <ResourceAttributeIsPresent> MAY be passed to the <Apply> element as an
320 argument.

321

```
322 <xs:element name="ResourceAttributeIsPresent"  
323           type="xacml:AttributeDesignatorType"/>
```

324

325 The <ResourceAttributeIsPresent> element is of the **AttributeDesignatorType** complex
326 type.

327 The <ResourceAttributeIsPresent> element has the following attributes:

328 `AttributeId` [Required]

329 This attribute SHALL specify the `AttributeId` with which to match the **resource**
330 **attribute**.

331 `DataType` [Required]

332 This attribute SHALL specify the `DataType` with which to match the **resource attribute**.

333 `Issuer` [Optional]

334 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **resource**
335 **attribute**.

336 `MustBePresent` [Optional]

337 This attribute governs the whether the <ResourceAttributeIsPresent> element returns **false**
338 or **indeterminate** in the case of the absence of the **named resource attribute**. If the
339 **named resource attribute** is absent and `MustBePresent` is set to `false`, this element
340 SHALL result in **false**. If the **named resource attribute** is absent and `MustBePresent` is
341 set to `true`, this element SHALL result in **indeterminate**. If `MustBePresent` is not
342 supplied, its default value SHALL be `false`.

5.11. Element <ResourceAttributeDesignator>

343

344

345 The <ResourceAttributeDesignator> element retrieves a bag of values for a **named**
346 **resource attribute**. A **resource attribute** is an **attribute** that SHALL only be located within the
347 <Resource> element of the <xacml-context:Request> element. A **named resource**
348 **attribute** has specific criteria (described below) with which to match a **resource attribute**. A
349 **named resource attribute** SHALL be considered *present*, i.e. *not absent*, if there is at least one
350 **resource attribute** that matches the criteria. A **resource attribute value** is an **attribute value** that
351 is contained within a **resource attribute**.

352 The <ResourceAttributeDesignator> element SHALL return a bag of all the **resource**
353 **attribute values** that are matched by the **named resource attribute**. The `MustBePresent`
354 attribute governs whether this element returns an empty bag or **indeterminate** in the case of the
355 absence of the **named resource attribute**. If the **named resource attribute** is not present and the
356 `MustBePresent` attribute is set to `false` (its default value) this element SHALL result in an empty
357 bag. If the **named resource attribute** is not present and the `MustBePresent` attribute is set to
358 `true`, this element SHALL result in **indeterminate**. Regardless of the `MustBePresent` attribute, if
359 it cannot be determined whether the **named resource attribute** is present or not present in the
360 request context, or the value of the **named resource attribute** is unavailable, then the expression
361 SHALL result in **indeterminate**.

362 A **named resource attribute** SHALL match a **resource attribute** as per the match semantics
363 specified in the `AttributeDesignatorType` complex type [Section 5.1].

364 The <ResourceAttributeDesignator> MAY appear in the <ResourceMatch> element and
365 MAY be passed to the <Apply> element as an argument.

366

```
367 <xs:element name="ResourceAttributeDesignator"  
368           type="xacml:AttributeDesignatorType"/>
```

369

370 The <ResourceAttributeDesignator> element is of the `AttributeDesignatorType` complex
371 type.

372 The <ResourceAttributeDesignator> element has the following attributes:

373 `AttributeId` [Required]

374 This attribute SHALL specify the `AttributeId` with which to match the **resource**
375 **attribute**.

376 `DataType` [Required]

377 This attribute SHALL specify the `DataType` with which to match the **resource attribute**.

378 `Issuer` [Optional]

379 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **resource**
380 **attribute**.

381 `MustBePresent` [Optional]

382 This attribute governs the whether the <ResourceAttributeDesignator> element
383 returns an empty bag or **indeterminate** in the case of the absence of the **named resource**
384 **attribute**. If the **named resource attribute** is absent and `MustBePresent` is set to

385 false, this element SHALL result in an empty bag. If the **named resource attribute** is
386 absent and `MustBePresent` is set to `true`, this element SHALL result in **indeterminate**.
387 If `MustBePresent` is not supplied, its default value SHALL be `false`.

388

389 5.12. Element <ActionAttributeIsPresent>

390

391 The <ActionAttributeIsPresent> element determines whether a **named action attribute** is
392 present. An **action attribute** is an **attribute** that SHALL only be located within the <Action>
393 element of the <xacml-context:Request> element. A **named action attribute** has specific
394 criteria (described below) with which to match an **action attribute**. A **named action attribute**
395 SHALL be considered *present*, i.e. *not absent*, if there is at least one **action attribute** that matches
396 the criteria.

397 The <ActionAttributeIsPresent> element SHALL result in **true** if its **named action attribute**
398 is present. A result of **true** SHALL mean that a <ActionAttributeDesignator> element for
399 this **named action attribute** SHALL return a bag consisting of at least one **attribute value**. The
400 `MustBePresent` attribute governs whether this element returns **false** or **indeterminate** in the case
401 of the absence of the **named action attribute**. If the **named action attribute** is not present and the
402 `MustBePresent` attribute is set to `false` (its default value) this element SHALL result in **false**. If
403 the **named action attribute** is not present and the `MustBePresent` attribute is set to `true`, this
404 element SHALL result in **indeterminate**. Regardless of the `MustBePresent` attribute, if it cannot
405 be determined whether the **named action attribute** is present or not present in the request context,
406 or the value of the **named action attribute** is unavailable, then the expression SHALL result in
407 **indeterminate**.

408 A **named action attribute** SHALL be considered present if at least one **action attribute** exists that
409 matches the values of its corresponding `AttributeId`, `DataType`, and `Issuer` attributes as per
410 the match semantics of the **AttributeDesignatorType** specification [Section 5.1].

411 The <ActionAttributeIsPresent> MAY be passed to the <Apply> element as an argument.

412

```
413 <xs:element name="ActionAttributeIsPresent"  
414           type="xacml:AttributeDesignatorType" />
```

415

416 The <ActionAttributeIsPresent> element is of the **AttributeDesignatorType** complex type.

417 The <ActionAttributeIsPresent> element has the following attributes:

418 `AttributeId` [Required]

419 This attribute SHALL specify the `AttributeId` with which to match the **action attribute**.

420 `DataType` [Required]

421 This attribute SHALL specify the `DataType` with which to match the **action attribute**.

422 `Issuer` [Optional]

423 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **action**
424 **attribute**.

425 `MustBePresent` [Optional]

426 This attribute governs the whether the <ActionAttributesPresent> element returns *false* or
427 *indeterminate* in the case of the absence of the **named action attribute**. If the **named**
428 **action attribute** is absent and `MustBePresent` is set to `false`, this element SHALL
429 result in *false*. If the **named action attribute** is absent and `MustBePresent` is set to
430 `true`, this element SHALL result in *indeterminate*. If `MustBePresent` is not supplied, its
431 default value SHALL be `false`.

432 5.13. Element <ActionAttributeDesignator>

433

434 The <ActionAttributeDesignator> element retrieves a bag of values for a **named action**
435 **attribute**. An **action attribute** is an **attribute** that SHALL only be located within the <Action>
436 element of the <xacml-context:Request> element. A **named action attribute** has specific
437 criteria (described below) with which to match an **action attribute**. A **named action attribute**
438 SHALL be considered *present*, i.e. *not absent*, if there is at least one **action attribute** that matches
439 the criteria. A **action attribute value** is an **attribute value** that is contained within a **action**
440 **attribute**.

441 The <ActionAttributeDesignator> element SHALL return a bag of all the **action attribute**
442 **values** that are matched by the **named action attribute**. The `MustBePresent` attribute governs
443 whether this element returns an empty bag or *indeterminate* in the case of the absence of the
444 **named action attribute**. If the **named action attribute** is not present and the `MustBePresent`
445 attribute is set to `false` (its default value) this element SHALL result in an empty bag. If the **named**
446 **action attribute** is not present and the `MustBePresent` attribute is set to `true`, this element
447 SHALL result in *indeterminate*. Regardless of the `MustBePresent` attribute, if it cannot be
448 determined whether the **named action attribute** is present or not present in the request context, or
449 the value of the **named action attribute** is unavailable, then the expression SHALL result in
450 *indeterminate*.

451 A **named action attribute** SHALL match a **action attribute** as per the match semantics specified
452 in the **AttributeDesignatorType** complex type [Section 5.1].

453 The <ActionAttributeDesignator> MAY appear in the <ActionMatch> element and MAY
454 be passed to the <Apply> element as an argument.

455

```
456 <xs:element name="ActionAttributeDesignator"  
457           type="xacml:AttributeDesignatorType"/>
```

458

459 The <ActionAttributeDesignator> element is of the **AttributeDesignatorType** complex
460 type.

461 The <ActionAttributeDesignator> element has the following attributes:

462 `AttributeId` [Required]

463 This attribute SHALL specify the `AttributeId` with which to match the **action attribute**.

464 `DataType` [Required]

465 This attribute SHALL specify the `DataType` with which to match the **action attribute**.

466 `Issuer` [Optional]

467 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **action**
468 **attribute**.

469 `MustBePresent` [Optional]

470 This attribute governs the whether the `<ActionAttributeDesignator>` element returns
471 an empty bag or **indeterminate** in the case of the absence of the **named action attribute**.
472 If the **named action attribute** is absent and `MustBePresent` is set to `false`, this element
473 SHALL result in an empty bag. If the **named action attribute** is absent and
474 `MustBePresent` is set to `true`, this element SHALL result in **indeterminate**. If
475 `MustBePresent` is not supplied, its default value SHALL be `false`.

476 5.14. Element `<EnvironmentAttributeIsPresent>`

477

478 The `<EnvironmentAttributeIsPresent>` element determines whether a **named environment**
479 **attribute** is present. An **environment attribute** is an **attribute** that SHALL only be located within
480 the `<Environment>` element of the `<xacml-context:Request>` element. A **named**
481 **environment attribute** has specific criteria (described below) with which to match an **environment**
482 **attribute**. A **named environment attribute** SHALL be considered *present*, i.e. *not absent*, if there
483 is at least one **environment attribute** that matches the criteria.

484 The `<EnvironmentAttributeIsPresent>` element SHALL result in **true** if its **named**
485 **environment attribute** is present. A result of **true** SHALL mean that a
486 `<EnvironmentAttributeDesignator>` element for this **named environment attribute** SHALL
487 return a bag consisting of at least one **attribute value**. The `MustBePresent` attribute governs
488 whether this element returns **false** or **indeterminate** in the case of the absence of the **named**
489 **environment attribute**. If the **named environment attribute** is not present and the
490 `MustBePresent` attribute is set to `false` (its default value) this element SHALL result in **false**. If
491 the **named environment attribute** is not present and the `MustBePresent` attribute is set to `true`,
492 this element SHALL result in **indeterminate**. Regardless of the `MustBePresent` attribute, if it
493 cannot be determined whether the **named environment attribute** is present or not present in the
494 request context, or the value of the **named environment attribute** is unavailable, then the
495 expression SHALL result in **indeterminate**.

496 A **named environment attribute** SHALL be considered present if at least one **environment**
497 **attribute** exists that matches the values of its corresponding `AttributeId`, `DataType`, and
498 `Issuer` attributes as per the match semantics of the **AttributeDesignatorType** specification
499 [Section 5.1].

500 The `<EnvironmentAttributeIsPresent>` MAY be passed to the `<Apply>` element as an
501 argument.

502

```
503 <xs:element name="EnvironmentAttributeIsPresent "  
504           type="xacml:AttributeDesignatorType" />
```

505

506 The `<EnvironmentAttributeIsPresent>` element is of the **AttributeDesignatorType**
507 complex type.

508 The `<EnvironmentAttributeIsPresent>` element has the following attributes:

509 `AttributeId` [Required]

510 This attribute SHALL specify the `AttributeId` with which to match the **environment**
511 **attribute**.

512 `DataType` [Required]

513 This attribute SHALL specify the `DataType` with which to match the **environment attribute**.

514 `Issuer` [Optional]

515 This attribute, if supplied, SHALL specify the `Issuer` with which to match the **environment**
516 **attribute**.

517 `MustBePresent` [Optional]

518 This attribute governs the whether the `<EnvironmentAttributesPresent>` element returns
519 **false** or **indeterminate** in the case of the absence of the **named environment attribute**. If
520 the **named environment attribute** is absent and `MustBePresent` is set to `false`, this
521 element SHALL result in **false**. If the **named environment attribute** is absent and
522 `MustBePresent` is set to `true`, this element SHALL result in **indeterminate**. If
523 `MustBePresent` is not supplied, its default value SHALL be `false`.

524 5.15. Element `<EnvironmentAttributeDesignator>`

525

526 The `<EnvironmentAttributeDesignator>` element retrieves a bag of values for a **named**
527 **environment attribute**. A **environment attribute** is an **attribute** that SHALL only be located within
528 the `<Environment>` element of the `<xacml-context:Request>` element. A **named**
529 **environment attribute** has specific criteria (described below) with which to match a **environment**
530 **attribute**. A **named environment attribute** SHALL be considered *present*, i.e. *not absent*, if there
531 is at least one **environment attribute** that matches the criteria. A **environment attribute value** is
532 an **attribute value** that is contained within a **environment attribute**.

533 The `<EnvironmentAttributeDesignator>` element SHALL return a bag of all the
534 **environment attribute values** that are matched by the **named environment attribute**. The
535 `MustBePresent` attribute governs whether this element returns an empty bag or **indeterminate** in
536 the case of the absence of the **named environment attribute**. If the **named environment**
537 **attribute** is not present and the `MustBePresent` attribute is set to `false` (its default value) this
538 element SHALL result in an empty bag. If the **named environment attribute** is not present and the
539 `MustBePresent` attribute is set to `true`, this element SHALL result in **indeterminate**. Regardless
540 of the `MustBePresent` attribute, if it cannot be determined whether the **named environment**
541 **attribute** is present or not present in the request context, or the value of the **named environment**
542 **attribute** is unavailable, then the expression SHALL result in **indeterminate**.

543 A **named environment attribute** SHALL match a **environment attribute** as per the match
544 semantics specified in the **AttributeDesignatorType** complex type [Section 5.1].

545 The `<EnvironmentAttributeDesignator>` MAY be passed to the `<Apply>` element as an
546 argument.

547

```
548 <xs:element name="EnvironmentAttributeDesignator"  
549           type="xacml:AttributeDesignatorType" />
```

550

551 The `<EnvironmentAttributeDesignator>` element is of the **AttributeDesignatorType**
552 complex type.

553 The <EnvironmentAttributeDesignator> element has the following attributes:

554 AttributeId [Required]

555 This attribute SHALL specify the AttributeId with which to match the **environment**
556 **attribute**.

557 DataType [Required]

558 This attribute SHALL specify the DataType with which to match the **environment attribute**.

559 Issuer [Optional]

560 This attribute, if supplied, SHALL specify the Issuer with which to match the **environment**
561 **attribute**.

562 MustBePresent [Optional]

563 This attribute governs the whether the <EnvironmentAttributeDesignator> element
564 returns an empty bag or **indeterminate** in the case of the absence of the **named**
565 **environment attribute**. If the **named environment attribute** is absent and
566 MustBePresent is set to *false*, this element SHALL result in an empty bag. If the
567 **named environment attribute** is absent and MustBePresent is set to *true*, this element
568 SHALL result in **indeterminate**. If MustBePresent is not supplied, its default value
569 SHALL be *false*.

570

571

572

573

574