

```

1 <?xml version="1.0" encoding="iso-8859-1"?>
2 <xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/2011/06/power" xmlns:emix="
http://docs.oasis-open.org/ns/emix/2011/06" xmlns:resource="http://docs.oasis-open.org/ns/emix/2011/06/power/resource"
xmlns:ei="http://docs.oasis-open.org/ns/energyinterop/201110" xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0" xmlns:strm="
urn:ietf:params:xml:ns:icalendar-2.0:stream" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:scale="
http://docs.oasis-open.org/ns/emix/2011/06/siscale" targetNamespace="http://docs.oasis-open.org/ns/energyinterop/201110"
elementFormDefault="qualified">
3   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0" schemaLocation="
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/cs01/xsd/iCalendar.xsd"/>
4   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0" schemaLocation="
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/cs01/xsd/iCalendar-wscal-extensions.xsd"/>
5   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0" schemaLocation="
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/cs01/xsd/iCalendar-availability-extension.xsd"/>
6   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0" schemaLocation="
http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/cs01/xsd/iCalendar-valtypes.xsd"/>
7   <xs:import namespace="http://docs.oasis-open.org/ns/emix/2011/06/power/resource" schemaLocation="
http://docs.oasis-open.org/emix/emix/v1.0/csprd04/xsd/resource.xsd"/>
8   <xs:import namespace="http://docs.oasis-open.org/ns/emix/2011/06/power" schemaLocation="
http://docs.oasis-open.org/emix/emix/v1.0/csprd04/xsd/power.xsd"/>
9   <xs:import namespace="http://docs.oasis-open.org/ns/emix/2011/06" schemaLocation="
http://docs.oasis-open.org/emix/emix/v1.0/csprd04/xsd/emix.xsd"/>
10  <xs:import namespace="http://docs.oasis-open.org/ns/emix/2011/06/siscale" schemaLocation="
http://docs.oasis-open.org/emix/emix/v1.0/csprd04/xsd/siscale.xsd"/>
11  <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0:stream" schemaLocation="iCalendar-streams-extensions.xsd"/>
12  <!-- 1.0 EI Market Context -->
13  <!-- 1.1 Market Context Defintion -->
14  <xs:element name="eiMarketContext" type="ei:EiMarketContextType"/>
15  <xs:complexType name="EiMarketContextType">
16    <xs:sequence>
17      <xs:element ref="emix:marketContext" minOccurs="1" maxOccurs="1"/>
18      <xs:element ref="ei:marketName" minOccurs="0" maxOccurs="1">
19        <xs:annotation>
20          <xs:documentation>Name associated with the Market Context</xs:documentation>
21        </xs:annotation>
22      </xs:element>
23      <xs:element ref="emix:envelopeContents" minOccurs="0" maxOccurs="1"/>
24      <xs:element ref="emix:standardTerms" minOccurs="0" maxOccurs="1"/>
25      <xs:element ref="ei:simpleLevelContext" minOccurs="0" maxOccurs="1"/>
26      <xs:element ref="ei:applicationSpecificContextBase" minOccurs="0" maxOccurs="1"/>
27      <xs:element ref="ei:reportSpecifier" minOccurs="0" maxOccurs="unbounded"/>
28      <xs:element ref="ei:createdDateTime" minOccurs="0" maxOccurs="1"/>
29    </xs:sequence>
30    <xs:attribute ref="ei:schemaVersion" use="optional"/>
31  </xs:complexType>
32  <!-- 1.2 Market Expectations -->
33  <xs:element name="eiMarketExpectations" type="ei:EiMarketExpectationsType"/>
34  <xs:complexType name="EiMarketExpectationsType">
35    <xs:annotation>
36      <xs:documentation>Market Expectations inform the VEN what the expectations and rules are for a given Market Context
</xs:documentation>
37    </xs:annotation>
38    <xs:sequence>
39      <xs:element ref="ei:eiMarketRuleSet" minOccurs="0" maxOccurs="unbounded"/>
40      <xs:element ref="xcal:granularity" minOccurs="0" maxOccurs="1">
41        <xs:annotation>
42          <xs:documentation>Temporal granularity of market, i.e., a 5 minute market operates in 5 minute chunks, with a
fixed offset from the beginning of the business schedule</xs:documentation>
43        </xs:annotation>
44      </xs:element>
45      <xs:element ref="emix:nonStandardTermsHandling" maxOccurs="1">
46        <xs:annotation>
47          <xs:documentation>Non-Standard terms handling defines what Parties should do with any Term not listed in the

```

```

Market Rule Sets.</xs:documentation>
48     </xs:annotation>
49     </xs:element>
50 </xs:sequence>
51 </xs:complexType>
52 <!-- 1.3 Market Rule Set -->
53 <xs:element name="eiMarketRuleSet" type="ei:EiMarketRuleSetType"/>
54 <xs:complexType name="EiMarketRuleSetType">
55     <xs:annotation>
56         <xs:documentation>A collection of Requirements (Constraints) and how they are processed within this market</
xs:documentation>
57     </xs:annotation>
58     <xs:sequence>
59         <xs:element ref="emix:standardTermsSet" minOccurs="1" maxOccurs="1"/>
60         <xs:element ref="ei:ruleSetPurpose" minOccurs="1" maxOccurs="1"/>
61         <xs:element ref="ei:eiAvailBehavior" minOccurs="0" maxOccurs="1"/>
62     </xs:sequence>
63 </xs:complexType>
64 <!-- 1.4 Rule Set Purpose -->
65 <xs:element name="ruleSetPurpose" type="ei:RuleSetPurposeType"/>
66 <xs:simpleType name="RuleSetPurposeType">
67     <xs:annotation>
68         <xs:documentation>Defines the purpose of a market rule set.</xs:documentation>
69     </xs:annotation>
70     <xs:restriction base="xs:string">
71         <xs:enumeration value="minimum">
72             <xs:annotation>
73                 <xs:documentation>The Market does not accept this Term with a parameter set to a lesser value than this</
xs:documentation>
74             </xs:annotation>
75         </xs:enumeration>
76         <xs:enumeration value="maximum">
77             <xs:annotation>
78                 <xs:documentation>The Market does not accept this Term with a parameter set to a greater value than this</
xs:documentation>
79             </xs:annotation>
80         </xs:enumeration>
81         <xs:enumeration value="force">
82             <xs:annotation>
83                 <xs:documentation>Regardless of what the market participant requests, force the Term to the value here.</
xs:documentation>
84             </xs:annotation>
85         </xs:enumeration>
86         <xs:enumeration value="mustUnderstand">
87             <xs:annotation>
88                 <xs:documentation>Participants in this market must understand this constraint.</xs:documentation>
89             </xs:annotation>
90         </xs:enumeration>
91         <xs:enumeration value="ignore">
92             <xs:annotation>
93                 <xs:documentation>This Constraint will be ignored in all its forms.</xs:documentation>
94             </xs:annotation>
95         </xs:enumeration>
96     </xs:restriction>
97 </xs:simpleType>
98 <!-- 1.5 Simple Levels -->
99 <!-- 1.5.1 Level Context -->
100 <xs:element name="simpleLevelContext" type="ei:SimpleLevelContextType"/>
101 <xs:complexType name="SimpleLevelContextType">
102     <xs:annotation>
103         <xs:documentation>Simple Levels are a set of simple indicators about scarcity and value, in which an ordered set of
values indicate energy scarcity is above normal, normal, or below normal. Presumably, at higher levels, the VEN will use less.</

```

```

xs:documentation>
104     </xs:annotation>
105     <xs:sequence>
106         <xs:element name="levelNormalValue" type="xs:unsignedInt" minOccurs="0" maxOccurs="1">
107             <xs:annotation>
108                 <xs:documentation>The "normal" level indicating normal energy availability. Levels below normal indicate surplus,
levels above normal indicate increasing scarcity. If levelUpperLimit is 7, the levels are 1-7, and the levelNormalValue might be 3.
</xs:documentation>
109             </xs:annotation>
110         </xs:element>
111         <xs:element name="levelUpperLimit" type="xs:unsignedInt" minOccurs="0" maxOccurs="1">
112             <xs:annotation>
113                 <xs:documentation>The upper level for this context. If levelUpperLimit is 5, the levels are 1-5, where 5 indicates the
greatest scarcity.</xs:documentation>
114             </xs:annotation>
115         </xs:element>
116     </xs:sequence>
117 </xs:complexType>
118 <!-- 1.5.2 Program Context -->
119 <xs:element name="level" type="xs:unsignedInt">
120     <xs:annotation>
121         <xs:documentation>Indicates a specific level in within a simple level context. For example, a context may specify a level
between 1-7, and the current level is 4.</xs:documentation>
122     </xs:annotation>
123 </xs:element>
124 <!-- 2. Availability and Opt -->
125 <!-- 2.1 EI Availability -->
126 <xs:element name="eiAvail" type="ei:EiAvailType"/>
127 <xs:complexType name="EiAvailType">
128     <xs:sequence>
129         <xs:element ref="ei:availID" minOccurs="1" maxOccurs="1">
130             <xs:annotation>
131                 <xs:documentation>ID for this artifact</xs:documentation>
132             </xs:annotation>
133         </xs:element>
134         <xs:element ref="ei:venID" minOccurs="1" maxOccurs="unbounded">
135             <xs:annotation>
136                 <xs:documentation>VEN that offers this availability</xs:documentation>
137             </xs:annotation>
138         </xs:element>
139         <xs:element ref="ei:resourceID" minOccurs="0" maxOccurs="unbounded">
140             <xs:annotation>
141                 <xs:documentation>Resources to which the Availability applies.</xs:documentation>
142             </xs:annotation>
143         </xs:element>
144         <xs:element name="available" type="xcal:VavailabilityType" minOccurs="1" maxOccurs="1">
145             <xs:annotation>
146                 <xs:documentation>Resource ID for the for what is offered by the VEN</xs:documentation>
147             </xs:annotation>
148         </xs:element>
149         <xs:element ref="ei:createdDateTime" minOccurs="1" maxOccurs="1"/>
150         <xs:element ref="ei:eiAvailBehavior" minOccurs="1" maxOccurs="1"/>
151         <xs:element ref="emix:marketContext" minOccurs="1" maxOccurs="1"/>
152     </xs:sequence>
153     <xs:attribute ref="ei:schemaVersion" use="optional"/>
154 </xs:complexType>
155 <!-- 2.2 Availability Behavior -->
156 <xs:element name="eiAvailBehavior" type="ei:EiAvailBehaviorType"/>
157 <xs:simpleType name="EiAvailBehaviorType">
158     <xs:annotation>
159         <xs:documentation>When an Event is issued by the VTN, it is validated against the parameters and constraints that
were established when the program [market context] was set up, i.e., the program was configured to support Events between

```

12:00 and 16:00. If the Event is not within 12:00 and 16:00 then VEN must take some action to resolve the conflict. </

xs:documentation>

```
160     </xs:annotation>
161     <xs:restriction base="xs:string">
162       <xs:enumeration value="accept">
163         <xs:annotation>
164           <xs:documentation>Simply accept the issued DR event regardless of any conflicts</xs:documentation>
165         </xs:annotation>
166       </xs:enumeration>
167       <xs:enumeration value="reject">
168         <xs:annotation>
169           <xs:documentation>Reject any DR events that conflict with configured constraints</xs:documentation>
170         </xs:annotation>
171       </xs:enumeration>
172       <xs:enumeration value="restrict">
173         <xs:annotation>
174           <xs:documentation>Modify the DR event parameters so that they legally fall within the bounds of the configured
parameters.</xs:documentation>
175         </xs:annotation>
176       </xs:enumeration>
177     </xs:restriction>
178   </xs:simpleType>
179   <!-- 2.5 Opt -->
180   <xs:element name="eiOpt" type="ei:EiOptType"/>
181   <xs:complexType name="EiOptType">
182     <xs:annotation>
183       <xs:documentation>Opts are used tby the VEN to make temporary over-rides to the pre-existing agreement. For
example, a VEN may Opt In to events during the evening, or Opt Out from events during the World Series.</xs:documentation>
184     </xs:annotation>
185     <xs:sequence>
186       <xs:element ref="ei:optID" maxOccurs="1">
187         <xs:annotation>
188           <xs:documentation>This is the identifier that may used by other entities to refer to this instance of an OptOut.
189         </xs:documentation>
190       </xs:annotation>
191     </xs:element>
192     <xs:element ref="ei:optType" minOccurs="1" maxOccurs="1"/>
193     <xs:element ref="ei:optReason" minOccurs="1" maxOccurs="1"/>
194     <xs:element ref="emix:marketContext" minOccurs="0" maxOccurs="1"/>
195     <xs:element ref="ei:venID" minOccurs="1" maxOccurs="unbounded"/>
196     <xs:element ref="xcal:vavailability" minOccurs="0"/>
197     <xs:element ref="ei:eiMarketRuleSet" minOccurs="0" maxOccurs="unbounded"/>
198     <xs:element ref="ei:createdDateTime" minOccurs="1" maxOccurs="1"/>
199   </xs:sequence>
200   <xs:attribute ref="ei:schemaVersion" use="optional"/>
201 </xs:complexType>
202 <!-- 4.0 Reports -->
203 <!-- 4.1 Report -->
204 <xs:element name="eiReport" type="ei:EiReportType"/>
205 <xs:complexType name="EiReportType">
206   <xs:annotation>
207     <xs:documentation>eiReport is a Stream of [measurements] recorded over time and delivered to the requestor
periodically. The readings may be actual, computed, summed ir derived in some other manner.</xs:documentation>
208   </xs:annotation>
209   <xs:complexContent>
210     <xs:extension base="strm:StreamBaseType">
211       <xs:sequence>
212         <xs:element ref="ei:eiReportID" minOccurs="1" maxOccurs="1">
213           <xs:annotation>
214             <xs:documentation>reference ID to this Report.</xs:documentation>
215           </xs:annotation>
216         </xs:element>
```

```

217 <xs:element ref="ei:reportDescription" minOccurs="1" maxOccurs="unbounded"/>
218 <xs:element ref="ei:snap" minOccurs="0" maxOccurs="1"/>
219 <xs:element ref="ei:reportRequestID" minOccurs="1" maxOccurs="1">
220 <xs:annotation>
221 <xs:documentation>reference to Request that created this Report.</xs:documentation>
222 </xs:annotation>
223 </xs:element>
224 <xs:element ref="ei:reportSpecifierID" minOccurs="1" maxOccurs="1">
225 <xs:annotation>
226 <xs:documentation>reference to Specifier that defined this Report.</xs:documentation>
227 </xs:annotation>
228 </xs:element>
229 <xs:element ref="ei:reportName" minOccurs="0" maxOccurs="1">
230 <xs:annotation>
231 <xs:documentation>Name possibly for use in a user interface.</xs:documentation>
232 </xs:annotation>
233 </xs:element>
234 <xs:element ref="ei:createdDateTime" minOccurs="1" maxOccurs="1"/>
235 </xs:sequence>
236 <xs:attribute ref="ei:schemaVersion" use="optional"/>
237 </xs:extension>
238 </xs:complexContent>
239 </xs:complexType>
240 <!-- 4.2 Snap -->
241 <xs:element name="snap" type="ei:SnapType"/>
242 <xs:complexType name="SnapType">
243 <xs:annotation>
244 <xs:documentation>Snap represents a single moment with the information in the Payload is Measured / Generated</
xs:documentation>
245 </xs:annotation>
246 <xs:sequence>
247 <xs:element ref="ei:reportPayload" minOccurs="1" maxOccurs="1"/>
248 <xs:element ref="ei:statusDateTime" minOccurs="1" maxOccurs="1"/>
249 </xs:sequence>
250 </xs:complexType>
251 <!-- 4.3 Report Specification -->
252 <xs:element name="reportSpecifier" type="ei:ReportSpecifierType"/>
253 <xs:complexType name="ReportSpecifierType">
254 <xs:annotation>
255 <xs:documentation>Parameters that define the content of a Report Stream</xs:documentation>
256 </xs:annotation>
257 <xs:sequence>
258 <xs:element ref="ei:reportSpecifierID"/>
259 <xs:element ref="emix:marketContext" minOccurs="0" maxOccurs="1"/>
260 <xs:element ref="xcal:granularity">
261 <xs:annotation>
262 <xs:documentation>How frequently the [measurement] is to be recorded.</xs:documentation>
263 </xs:annotation>
264 </xs:element>
265 <xs:element name="reportBackDuration" type="xcal:DurationValueType" minOccurs="0" maxOccurs="1">
266 <xs:annotation>
267 <xs:documentation>Report back with the Report-To-Date each the passing of this Duration during the Report
Interval.</xs:documentation>
268 </xs:annotation>
269 </xs:element>
270 <xs:element name="reportInterval" type="xcal:WsCalendarIntervalType" minOccurs="0" maxOccurs="1">
271 <xs:annotation>
272 <xs:documentation>This is the overall period of reporting.</xs:documentation>
273 </xs:annotation>
274 </xs:element>
275 <xs:element ref="ei:specifierPayload" minOccurs="1" maxOccurs="unbounded"/>
276 </xs:sequence>

```



```

277 </xs:complexType>
278 <!-- 4.3.1 Specifier Payload -->
279 <xs:element name="specifierPayload" type="ei:SpecifierPayloadType"/>
280 <xs:complexType name="SpecifierPayloadType">
281   <xs:annotation>
282     <xs:documentation>Payload for use in Report Specifiers.</xs:documentation>
283   </xs:annotation>
284   <xs:sequence>
285     <xs:element ref="ei:rID" minOccurs="0" maxOccurs="1"/>
286     <xs:element ref="emix:itemBase" minOccurs="0" maxOccurs="1"/>
287     <xs:element ref="ei:readingType" minOccurs="1" maxOccurs="1"/>
288   </xs:sequence>
289 </xs:complexType>
290 <!-- 4.5 Report Requests -->
291 <!-- 4.5.1 Report Requests -->
292 <xs:element name="eiReportRequests" type="ei:ArrayOfReportRequests">
293   <xs:annotation>
294     <xs:documentation>Multiple Report and Snap Requests. </xs:documentation>
295   </xs:annotation>
296 </xs:element>
297 <xs:complexType name="ArrayOfReportRequests">
298   <xs:annotation>
299     <xs:documentation>Collection of Report and Snap Requests</xs:documentation>
300   </xs:annotation>
301   <xs:sequence>
302     <xs:element ref="ei:eiReportRequest" minOccurs="0" maxOccurs="unbounded"/>
303   </xs:sequence>
304 </xs:complexType>
305 <!-- 4.5.2 Report Request -->
306 <xs:element name="eiReportRequest" type="ei:EiReportRequestType"/>
307 <xs:complexType name="EiReportRequestType">
308   <xs:annotation>
309     <xs:documentation>This type is used to request an EiReport</xs:documentation>
310   </xs:annotation>
311   <xs:sequence>
312     <xs:element ref="ei:reportRequestID" minOccurs="1" maxOccurs="1"/>
313     <xs:element ref="ei:reportSpecifierID" minOccurs="1" maxOccurs="1"/>
314     <xs:element ref="ei:reportSpecifier" minOccurs="0" maxOccurs="1"/>
315     <xs:element ref="ei:eiTarget" minOccurs="0" maxOccurs="1"/>
316     <xs:element ref="ei:reportScheduler" minOccurs="1" maxOccurs="1"/>
317     <xs:element ref="ei:aggregateReport" minOccurs="1" maxOccurs="1"/>
318   <xs:annotation>
319     <xs:documentation>If true, aggregate all matching targets, if false, report each matching target individually.</
xs:documentation>
320   </xs:annotation>
321 </xs:element>
322 </xs:sequence>
323 </xs:complexType>
324 <!-- 4.6 Report Scheduling -->
325 <!-- 4.6.1 Event Gluon -->
326 <xs:element name="eventGluon" type="ei:EventGluonType" substitutionGroup="ei:reportScheduler"/>
327 <xs:complexType name="EventGluonType">
328   <xs:annotation>
329     <xs:documentation>Establishes relation between Active Interval of an Event and the Report Interval.</xs:documentation>
330   </xs:annotation>
331   <xs:complexContent>
332     <xs:extension base="ei:ReportSchedulerBaseType">
333       <xs:sequence>
334         <xs:element ref="xcal:gluon" minOccurs="0" maxOccurs="1">
335           <xs:annotation>
336             <xs:documentation>Used to establish relationship between Active Interval and Report Schedule. For example,
SS -T30M and FF T1H would convey that the report runs from 30 minutes before the Active Period to one hour after. If absent,

```

the Report Interval is equivalent to the Active Interval, i.e., the Report runs during Active Interval.</xs:documentation>

```
337     </xs:annotation>
338   </xs:element>
339   <xs:element ref="ei:eventID" minOccurs="0" maxOccurs="1">
340     <xs:annotation>
341       <xs:documentation>Indicates the Event this report is related to. If absent, must be delivered as part of a an
EiEvent. </xs:documentation>
342     </xs:annotation>
343   </xs:element>
344 </xs:sequence>
345 </xs:extension>
346 </xs:complexContent>
347 </xs:complexType>
348 <!-- 4.6.2 Request Report Gluon -->
349 <xs:element name="requestReportGluon" type="ei:RequestReportGluonType" substitutionGroup="ei:reportScheduler"/>
350 <xs:complexType name="RequestReportGluonType">
351   <xs:annotation>
352     <xs:documentation>Used if the Report Specifier includes a Report Interval to influence the expression of that Interval.
Information in the Gluon is inherited by the Report Interval in conformance with WS-Calendar.</xs:documentation>
353   </xs:annotation>
354   <xs:complexContent>
355     <xs:extension base="ei:ReportSchedulerBaseType">
356       <xs:sequence>
357         <xs:element ref="xcal:gluon" minOccurs="1" maxOccurs="1"/>
358       </xs:sequence>
359     </xs:extension>
360   </xs:complexContent>
361 </xs:complexType>
362 <!-- 4.6.3 Request Report Interval -->
363 <xs:element name="requestReportInterval" type="ei:RequestReportIntervalType" substitutionGroup="ei:reportScheduler"/>
364 <xs:complexType name="RequestReportIntervalType">
365   <xs:annotation>
366     <xs:documentation>request replaces report Interval in the Report Specification.</xs:documentation>
367   </xs:annotation>
368   <xs:complexContent>
369     <xs:extension base="ei:ReportSchedulerBaseType">
370       <xs:sequence>
371         <xs:element ref="xcal:interval" minOccurs="1" maxOccurs="1"/>
372       </xs:sequence>
373     </xs:extension>
374   </xs:complexContent>
375 </xs:complexType>
376 <!-- 4.6.3 Request Report Snap -->
377 <xs:element name="requestReportSnap" type="ei:RequestReportSnapType" substitutionGroup="ei:reportScheduler"/>
378 <xs:complexType name="RequestReportSnapType">
379   <xs:annotation>
380     <xs:documentation>request is for a single moment status. If Empty, the request is for "when received"</
xs:documentation>
381   </xs:annotation>
382   <xs:complexContent>
383     <xs:extension base="ei:ReportSchedulerBaseType">
384       <xs:sequence>
385         <xs:element ref="ei:statusDateTime" minOccurs="0" maxOccurs="1"/>
386       </xs:sequence>
387     </xs:extension>
388   </xs:complexContent>
389 </xs:complexType>
390 <!-- 4.6.6 Report Scheduler -->
391 <xs:element name="reportScheduler" type="ei:ReportSchedulerBaseType" abstract="true"/>
392 <xs:complexType name="ReportSchedulerBaseType" abstract="true">
393   <xs:annotation>
394     <xs:documentation>Base Class for Schedulers in Report Requests.</xs:documentation>
```

```

395     </xs:annotation>
396 </xs:complexType>
397 <!-- 4.7 Report Payload -->
398 <xs:element name="reportPayload" type="ei:ReportPayloadType" substitutionGroup="strm:streamPayloadBase"/>
399 <xs:complexType name="ReportPayloadType">
400     <xs:annotation>
401         <xs:documentation>Report Payload for use in Reports, snaps, and projections.</xs:documentation>
402     </xs:annotation>
403     <xs:complexContent>
404         <xs:extension base="strm:StreamPayloadBaseType">
405             <xs:sequence>
406                 <xs:element ref="ei:rID" minOccurs="0" maxOccurs="1"/>
407                 <xs:element ref="ei:confidence" minOccurs="0" maxOccurs="1"/>
408                 <xs:element ref="ei:readingType" minOccurs="0" maxOccurs="1"/>
409                 <xs:element ref="ei:accuracy" minOccurs="0" maxOccurs="1"/>
410                 <xs:element ref="ei:payloadBase" minOccurs="1" maxOccurs="1"/>
411             </xs:sequence>
412         </xs:extension>
413     </xs:complexContent>
414 </xs:complexType>
415 <!-- 4.8 Delivery -->
416 <!-- Minimal Report model for use in transactive energy, etc -->
417 <!-- Also represents minimum model for EMIX Delivery -->
418 <!-- 4.8.1 Delivery -->
419 <xs:element name="eiDelivery" type="ei:EiDeliveryType"/>
420 <xs:complexType name="EiDeliveryType">
421     <xs:annotation>
422         <xs:documentation>eiDelivery is a minimal report that assume that all esential paramters (type, etc) come form an
existing market context or transaction.</xs:documentation>
423     </xs:annotation>
424     <xs:complexContent>
425         <xs:extension base="strm:StreamBaseType">
426             <xs:sequence>
427                 <xs:element ref="ei:eiDeliveryID" minOccurs="1" maxOccurs="1"/>
428                 <xs:element ref="ei:createdDateTime" minOccurs="1" maxOccurs="1"/>
429                 <xs:element ref="emix:delivery" minOccurs="1" maxOccurs="1"/>
430             </xs:sequence>
431             <xs:attribute ref="ei:schemaVersion" use="optional"/>
432         </xs:extension>
433     </xs:complexContent>
434 </xs:complexType>
435 <!-- 4.8.2 Delivery Payload -->
436 <xs:element name="deliveryPayload" type="ei:DeliveryPayloadType" substitutionGroup="strm:streamPayloadBase"/>
437 <xs:complexType name="DeliveryPayloadType">
438     <xs:annotation>
439         <xs:documentation>This is the Payload for EiDelivery.</xs:documentation>
440     </xs:annotation>
441     <xs:complexContent>
442         <xs:extension base="strm:StreamPayloadBaseType">
443             <xs:sequence>
444                 <xs:element ref="emix:quantity" minOccurs="1" maxOccurs="1"/>
445             </xs:sequence>
446         </xs:extension>
447     </xs:complexContent>
448 </xs:complexType>
449 <!-- 4.9 Report Description Type -->
450 <xs:element name="reportDescription" type="ei:ReportDescriptionType"/>
451 <xs:complexType name="ReportDescriptionType">
452     <xs:annotation>
453         <xs:documentation>Describes the subject and attributes of a report.</xs:documentation>
454     </xs:annotation>
455     <xs:sequence>

```



```

456     <xs:element ref="ei:rID" minOccurs="0" maxOccurs="1"/>
457     <xs:annotation>
458         <xs:documentation>Optional element for use only if more than one description.</xs:documentation>
459     </xs:annotation>
460 </xs:element>
461 <xs:element ref="ei:reportSubject" minOccurs="0" maxOccurs="1"/>
462 <xs:element ref="ei:reportDataSource" minOccurs="0" maxOccurs="1"/>
463 <xs:element ref="ei:reportType" minOccurs="1" maxOccurs="1"/>
464 <xs:element ref="emix:itemBase" minOccurs="0" maxOccurs="1"/>
465     <xs:annotation>
466         <xs:documentation>What is measured or tracked in this report. Absent if report tracks Price or Level.</
xs:documentation>
467     </xs:annotation>
468 </xs:element>
469 <xs:element ref="ei:readingType" minOccurs="0" maxOccurs="1"/>
470 <xs:element ref="ei:aggregateReport" minOccurs="0" maxOccurs="1"/>
471 </xs:sequence>
472 </xs:complexType>
473 <!-- 5.0 Registration and Parties-->
474 <!-- 5.1 Registration-->
475 <xs:element name="eiRegistration" type="ei:EiRegistrationType"/>
476 <xs:complexType name="EiRegistrationType" abstract="true">
477     <xs:annotation>
478         <xs:documentation>Registration varies from Market to Market and VEN to VEN and so cannot be defined here. Parties
wishing to exchange Registration should extend this abstract type to meet their particular needs.</xs:documentation>
479     </xs:annotation>
480     <xs:sequence>
481         <xs:element ref="ei:registreePartyID" minOccurs="0" maxOccurs="unbounded"/>
482         <xs:element ref="ei:registrarPartyID" minOccurs="1" maxOccurs="1"/>
483         <xs:element ref="ei:eiRegistrationInfo" minOccurs="1" maxOccurs="1"/>
484     </xs:sequence>
485 </xs:complexType>
486 <!-- 5.2 Party-->
487 <!--
488 <xs:element name="registeredParty" type="ei:EiPartyType"/>
489
490 -->
491 <xs:element name="eiParty" type="ei:EiPartyType"/>
492 <xs:complexType name="EiPartyType">
493     <xs:sequence>
494         <xs:element ref="ei:partyID" minOccurs="0" maxOccurs="1"/>
495         <xs:element ref="ei:partyName" minOccurs="0" maxOccurs="1"/>
496         <xs:element ref="ei:partyRole" minOccurs="0" maxOccurs="1"/>
497     </xs:sequence>
498     <xs:attribute ref="ei:schemaVersion" use="optional"/>
499 </xs:complexType>
500 <!-- 5.3 Resource Target -->
501 <xs:element name="reportSubject" type="ei:EiTargetType">
502     <xs:annotation>
503         <xs:documentation>Subject[s] of this report. Examples might be meter1 or chiller2, etc. Subjects may be tangible
devices or subsystems that are metered or they may be logical entities.</xs:documentation>
504     </xs:annotation>
505 </xs:element>
506 <xs:element name="reportDataSource" type="ei:EiTargetType">
507     <xs:annotation>
508         <xs:documentation>Sources for data in this report. Examples include meters or submeters. For example, if a meter is
capable of providing two different types of measurements, then each measurement stream would be separately identified.</
xs:documentation>
509     </xs:annotation>
510 </xs:element>
511 <xs:element name="eiTarget" type="ei:EiTargetType"/>
512 <xs:complexType name="EiTargetType">

```

```

513     <xs:annotation>
514         <xs:documentation>A set of elements to that collectively name who is participating or should participate in an EI
interactions</xs:documentation>
515     </xs:annotation>
516     <xs:sequence>
517         <xs:element ref="emix:emixInterface" minOccurs="0" maxOccurs="unbounded"/>
518         <xs:element ref="ei:groupID" minOccurs="0" maxOccurs="unbounded"/>
519         <xs:element ref="ei:groupName" minOccurs="0" maxOccurs="unbounded"/>
520         <xs:element ref="ei:resourceID" minOccurs="0" maxOccurs="unbounded"/>
521         <xs:element ref="ei:venID" minOccurs="0" maxOccurs="unbounded"/>
522         <xs:element ref="ei:partyID" minOccurs="0" maxOccurs="unbounded"/>
523     </xs:sequence>
524 </xs:complexType>
525 <!-- 5.4 Registration Info Type -->
526 <xs:element name="eiRegistrationInfo" type="ei:EiRegistrationInfoType"/>
527 <xs:complexType name="EiRegistrationInfoType" abstract="true">
528     <xs:annotation>
529         <xs:documentation>RegistrationInfo is an an abstract class to which additional market-relevant registration information
might be added. This specification does not define registration. Markets that use registration should extend this abstract class
for use in Registration payloads.</xs:documentation>
530     </xs:annotation>
531     <xs:sequence/>
532 </xs:complexType>
533 <!-- 6.0 Events and Event Information -->
534 <!-- 6.1 Events -->
535 <xs:element name="eiEvent" type="ei:EiEventType"/>
536 <xs:complexType name="EiEventType">
537     <xs:annotation>
538         <xs:documentation>Core element of event-based demand response. An Event consists of the time periods, deadlines,
and transitions during which Demand Resources perform. The VTN specifies the duration and applicability of an Event. Some
deadlines, time periods, and transitions may not be not applicable to all products or services.</xs:documentation>
539     </xs:annotation>
540     <xs:sequence>
541         <xs:element ref="ei:eventMessageID" minOccurs="0" maxOccurs="1">
542             <xs:annotation>
543                 <xs:documentation>Reference ID for the EI Event instance</xs:documentation>
544             </xs:annotation>
545         </xs:element>
546         <xs:element ref="ei:eventDescriptor" minOccurs="1" maxOccurs="1">
547             <xs:annotation>
548                 <xs:documentation>These are the core attributes of the Event itself.</xs:documentation>
549             </xs:annotation>
550         </xs:element>
551         <xs:element ref="ei:eiActivePeriod" minOccurs="1" maxOccurs="1"/>
552         <xs:element ref="ei:eiEventSignals" minOccurs="1" maxOccurs="1"/>
553         <xs:element ref="ei:eiTarget" minOccurs="0" maxOccurs="1"/>
554         <xs:element ref="ei:eiReportRequests" minOccurs="0" maxOccurs="unbounded"/>
555     </xs:sequence>
556     <xs:attribute ref="ei:schemaVersion" use="optional"/>
557 </xs:complexType>
558 <!-- 6.2 Event Signals -->
559 <xs:element name="eiEventSignals" type="ei:ArrayOfSignals">
560     <xs:annotation>
561         <xs:documentation>Multiple signals conveyed with an event. </xs:documentation>
562     </xs:annotation>
563 </xs:element>
564 <xs:complexType name="ArrayOfSignals">
565     <xs:annotation>
566         <xs:documentation>Collection of Signal Base derived elements</xs:documentation>
567     </xs:annotation>
568     <xs:sequence>
569         <xs:element ref="ei:eiEventSignal" minOccurs="0" maxOccurs="unbounded"/>

```

```

570     <xs:element ref="ei:eiEventBaseline" minOccurs="0" maxOccurs="unbounded"/>
571   </xs:sequence>
572 </xs:complexType>
573 <!-- 6.3 Event Descriptor -->
574 <xs:element name="eventDescriptor" type="ei:EventDescriptorType"/>
575 <xs:complexType name="EventDescriptorType">
576   <xs:sequence>
577     <xs:element ref="ei:eventID" minOccurs="1" maxOccurs="1"/>
578     <xs:element ref="ei:modificationNumber" minOccurs="1" maxOccurs="1"/>
579     <xs:element ref="ei:modificationDateTime" minOccurs="0" maxOccurs="1"/>
580     <xs:element name="modificationReason" type="xs:string" minOccurs="0" maxOccurs="1">
581       <xs:annotation>
582         <xs:documentation>The reason the event is being cancelled or modified.</xs:documentation>
583       </xs:annotation>
584     </xs:element>
585     <xs:element ref="ei:priority" minOccurs="0" maxOccurs="1"/>
586     <xs:element ref="ei:eiMarketContext" minOccurs="1" maxOccurs="1"/>
587     <xs:element ref="ei:createdDateTime" minOccurs="0" maxOccurs="1"/>
588     <xs:element ref="ei:eventStatus" minOccurs="0" maxOccurs="1"/>
589     <xs:element name="operatingDay" type="xcal:DateTimeType" minOccurs="0" maxOccurs="1">
590       <xs:annotation>
591         <xs:documentation>Date of Start of Event </xs:documentation>
592       </xs:annotation>
593     </xs:element>
594     <xs:element ref="ei:testEvent" minOccurs="0" maxOccurs="1">
595       <xs:annotation>
596         <xs:documentation>testEvent can be treated as a boolean by either not including it (= false) or using the null
597 string. For new work, indicated the Operation Payload. Supports backward compatibility with OpenADR 1.0.</xs:documentation>
598       </xs:annotation>
599     </xs:element>
600     <xs:element name="vtnComment" type="xs:string" minOccurs="0">
601       <xs:annotation>
602         <xs:documentation>Additional Event information provided by the Operator [VTN].</xs:documentation>
603       </xs:annotation>
604     </xs:element>
605   </xs:sequence>
606 </xs:complexType>
607 <!-- 6.4 Event Signal -->
608 <xs:element name="eiEventSignal" type="ei:EiEventSignalType" substitutionGroup="strm:streamBase"/>
609 <xs:complexType name="EiEventSignalType">
610   <xs:annotation>
611     <xs:documentation>This type is used for describing the signal type communications.</xs:documentation>
612     <xs:appinfo>eiEventSignal should be paired with information in the Type acts AS IF a Gluon to the Intervals in the
613 Stream</xs:appinfo>
614   </xs:annotation>
615   <xs:complexContent>
616     <xs:extension base="strm:StreamBaseType">
617       <xs:sequence>
618         <xs:element ref="ei:eiTarget" minOccurs="0" maxOccurs="unbounded"/>
619         <xs:element ref="emix:marketContext" minOccurs="0" maxOccurs="1"/>
620         <xs:element ref="ei:signalName" minOccurs="1" maxOccurs="1"/>
621         <xs:element ref="ei:signalType" minOccurs="1" maxOccurs="1"/>
622         <xs:element ref="ei:signalID"/>
623         <xs:element ref="emix:itemBase" minOccurs="0" maxOccurs="1">
624           <xs:annotation>
625             <xs:documentation>This is the units of the signal. </xs:documentation>
626           </xs:annotation>
627         </xs:element>
628         <xs:element ref="ei:currentValue" minOccurs="0" maxOccurs="1"/>
629       </xs:sequence>
630     </xs:extension>
631   </xs:complexContent>
632   <xs:attribute ref="ei:schemaVersion"/>
633 </xs:complexType>

```

```

630     </xs:complexContent>
631 </xs:complexType>
632 <!-- 6.4.1 Current Signal Payload -->
633 <xs:element name="currentValue" type="ei:CurrentValueType"/>
634 <xs:complexType name="CurrentValueType">
635     <xs:annotation>
636         <xs:documentation>Current Value reprises what the Signal Payload is at the moment EventInfo is created.</
xs:documentation>
637     </xs:annotation>
638     <xs:sequence>
639         <xs:element ref="ei:payloadBase" minOccurs="1" maxOccurs="1"/>
640     </xs:sequence>
641 </xs:complexType>
642 <!-- 6.4.2 Signal Payload -->
643 <xs:element name="signalPayload" type="ei:SignalPayloadType" substitutionGroup="strm:streamPayloadBase"/>
644 <xs:complexType name="SignalPayloadType">
645     <xs:annotation>
646         <xs:documentation>The Signal Payload is a Stream Payload for conveyance within an EiEventSignal.</
xs:documentation>
647     </xs:annotation>
648     <xs:complexContent>
649         <xs:extension base="strm:StreamPayloadBaseType">
650             <xs:sequence>
651                 <xs:element ref="ei:payloadBase" minOccurs="1" maxOccurs="1"/>
652             </xs:sequence>
653         </xs:extension>
654     </xs:complexContent>
655 </xs:complexType>
656 <!-- 6.5 Event Baseline Elements -->
657 <!-- 6.5.1 Baseline Information -->
658 <xs:element name="eiEventBaseline" type="ei:EiEventBaselineType" substitutionGroup="strm:streamBase"/>
659 <xs:complexType name="EiEventBaselineType">
660     <xs:annotation>
661         <xs:documentation>This type is used for describing baseline for the event.</xs:documentation>
662         <xs:appinfo>eventBaseline should be paired with eventBaselinePayloads types in the gluons (Stream header and
market context) and the intervals in the stream (or sequence)</xs:appinfo>
663     </xs:annotation>
664     <xs:complexContent>
665         <xs:extension base="strm:StreamBaseType">
666             <xs:sequence>
667                 <xs:element ref="ei:baselineID" minOccurs="1" maxOccurs="1"/>
668                 <xs:element name="baselineInterval" type="xcal:WsCalendarIntervalType" minOccurs="0" maxOccurs="unbounded
">
669                     <xs:annotation>
670                         <xs:documentation>One or more Baseline Intervals indicate comprable times that MAY be used as a Baseline.
Information for the Baseline Intervals is not conveyed in this type, and the Interval may be past, present, or future.</
xs:documentation>
671                     </xs:annotation>
672                     </xs:element>
673                     <xs:element ref="ei:resourceID" minOccurs="0" maxOccurs="unbounded"/>
674                     <xs:element ref="ei:baselineName" minOccurs="0" maxOccurs="1"/>
675                     <xs:element ref="emix:itemBase" minOccurs="0" maxOccurs="1">
676                         <xs:annotation>
677                             <xs:documentation>This indicates the units of the signal. </xs:documentation>
678                         </xs:annotation>
679                     </xs:element>
680                     <xs:element ref="ei:currentValue" minOccurs="0" maxOccurs="1"/>
681                 </xs:sequence>
682             </xs:extension>
683         </xs:complexContent>
684     </xs:complexType>
685 <!-- 6.5.3 Baseline Description Information -->

```

```

686 <xs:element name="eventBaselinePayload" type="ei:EventBaselinePayloadType" substitutionGroup="
strm:streamPayloadBase"/>
687 <xs:complexType name="EventBaselinePayloadType">
688 <xs:annotation>
689 <xs:documentation>Calculated Energy Baseline: A Baseline is an estimate of the electricity that would have been
consumed by a Demand Resource in the absence of a Demand Response Event. The Baseline is compared to the actual
metered electricity consumption during the Demand Response Event to determine the Demand Modification Value. Depending
on the type of Demand Response product or service, Baseline calculations may be performed in real time or after the fact.
Baseline may not be known at time of event, in which case missing payload is used.</xs:documentation>
690 </xs:annotation>
691 <xs:complexContent>
692 <xs:extension base="strm:StreamPayloadBaseType">
693 <xs:sequence>
694 <xs:element ref="ei:payloadQuantity" minOccurs="0" maxOccurs="1"/>
695 </xs:sequence>
696 </xs:extension>
697 </xs:complexContent>
698 </xs:complexType>
699 <!-- 6.9 Payload Base -->
700 <!-- These types are used in all EI Streams (Signal, Payload, Baseline) to convey the information that changes as conformed
by the SignalType/ReportType/Baseline -->
701 <xs:element name="payloadApplicationSpecific" type="ei:PayloadApplicationSpecificType" abstract="true" substitutionGroup
="ei:payloadBase"/>
702 <xs:complexType name="PayloadApplicationSpecificType">
703 <xs:annotation>
704 <xs:documentation>This Payload contains the information that changes in each Stream payload.</xs:documentation>
705 <xs:appinfo> Applications that wish to use this should (1) define the appropriate extension to the abstract
applicationSpecificPayloadBase and (2) agree to indicates its use with a signalType conforming to the EI Extension pattern
(x-*). Different extensions should be made for Signals, Reports, and perhaps Baselines.</xs:appinfo>
706 </xs:annotation>
707 <xs:complexContent>
708 <xs:extension base="ei:PayloadBaseType">
709 <xs:sequence>
710 <xs:element ref="ei:applicationSpecificPayloadBase" minOccurs="1" maxOccurs="1"/>
711 </xs:sequence>
712 </xs:extension>
713 </xs:complexContent>
714 </xs:complexType>
715 <xs:element name="payloadProduct" type="ei:PayloadProductType" abstract="true" substitutionGroup="ei:payloadBase"/>
716 <xs:complexType name="PayloadProductType">
717 <xs:annotation>
718 <xs:documentation>This Payload contains the information that changes in each Stream payload.</xs:documentation>
719 </xs:annotation>
720 <xs:complexContent>
721 <xs:extension base="ei:PayloadBaseType">
722 <xs:sequence>
723 <xs:element ref="emix:productDescription" minOccurs="1" maxOccurs="1"/>
724 </xs:sequence>
725 </xs:extension>
726 </xs:complexContent>
727 </xs:complexType>
728 <xs:element name="payloadLevel" type="ei:PayloadLevelType" substitutionGroup="ei:payloadBase"/>
729 <xs:complexType name="PayloadLevelType">
730 <xs:annotation>
731 <xs:documentation>This is the Payload for Signals that convey Simple Levels.</xs:documentation>
732 </xs:annotation>
733 <xs:complexContent>
734 <xs:extension base="ei:PayloadBaseType">
735 <xs:sequence>
736 <xs:element ref="ei:level" minOccurs="1" maxOccurs="1"/>
737 </xs:sequence>
738 </xs:extension>

```



```
739     </xs:complexContent>
740 </xs:complexType>
741 <xs:element name="payloadFloat" type="ei:PayloadFloatType" substitutionGroup="ei:payloadBase"/>
742 <xs:complexType name="PayloadFloatType">
743     <xs:annotation>
744         <xs:documentation>This is the Payload for Signals that require a Quantity.</xs:documentation>
745     </xs:annotation>
746     <xs:complexContent>
747         <xs:extension base="ei:PayloadBaseType">
748             <xs:sequence>
749                 <xs:element name="value" type="xs:float" minOccurs="1" maxOccurs="1"/>
750             </xs:sequence>
751         </xs:extension>
752     </xs:complexContent>
753 </xs:complexType>
754 <xs:element name="payloadPrice" type="ei:PayloadPriceType" substitutionGroup="ei:payloadBase"/>
755 <xs:complexType name="PayloadPriceType">
756     <xs:annotation>
757         <xs:documentation>This is the Payload for Signals that require a Price.</xs:documentation>
758     </xs:annotation>
759     <xs:complexContent>
760         <xs:extension base="ei:PayloadBaseType">
761             <xs:sequence>
762                 <xs:element ref="emix:price" minOccurs="1" maxOccurs="1"/>
763             </xs:sequence>
764         </xs:extension>
765     </xs:complexContent>
766 </xs:complexType>
767 <xs:element name="payloadPriceMultiplier" type="ei:PayloadPriceMultiplierType" substitutionGroup="ei:payloadBase"/>
768 <xs:complexType name="PayloadPriceMultiplierType">
769     <xs:annotation>
770         <xs:documentation>This is the Payload for Signals that require a Quantity.</xs:documentation>
771     </xs:annotation>
772     <xs:complexContent>
773         <xs:extension base="ei:PayloadBaseType">
774             <xs:sequence>
775                 <xs:element ref="emix:priceMultiplier" minOccurs="1" maxOccurs="1"/>
776             </xs:sequence>
777         </xs:extension>
778     </xs:complexContent>
779 </xs:complexType>
780 <xs:element name="payloadPriceRelative" type="ei:PayloadPriceRelativeType" substitutionGroup="ei:payloadBase"/>
781 <xs:complexType name="PayloadPriceRelativeType">
782     <xs:annotation>
783         <xs:documentation>This is the Payload for Signals that require a Quantity.</xs:documentation>
784     </xs:annotation>
785     <xs:complexContent>
786         <xs:extension base="ei:PayloadBaseType">
787             <xs:sequence>
788                 <xs:element ref="emix:priceRelative" minOccurs="1" maxOccurs="1"/>
789             </xs:sequence>
790         </xs:extension>
791     </xs:complexContent>
792 </xs:complexType>
793 <xs:element name="payloadQuantity" type="ei:PayloadQuantityType" substitutionGroup="ei:payloadBase"/>
794 <xs:complexType name="PayloadQuantityType">
795     <xs:annotation>
796         <xs:documentation>This is the Payload for Signals that require a Quantity.</xs:documentation>
797     </xs:annotation>
798     <xs:complexContent>
799         <xs:extension base="ei:PayloadBaseType">
800             <xs:sequence>
```

```

801         <xs:element ref="emix:quantity" minOccurs="1" maxOccurs="1"/>
802     </xs:sequence>
803 </xs:extension>
804 </xs:complexContent>
805 </xs:complexType>
806 <!-- 6.9.9 Base Payload -->
807 <xs:element name="payloadBase" type="ei:PayloadBaseType" abstract="true"/>
808 <xs:complexType name="PayloadBaseType" abstract="true">
809     <xs:annotation>
810         <xs:documentation>base for information in Signal / Baseline / Report Payloads</xs:documentation>
811     </xs:annotation>
812 </xs:complexType>
813 <!-- 7.0 Tenders, Quotes, and Transactions -->
814 <!-- 7.1 Tenders -->
815 <xs:element name="eiTender" type="ei:EiTenderType"/>
816 <xs:complexType name="EiTenderType">
817     <xs:annotation>
818         <xs:documentation>Tender is an offer to buy or sell. A Tender can be for one EmixBase derived type.</
xs:documentation>
819     </xs:annotation>
820     <xs:sequence>
821         <xs:element ref="ei:tenderID" minOccurs="1" maxOccurs="1"/>
822         <xs:element ref="emix:emixBase" minOccurs="1" maxOccurs="1"/>
823     </xs:sequence>
824 </xs:complexType>
825 <!-- 7.2 Quotes -->
826 <xs:element name="eiQuote" type="ei:EiQuoteType"/>
827 <xs:complexType name="EiQuoteType">
828     <xs:annotation>
829         <xs:documentation>An Indication of the price of a possible Tender such as the transacted price of a previous Tender or
a published forecast of a price of a possible Tender. A Quote can be for one EmixBase derived type.</xs:documentation>
830     </xs:annotation>
831     <xs:sequence>
832         <xs:element ref="ei:quoteID" minOccurs="1" maxOccurs="1"/>
833         <xs:element ref="emix:emixBase" minOccurs="1" maxOccurs="unbounded"/>
834     </xs:sequence>
835 </xs:complexType>
836 <!-- 7.3 Transactions -->
837 <xs:element name="eiTransaction" type="ei:EiTransactionType"/>
838 <xs:complexType name="EiTransactionType">
839     <xs:annotation>
840         <xs:documentation>A Transaction is a specific agreement to accept a specific Tender.</xs:documentation>
841     </xs:annotation>
842     <xs:sequence>
843         <xs:element ref="ei:transactionID" minOccurs="1" maxOccurs="1"/>
844         <xs:element ref="ei:tenderID" minOccurs="0" maxOccurs="1"/>
845         <xs:element ref="emix:emixBase" minOccurs="1" maxOccurs="1"/>
846     </xs:sequence>
847 </xs:complexType>
848 <!-- 8. Responses -->
849 <!-- 8.1 EiResponse -->
850 <xs:element name="eiResponse" type="ei:EiResponseType"/>
851 <xs:complexType name="EiResponseType">
852     <xs:sequence>
853         <xs:element ref="ei:responseCode" minOccurs="1" maxOccurs="1"/>
854         <xs:element ref="ei:responseDescription" minOccurs="0" maxOccurs="1"/>
855         <xs:element ref="ei:refID" minOccurs="1" maxOccurs="1">
856             <xs:annotation>
857                 <xs:documentation>Identifier of the Message/Request that is a response to</xs:documentation>
858             </xs:annotation>
859         </xs:element>
860         <xs:element ref="ei:responseTermsViolated" minOccurs="0" maxOccurs="1"/>

```

```

861     <xs:element ref="ei:createdDateTime" minOccurs="0" maxOccurs="1"/>
862   </xs:sequence>
863 </xs:complexType>
864 <!-- 8.2 Responses -->
865 <xs:element name="responses" type="ei:ArrayofResponses"/>
866 <xs:complexType name="ArrayOfResponses">
867   <xs:annotation>
868     <xs:documentation>Collection of Responses. When a service operation regards multiple referenceable items, each
referenced item MAY have its own response. Always accompanied by an overall Response Type.</xs:documentation>
869   </xs:annotation>
870   <xs:sequence>
871     <xs:element name="response" type="ei:EiResponseType" minOccurs="0" maxOccurs="unbounded"/>
872   </xs:sequence>
873 </xs:complexType>
874 <!-- 8.3 EI Event Response -->
875 <xs:element name="eiEventResponse" type="ei:EiEventResponseType"/>
876 <xs:complexType name="EiEventResponseType">
877   <xs:annotation>
878     <xs:documentation>Response adding information to manage event-based messages</xs:documentation>
879     <xs:appinfo>EventResponses require Modification number as well as other Response information</xs:appinfo>
880   </xs:annotation>
881   <xs:complexContent>
882     <xs:extension base="ei:EiResponseType">
883       <xs:sequence>
884         <xs:element ref="ei:qualifiedEventID" minOccurs="1" maxOccurs="1"/>
885         <xs:element ref="ei:optType" minOccurs="1" maxOccurs="1"/>
886       </xs:sequence>
887     </xs:extension>
888   </xs:complexContent>
889 </xs:complexType>
890 <!-- 8.4 Event Responses -->
891 <xs:element name="eventResponses" type="ei:ArrayofEventResponses"/>
892 <xs:complexType name="ArrayOfEventResponses">
893   <xs:annotation>
894     <xs:documentation>Collection of Event Responses. When an regards multiple referenceable items, each referenced
item MAY have its own response. Always accompanied by an overall Response Type. </xs:documentation>
895   </xs:annotation>
896   <xs:sequence>
897     <xs:element name="eventResponse" type="ei:EiEventResponseType" minOccurs="0" maxOccurs="unbounded"/>
898   </xs:sequence>
899 </xs:complexType>
900 <!-- 8.7 Response Code -->
901 <xs:element name="responseCode" type="ei:ResponseCodeType"/>
902 <xs:simpleType name="ResponseCodeType">
903   <xs:annotation>
904     <xs:documentation>Similar to HTTP 1.1 Error Pattern, 1st digit sufficient for most error processing
905     - 1xx: Informational - Request received, continuing process
906     - 2xx: Success - The action was successfully received, understood, and accepted
907     - 3xx: Pending - Further action must be taken in order to complete the request
908     - 4xx: Requester Error - The request contains bad syntax or cannot be fulfilled
909     - 5xx: Responder Error - The responder failed to fulfill an apparently valid request
910     xx is used for defining more fine grained errors
911   </xs:documentation>
912   </xs:annotation>
913   <xs:restriction base="xs:string">
914     <xs:pattern value="[0-9][0-9][0-9]"/>
915   </xs:restriction>
916 </xs:simpleType>
917 <!-- 8.7 Response Reason -->
918 <xs:element name="responseDescription" type="ei:ResponseDescriptionType"/>
919 <xs:simpleType name="ResponseDescriptionType">
920   <xs:annotation>

```

```

921         <xs:documentation>Human Readable Response description. Should be standardized and language-specific.</
xs:documentation>
922     </xs:annotation>
923     <xs:restriction base="xs:string"/>
924 </xs:simpleType>
925 <!-- 8.8 Response Terms -->
926 <xs:element name="responseTermsViolated" type="ei:ArrayofResponseTermsViolated"/>
927 <xs:complexType name="ArrayOfResponseTermsViolated">
928     <xs:annotation>
929         <xs:documentation>Collection of Detailed response on Terms that cause rejection of Request</xs:documentation>
930     </xs:annotation>
931     <xs:sequence>
932         <xs:element ref="ei:responseTermViolated" minOccurs="0" maxOccurs="unbounded"/>
933     </xs:sequence>
934 </xs:complexType>
935 <!-- 8.9 Response Terms Violated -->
936 <xs:element name="responseTermViolated" type="ei:ResponseTermViolatedType"/>
937 <xs:complexType name="ResponseTermViolatedType">
938     <xs:sequence>
939         <xs:element ref="ei:responseDescription" minOccurs="1" maxOccurs="1"/>
940         <xs:element ref="emix:baseTerm" minOccurs="1" maxOccurs="1"/>
941     </xs:sequence>
942 </xs:complexType>
943 <!-- 9 Basic Semantics -->
944 <!-- 9.1 Simple Semantic Elements -->
945 <xs:element name="accuracy" type="ei:AccuracyType"/>
946 <xs:simpleType name="AccuracyType">
947     <xs:annotation>
948         <xs:documentation>Number is in same units as the payload variable for an Interval. When present with Confidence,
indicates the likely variability of the prediction. When present with ReadingType, indicates likely error of Reading.</
xs:documentation>
949     </xs:annotation>
950     <xs:restriction base="xs:float"/>
951 </xs:simpleType>
952 <xs:element name="aggregateReport" type="xs:boolean">
953     <xs:annotation>
954         <xs:documentation>Is either a request that multiple sources be combined in a report and an indication that they have
been.</xs:documentation>
955     </xs:annotation>
956 </xs:element>
957 <xs:element name="createdDateTime" type="xcal:DateTimeType">
958     <xs:annotation>
959         <xs:documentation>Date and Time this artifact was created.</xs:documentation>
960     </xs:annotation>
961 </xs:element>
962 <xs:element name="statusDateTime" type="xcal:DateTimeType">
963     <xs:annotation>
964         <xs:documentation>Date and Time this artifact references.</xs:documentation>
965     </xs:annotation>
966 </xs:element>
967 <xs:element name="confidence" type="ei:ConfidenceType"/>
968 <xs:simpleType name="ConfidenceType">
969     <xs:restriction base="xs:unsignedInt">
970         <xs:minInclusive value="0"/>
971         <xs:maxInclusive value="100"/>
972     </xs:restriction>
973 </xs:simpleType>
974 <xs:element name="deployment" type="power:EnergyItemType"/>
975 <xs:element name="modificationNumber" type="xs:unsignedInt">
976     <xs:annotation>
977         <xs:documentation>A modification number for [event]. Used to indicate if the [event] has been modified and is
incremented each time the [event] is modified </xs:documentation>

```

```

978     </xs:annotation>
979 </xs:element>
980 <xs:element name="modificationDateTime" type="xcal:DateTimeType">
981     <xs:annotation>
982         <xs:documentation>The date and time a modification takes effect.</xs:documentation>
983     </xs:annotation>
984 </xs:element>
985 <xs:element name="offline" type="xs:boolean"/>
986 <xs:element name="partyRole" type="xs:string">
987     <xs:annotation>
988         <xs:documentation>Role that Party is assuming for this interaction, e.g buyer or seller</xs:documentation>
989     </xs:annotation>
990 </xs:element>
991 <xs:element name="priority" type="xs:unsignedInt">
992     <xs:annotation>
993         <xs:documentation>This is the priority of this event relative to other events. The lower the number higher the priority. A
value of zero (0) indicates NO priority and in essence is the lowest priority by default</xs:documentation>
994     </xs:annotation>
995 </xs:element>
996 <xs:element name="rID" type="ei:rIDType"/>
997 <xs:simpleType name="rIDType">
998     <xs:annotation>
999         <xs:documentation>ID for internal use in reports</xs:documentation>
1000     </xs:annotation>
1001     <xs:restriction base="xs:string">
1002         <xs:pattern value="[0-9][0-9][0-9]"/>
1003     </xs:restriction>
1004 </xs:simpleType>
1005 <xs:attribute name="schemaVersion" type="xs:string">
1006     <xs:annotation>
1007         <xs:documentation>The version of the schema representing this entity.</xs:documentation>
1008     </xs:annotation>
1009 </xs:attribute>
1010 <xs:element name="testEvent" type="xs:string"/>
1011 <xs:element name="timeStamp" type="xcal:DateTimeType"/>
1012 <!-- 9.4 Application Extension Types -->
1013 <!-- Application Extensions are used to to provide hints or interactions with Applications running on the other side of an
interaction. They are not defined in Energy Interoperation -->
1014 <xs:element name="applicationSpecificContextBase" type="ei:ApplicationSpecificContextBaseType" abstract="true"/>
1015 <xs:complexType name="ApplicationSpecificContextBaseType" abstract="true">
1016     <xs:annotation>
1017         <xs:documentation>The Application Specific Context Base is an abstract class to exchange invariant or setup
information with an Application running on the other side of an interaction. They are not defined in Energy Interoperation,
although there are specific conformance rules that must be followed</xs:documentation>
1018     </xs:annotation>
1019     <xs:complexContent>
1020         <xs:extension base="ei:ApplicationSpecificExtensionBaseType"/>
1021     </xs:complexContent>
1022 </xs:complexType>
1023 <xs:element name="applicationSpecificPayloadBase" type="ei:ApplicationSpecificPayloadBaseType" abstract="true"/>
1024 <xs:complexType name="ApplicationSpecificPayloadBaseType" abstract="true">
1025     <xs:annotation>
1026         <xs:documentation>The Application Specific Payload Base is an abstract class to exchange feedback with an
Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific
conformance rules that must be followed</xs:documentation>
1027     </xs:annotation>
1028     <xs:complexContent>
1029         <xs:extension base="ei:ApplicationSpecificExtensionBaseType"/>
1030     </xs:complexContent>
1031 </xs:complexType>
1032 <xs:element name="applicationSpecificReportBase" type="ei:ApplicationSpecificReportBaseType" abstract="true"/>
1033 <xs:complexType name="ApplicationSpecificReportBaseType" abstract="true">

```



```
1034     <xs:annotation>
1035     <xs:documentation>The Application Specific Report Base is an abstract class to exchange feedback with an Application
running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific
conformance rules that must be followed</xs:documentation>
1036   </xs:annotation>
1037   <xs:complexContent>
1038     <xs:extension base="ei:ApplicationSpecificExtensionBaseType"/>
1039   </xs:complexContent>
1040 </xs:complexType>
1041 <xs:element name="applicationSpecificSignalBase" type="ei:ApplicationSpecificSignalBaseType" abstract="true"/>
1042 <xs:complexType name="ApplicationSpecificSignalBaseType" abstract="true">
1043   <xs:annotation>
1044     <xs:documentation>The Application Specific Signal Base is an abstract class to exchange current information with an
Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific
conformance rules that must be followed</xs:documentation>
1045   </xs:annotation>
1046   <xs:complexContent>
1047     <xs:extension base="ei:ApplicationSpecificExtensionBaseType"/>
1048   </xs:complexContent>
1049 </xs:complexType>
1050 <xs:element name="applicationSpecificExtensionBase" type="ei:ApplicationSpecificExtensionBaseType" abstract="true"/>
1051 <xs:complexType name="ApplicationSpecificExtensionBaseType" abstract="true">
1052   <xs:annotation>
1053     <xs:documentation>Application Extensions are used to provide hints to or interactions with Applications running on the
other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that
must be followed</xs:documentation>
1054   </xs:annotation>
1055 </xs:complexType>
1056 <!-- 9.5 Names -->
1057 <xs:element name="agreementName" type="xs:string">
1058   <xs:annotation>
1059     <xs:documentation>Optional Name for an Agreement between Parties, used perhaps in a user interface.</
xs:documentation>
1060   </xs:annotation>
1061 </xs:element>
1062 <xs:element name="baselineName" type="xs:string">
1063   <xs:annotation>
1064     <xs:documentation>Optional name for a Baseline, used perhaps in a user interface.</xs:documentation>
1065   </xs:annotation>
1066 </xs:element>
1067 <xs:element name="groupName" type="xs:string">
1068   <xs:annotation>
1069     <xs:documentation>Name of a Group which may be the target of an Event.</xs:documentation>
1070   </xs:annotation>
1071 </xs:element>
1072 <xs:element name="marketName" type="xs:string">
1073   <xs:annotation>
1074     <xs:documentation>Optional Name for a Market Context, used perhaps in a user interface.</xs:documentation>
1075   </xs:annotation>
1076 </xs:element>
1077 <xs:element name="partyName" type="xs:string">
1078   <xs:annotation>
1079     <xs:documentation>Optional name for a Party, used perhaps in a user interface.</xs:documentation>
1080   </xs:annotation>
1081 </xs:element>
1082 <xs:element name="reportName" type="xs:string">
1083   <xs:annotation>
1084     <xs:documentation>Optional name for a Report, used perhaps in a user interface.</xs:documentation>
1085   </xs:annotation>
1086 </xs:element>
1087 <xs:element name="signalName" type="xs:string">
1088   <xs:annotation>
```

```
1089     <xs:documentation>Optional name for a Signal, used perhaps in a user interface.</xs:documentation>
1090   </xs:annotation>
1091 </xs:element>
1092 <xs:element name="transactionName" type="xs:string">
1093   <xs:annotation>
1094     <xs:documentation>Optional name for a Transaction, used perhaps in a user interface.</xs:documentation>
1095   </xs:annotation>
1096 </xs:element>
1097 <!-- 9.6 Response Smoothing -->
1098 <xs:element name="eiResponseSmoothing" type="ei:EiResponseSmoothingType" substitutionGroup="emix:baseTerm"/>
1099 <xs:complexType name="EiResponseSmoothingType" mixed="false">
1100   <xs:annotation>
1101     <xs:documentation>Response Smoothing defines a Term that obligates the recipient to ensure that the response not be
in a single step. Response Smoothing is applied to the tolerance interval[s] indicated by the Start Before, Start After, End
Before, and End After tolerances. Response Smoothing is implimented as a Term so it can be delivered, if desired, as part of a
market context.</xs:documentation>
1102   </xs:annotation>
1103   <xs:complexContent>
1104     <xs:extension base="emix:BaseTermType">
1105       <xs:sequence>
1106         <xs:element ref="ei:eiSmoothing" minOccurs="1" maxOccurs="1"/>
1107       </xs:sequence>
1108     </xs:extension>
1109   </xs:complexContent>
1110 </xs:complexType>
1111 <xs:element name="eiSmoothing" type="ei:EiSmoothingType"/>
1112 <xs:simpleType name="EiSmoothingType">
1113   <xs:annotation>
1114     <xs:documentation>Smoothing in EventSignal to indicate a requirement that a response not be in a single step.
Smoothing is applied to the interval[s] defined by the Start Before, Start After, End Before, and End after tolerances.</
xs:documentation>
1115   </xs:annotation>
1116   <xs:restriction base="xs:token">
1117     <xs:enumeration value="ramp">
1118       <xs:annotation>
1119         <xs:documentation>A smooth or uniform step ramp is indicated between the initial and end values in the respective
Tolerance Interval.</xs:documentation>
1120       </xs:annotation>
1121     </xs:enumeration>
1122     <xs:enumeration value="uniform">
1123       <xs:annotation>
1124         <xs:documentation>A uniform distribution is indicated over the entire respective Tolerance Interval.</
xs:documentation>
1125       </xs:annotation>
1126     </xs:enumeration>
1127     <xs:enumeration value="none">
1128       <xs:annotation>
1129         <xs:documentation>No specific smoothing is indicated. Applications need not react in a stepwise manner, so some
degree of smoothing MAY occur in response to this request. If the Smoothing Term is absent, the behavior requested is the
same as None.</xs:documentation>
1130       </xs:annotation>
1131     </xs:enumeration>
1132   </xs:restriction>
1133 </xs:simpleType>
1134 <!-- 9.7 WS-Calendar extensions -->
1135 <!-- 9.7.1 Active Period is extension of WS-Calendar schedule -->
1136 <xs:element name="eiActivePeriod" type="xcal:VcalendarType">
1137   <xs:annotation>
1138     <xs:documentation>Event Schedule contains sequence containing an Active Period (at least) and potentially other
components</xs:documentation>
1139   </xs:annotation>
1140 </xs:element>
```

```

1141 <!-- 9.7.2 Degenerate Intervals used in Active Period Definition -->
1142 <xs:element name="x-eiNotification" type="xcal:DurationPropType" substitutionGroup="xcal:baseProperty"/>
1143 <xs:element name="x-eiRampUp" type="xcal:DurationPropType" substitutionGroup="xcal:baseProperty"/>
1144 <xs:element name="x-eiRecovery" type="xcal:DurationPropType" substitutionGroup="xcal:baseProperty"/>
1145 <!-- 9.7.3 Qualification of Information Payloads -->
1146 <xs:element name="x-IntervalQualification" type="ei:X-IntervalQualificationType" substitutionGroup="xcal:baseProperty"/>
1147 <xs:complexType name="X-IntervalQualificationType" mixed="false">
1148   <xs:annotation>
1149     <xs:documentation>Interval Qualification is a WS-Calendar derived property that conveys several values that indicate
something about interpreting the accuracy of the information in the payload. An application that relies on this information
should profile precise meanings of these values within their domain.</xs:documentation>
1150   </xs:annotation>
1151   <xs:complexContent>
1152     <xs:extension base="xcal:BasePropertyType">
1153       <xs:sequence>
1154         <xs:choice minOccurs="0" maxOccurs="1">
1155           <xs:element ref="ei:confidence" maxOccurs="1"/>
1156           <xs:element ref="ei:readingType" maxOccurs="1"/>
1157         </xs:choice>
1158         <xs:element ref="ei:accuracy" minOccurs="0" maxOccurs="1"/>
1159       </xs:sequence>
1160     </xs:extension>
1161   </xs:complexContent>
1162 </xs:complexType>
1163 <!-- 9.7.4 Interval for use in Streams -->
1164 <xs:element name="interval" type="ei:IntervalType" substitutionGroup="strm:streamInterval"/>
1165 <xs:complexType name="IntervalType">
1166   <xs:annotation>
1167     <xs:documentation>This is the type for an Interval in a Signal / Baseline / Report Stream.</xs:documentation>
1168   </xs:annotation>
1169   <xs:complexContent>
1170     <xs:extension base="strm:StreamIntervalType"/>
1171   </xs:complexContent>
1172 </xs:complexType>
1173 <!-- 9.8 Identifiers -->
1174 <!-- 9.8.1.1 Unclassified ID Types -->
1175 <!-- 9.8.1.1 Unclassified Complex ID Types -->
1176 <xs:element name="qualifiedEventID" type="ei:QualifiedEventIDType"/>
1177 <xs:complexType name="QualifiedEventIDType">
1178   <xs:annotation>
1179     <xs:documentation>Fully Qualified Event ID includes the eventID and the Modification Number</xs:documentation>
1180   </xs:annotation>
1181   <xs:sequence>
1182     <xs:element ref="ei:eventID" minOccurs="1" maxOccurs="1"/>
1183     <xs:element ref="ei:modificationNumber" minOccurs="0" maxOccurs="1"/>
1184   </xs:sequence>
1185 </xs:complexType>
1186 <!-- 9.8.1.2 Unclassified Simple UID Types -->
1187 <xs:element name="eventID" substitutionGroup="ei:uid">
1188   <xs:annotation>
1189     <xs:documentation>Identifier for an Event State interaction</xs:documentation>
1190   </xs:annotation>
1191 </xs:element>
1192 <xs:element name="eventStateID" substitutionGroup="ei:uid">
1193   <xs:annotation>
1194     <xs:documentation>Identifier for an Event State interaction</xs:documentation>
1195   </xs:annotation>
1196 </xs:element>
1197 <xs:element name="groupID" substitutionGroup="ei:uid">
1198   <xs:annotation>
1199     <xs:documentation>Identifier of a Group which may be the target of an Event.</xs:documentation>
1200   </xs:annotation>

```

```
1201     </xs:element>
1202     <xs:element name="resourceID" substitutionGroup="ei:uid">
1203         <xs:annotation>
1204             <xs:documentation>Identifier for a Resource. Resources are associated with a VEN during Enrollment</
xs:documentation>
1205         </xs:annotation>
1206     </xs:element>
1207     <!-- 9.8.2 Reference IDs -->
1208     <xs:element name="acceptanceID" substitutionGroup="ei:refID">
1209         <xs:annotation>
1210             <xs:documentation>Identifier for Acceptance of a Quote.</xs:documentation>
1211         </xs:annotation>
1212     </xs:element>
1213     <xs:element name="availID" substitutionGroup="ei:refID">
1214         <xs:annotation>
1215             <xs:documentation>Identifier for Avail in an interaction.</xs:documentation>
1216         </xs:annotation>
1217     </xs:element>
1218     <xs:element name="baselineID" substitutionGroup="ei:refID">
1219         <xs:annotation>
1220             <xs:documentation>Identifier for a Baseline.</xs:documentation>
1221         </xs:annotation>
1222     </xs:element>
1223     <xs:element name="constraintID" substitutionGroup="ei:refID">
1224         <xs:annotation>
1225             <xs:documentation>This is the identifier that may used by other entities to refer to this instance of an EiConstraint.</
xs:documentation>
1226         </xs:annotation>
1227     </xs:element>
1228     <xs:element name="eiDeliveryID" substitutionGroup="ei:refID">
1229         <xs:annotation>
1230             <xs:documentation>Reference ID for a Delivery</xs:documentation>
1231         </xs:annotation>
1232     </xs:element>
1233     <xs:element name="eiReportID" substitutionGroup="ei:refID">
1234         <xs:annotation>
1235             <xs:documentation>Reference ID for a Report</xs:documentation>
1236         </xs:annotation>
1237     </xs:element>
1238     <xs:element name="eventMessageID" substitutionGroup="ei:refID">
1239         <xs:annotation>
1240             <xs:documentation>Identifier assigned to the message (EiEvent) describing an event and it associated signals. </
xs:documentation>
1241         </xs:annotation>
1242     </xs:element>
1243     <xs:element name="historianID" substitutionGroup="ei:refID">
1244         <xs:annotation>
1245             <xs:documentation>Reference ID for an ongoing Historian</xs:documentation>
1246         </xs:annotation>
1247     </xs:element>
1248     <xs:element name="optID" substitutionGroup="ei:refID">
1249         <xs:annotation>
1250             <xs:documentation>Identifier for an Opt interaction</xs:documentation>
1251         </xs:annotation>
1252     </xs:element>
1253     <xs:element name="originalReferenceID" substitutionGroup="ei:refID">
1254         <xs:annotation>
1255             <xs:documentation>Identifier for a superceded Reference ID</xs:documentation>
1256         </xs:annotation>
1257     </xs:element>
1258     <xs:element name="quoteID" substitutionGroup="ei:refID">
1259         <xs:annotation>
```

```
1260     <xs:documentation>Identifier for a Quote</xs:documentation>
1261   </xs:annotation>
1262 </xs:element>
1263 <xs:element name="registrationID" substitutionGroup="ei:refID">
1264   <xs:annotation>
1265     <xs:documentation>Identifier for Registration transaction</xs:documentation>
1266   </xs:annotation>
1267 </xs:element>
1268 <xs:element name="reportSpecifierID" substitutionGroup="ei:refID">
1269   <xs:annotation>
1270     <xs:documentation>Identifier for a particular Report Specification</xs:documentation>
1271   </xs:annotation>
1272 </xs:element>
1273 <xs:element name="reportRequestID" substitutionGroup="ei:refID">
1274   <xs:annotation>
1275     <xs:documentation>Identifier for a particular Report Request</xs:documentation>
1276   </xs:annotation>
1277 </xs:element>
1278 <xs:element name="signalID" substitutionGroup="ei:refID">
1279   <xs:annotation>
1280     <xs:documentation>Identifier for a particular Signal</xs:documentation>
1281   </xs:annotation>
1282 </xs:element>
1283 <xs:element name="tenderID" substitutionGroup="ei:refID">
1284   <xs:annotation>
1285     <xs:documentation>Identifier for a market Tender</xs:documentation>
1286   </xs:annotation>
1287 </xs:element>
1288 <xs:element name="transactionID" substitutionGroup="ei:refID">
1289   <xs:annotation>
1290     <xs:documentation>Identification of Transaction</xs:documentation>
1291   </xs:annotation>
1292 </xs:element>
1293 <!-- 9.8.2.9 Base Type for References -->
1294 <xs:element name="refID" substitutionGroup="ei:uid">
1295   <xs:annotation>
1296     <xs:documentation>Reference ID for a particular instance, transmittal, or artifact. Note: not the same as the native ID of
the object being transmitted or shared.</xs:documentation>
1297   </xs:annotation>
1298 </xs:element>
1299 <!-- 9.8.3 Parties -->
1300 <xs:element name="counterPartyID" substitutionGroup="ei:actorID">
1301   <xs:annotation>
1302     <xs:documentation>The "other" party in any interaction</xs:documentation>
1303   </xs:annotation>
1304 </xs:element>
1305 <xs:element name="initiatingPartyID" substitutionGroup="ei:actorID">
1306   <xs:annotation>
1307     <xs:documentation>The party requesting or starting an interaction</xs:documentation>
1308   </xs:annotation>
1309 </xs:element>
1310 <xs:element name="partyID" substitutionGroup="ei:actorID">
1311   <xs:annotation>
1312     <xs:documentation>Identifier for any Party</xs:documentation>
1313   </xs:annotation>
1314 </xs:element>
1315 <xs:element name="publisherPartyID" substitutionGroup="ei:actorID">
1316   <xs:annotation>
1317     <xs:documentation>Identifier for the party publishing a broadcast message</xs:documentation>
1318   </xs:annotation>
1319 </xs:element>
1320 <xs:element name="registreePartyID" substitutionGroup="ei:actorID">
```



```
1321     <xs:annotation>
1322         <xs:documentation>Identifier for Party attempting to Register</xs:documentation>
1323     </xs:annotation>
1324 </xs:element>
1325 <xs:element name="registrarPartyID" substitutionGroup="ei:actorID">
1326     <xs:annotation>
1327         <xs:documentation>Identifier of Party acting as a Registrar</xs:documentation>
1328     </xs:annotation>
1329 </xs:element>
1330 <xs:element name="requestorPartyID" substitutionGroup="ei:actorID">
1331     <xs:annotation>
1332         <xs:documentation>Identifier of Party making a Request</xs:documentation>
1333     </xs:annotation>
1334 </xs:element>
1335 <xs:element name="respondingPartyID" substitutionGroup="ei:actorID">
1336     <xs:annotation>
1337         <xs:documentation>Identifier of Party making a Resonse (note: in CancelledPartyRegistration payload and I do not
know defintion)</xs:documentation>
1338     </xs:annotation>
1339 </xs:element>
1340 <xs:element name="subscriberPartyID" substitutionGroup="ei:actorID">
1341     <xs:annotation>
1342         <xs:documentation>Identifier of Party subscribing to a broadcast or service</xs:documentation>
1343     </xs:annotation>
1344 </xs:element>
1345 <xs:element name="venID" substitutionGroup="ei:actorID">
1346     <xs:annotation>
1347         <xs:documentation>Identifier of Party acting as a VEN.</xs:documentation>
1348     </xs:annotation>
1349 </xs:element>
1350 <xs:element name="vtnID" substitutionGroup="ei:actorID">
1351     <xs:annotation>
1352         <xs:documentation>Identifier of Party acting as a VTN.</xs:documentation>
1353     </xs:annotation>
1354 </xs:element>
1355 <!-- 9.8.3.9 Base Party ID -->
1356 <xs:element name="actorID" abstract="true" substitutionGroup="ei:uid">
1357     <xs:annotation>
1358         <xs:documentation>Identifier of a Party. May be ws-addressing endpoint descriptor.</xs:documentation>
1359     </xs:annotation>
1360 </xs:element>
1361 <!-- 9.8.9 Base UID -->
1362 <xs:element name="uid" type="ei:UidType" abstract="true"/>
1363 <xs:simpleType name="UidType">
1364     <xs:annotation>
1365         <xs:documentation>Unique Identifier</xs:documentation>
1366     </xs:annotation>
1367     <xs:restriction base="xs:string"/>
1368 </xs:simpleType>
1369 <!-- 9.9 Standard Enumerations -->
1370 <!-- 9.9.1 Event Status -->
1371 <xs:element name="eventStatus" type="ei:EventStatusType"/>
1372 <xs:simpleType name="EventStatusType">
1373     <xs:annotation>
1374         <xs:documentation>Indicates the current status of an event.</xs:documentation>
1375     </xs:annotation>
1376     <xs:union memberTypes="ei:EventStatusEnumeratedType ei:EiExtensionTokenType"/>
1377 </xs:simpleType>
1378 <xs:element name="eventStatusEnumerated" type="ei:EventStatusEnumeratedType"/>
1379 <xs:simpleType name="EventStatusEnumeratedType">
1380     <xs:restriction base="xs:token">
1381         <xs:enumeration value="none">
```

```

1382     <xs:annotation>
1383         <xs:documentation>No event pending</xs:documentation>
1384     </xs:annotation>
1385 </xs:enumeration>
1386 <xs:enumeration value="far">
1387     <xs:annotation>
1388         <xs:documentation>event pending in the far future. The exact definition of how far in the future this refers is
dependent upon the market context, but typically means the next day.</xs:documentation>
1389     </xs:annotation>
1390 </xs:enumeration>
1391 <xs:enumeration value="near">
1392     <xs:annotation>
1393         <xs:documentation>event pending in the near future. The exact definition of how near in the future the pending
event is active is dependent on the market context</xs:documentation>
1394     </xs:annotation>
1395 </xs:enumeration>
1396 <xs:enumeration value="active">
1397     <xs:annotation>
1398         <xs:documentation>The event has been initiated and is currently active.</xs:documentation>
1399     </xs:annotation>
1400 </xs:enumeration>
1401 <xs:enumeration value="completed">
1402     <xs:annotation>
1403         <xs:documentation>The event has completed.</xs:documentation>
1404     </xs:annotation>
1405 </xs:enumeration>
1406 <xs:enumeration value="cancelled">
1407     <xs:annotation>
1408         <xs:documentation>The event has been canceled.</xs:documentation>
1409     </xs:annotation>
1410 </xs:enumeration>
1411 </xs:restriction>
1412 </xs:simpleType>
1413 <!-- 9.9.2 Reading Type -->
1414 <xs:element name="readingType" type="ei:ReadingTypeType"/>
1415 <xs:simpleType name="ReadingTypeType">
1416     <xs:annotation>
1417         <xs:documentation>Type of Reading.</xs:documentation>
1418     </xs:annotation>
1419     <xs:union memberTypes="ei:ReadingTypeEnumeratedType ei:EiExtensionTokenType"/>
1420 </xs:simpleType>
1421 <xs:element name="readingTypeEnumerated" type="ei:ReadingTypeEnumeratedType"/>
1422 <xs:simpleType name="ReadingTypeEnumeratedType">
1423     <xs:restriction base="xs:token">
1424         <xs:enumeration value="Direct Read">
1425             <xs:annotation>
1426                 <xs:documentation>Reading is read from a device that increases monotonically, and usage must be computed
from pairs of start and stop readings.</xs:documentation>
1427             </xs:annotation>
1428         </xs:enumeration>
1429         <xs:enumeration value="Net">
1430             <xs:annotation>
1431                 <xs:documentation>Meter or [resource] prepares its own calculation of total use over time.</xs:documentation>
1432             </xs:annotation>
1433         </xs:enumeration>
1434         <xs:enumeration value="Allocated">
1435             <xs:annotation>
1436                 <xs:documentation>Meter covers several [resources] and usage is inferred through some sort of pro rata
computation.</xs:documentation>
1437             </xs:annotation>
1438         </xs:enumeration>
1439         <xs:enumeration value="Estimated">

```

```

1440         <xs:annotation>
1441             <xs:documentation>Used when a reading is absent in a series in which most readings are present.</
xs:documentation>
1442         </xs:annotation>
1443     </xs:enumeration>
1444     <xs:enumeration value="Summed">
1445         <xs:annotation>
1446             <xs:documentation>Several meters together provide the reading for this [resource]. This is specifically a different
than aggregated, which refers to multiple [resources] in the same payload. See also Hybrid.</xs:documentation>
1447         </xs:annotation>
1448     </xs:enumeration>
1449     <xs:enumeration value="Derived">
1450         <xs:annotation>
1451             <xs:documentation>Usage is inferred through knowledge of run-time, normal operation, etc.</xs:documentation>
1452         </xs:annotation>
1453     </xs:enumeration>
1454     <xs:enumeration value="Mean">
1455         <xs:annotation>
1456             <xs:documentation>Reading is the mean value over the period indicated in Granularity</xs:documentation>
1457         </xs:annotation>
1458     </xs:enumeration>
1459     <xs:enumeration value="Peak">
1460         <xs:annotation>
1461             <xs:documentation>Reading is Peak (highest) value over the period indicated in granularity. For some
measurements, it may make more sense as the lowest value. May not be consistent with aggregate readings. Only valid for
flow-rate Item Bases, i.e., Power not Energy.</xs:documentation>
1462         </xs:annotation>
1463     </xs:enumeration>
1464     <xs:enumeration value="Hybrid">
1465         <xs:annotation>
1466             <xs:documentation>If aggregated, refers to different reading types in the aggregate number.</xs:documentation>
1467         </xs:annotation>
1468     </xs:enumeration>
1469     <xs:enumeration value="Contract">
1470         <xs:annotation>
1471             <xs:documentation>Indicates reading is pro forma, i.e., is reported at agreed upon rates</xs:documentation>
1472         </xs:annotation>
1473     </xs:enumeration>
1474     <xs:enumeration value="Projected">
1475         <xs:annotation>
1476             <xs:documentation>Indicates reading is in the future, and has not yet been measured.</xs:documentation>
1477         </xs:annotation>
1478     </xs:enumeration>
1479 </xs:restriction>
1480 </xs:simpleType>
1481 <!-- 9.9.3 Opt Reasons -->
1482 <xs:element name="optReason" type="ei:OptReasonType"/>
1483 <xs:simpleType name="OptReasonType">
1484     <xs:annotation>
1485         <xs:documentation>Reason for Opting.</xs:documentation>
1486     </xs:annotation>
1487     <xs:union memberTypes="ei:OptReasonEnumeratedType ei:EiExtensionTokenType"/>
1488 </xs:simpleType>
1489 <xs:element name="optReasonEnumerated" type="ei:OptReasonEnumeratedType"/>
1490 <xs:simpleType name="OptReasonEnumeratedType">
1491     <xs:annotation>
1492         <xs:documentation>Enumerated Reasons for Opting.</xs:documentation>
1493     </xs:annotation>
1494     <xs:restriction base="xs:token">
1495         <xs:enumeration value="economic"/>
1496         <xs:enumeration value="emergency"/>
1497         <xs:enumeration value="mustRun"/>

```

```

1498     <xs:enumeration value="notParticipating"/>
1499     <xs:enumeration value="outageRunStatus"/>
1500     <xs:enumeration value="overrideStatus"/>
1501     <xs:enumeration value="participating"/>
1502 </xs:restriction>
1503 </xs:simpleType>
1504 <!-- 9.9.4 Opt Type -->
1505 <xs:element name="optType" type="ei:OptTypeType"/>
1506 <xs:simpleType name="OptTypeType">
1507     <xs:restriction base="xs:token">
1508         <xs:enumeration value="optIn"/>
1509         <xs:enumeration value="optOut"/>
1510     </xs:restriction>
1511 </xs:simpleType>
1512 <!-- 9.9.5 Enumerations for Signals -->
1513 <xs:element name="signalType" type="ei:SignalTypeType"/>
1514 <xs:simpleType name="SignalTypeType">
1515     <xs:annotation>
1516         <xs:documentation>SignalType is used in EventSignals to specify the Payload Types in a Signal.</xs:documentation>
1517     </xs:annotation>
1518     <xs:union memberTypes="ei:SignalTypeEnumeratedType ei:EiExtensionTokenType"/>
1519 </xs:simpleType>
1520 <xs:element name="signalTypeEnumerated" type="ei:SignalTypeEnumeratedType"/>
1521 <xs:simpleType name="SignalTypeEnumeratedType">
1522     <xs:annotation>
1523         <xs:documentation>SignalTypeEnumerated lists the pre-defined Types used to specify the Payload Types and
conformance in a Stream</xs:documentation>
1524     </xs:annotation>
1525     <xs:restriction base="xs:token">
1526         <xs:enumeration value="delta">
1527             <xs:annotation>
1528                 <xs:documentation>Signal indicates the amount to change (denominated in Itembase or in the EMIX Product) from
what one would have used without the Signal. This may or may not be accompanied by a baseline. Payload Type Quantity</
xs:documentation>
1529             </xs:annotation>
1530         </xs:enumeration>
1531         <xs:enumeration value="level">
1532             <xs:annotation>
1533                 <xs:documentation>Signal indicates a Program Level. Payload Type is Program Level</xs:documentation>
1534             </xs:annotation>
1535         </xs:enumeration>
1536         <xs:enumeration value="multiplier">
1537             <xs:annotation>
1538                 <xs:documentation>Signal indicates a multiplier applied to the current rate of delivery or usage (denominated in
Itembase or in the EMIX Product) from what one would have used without the Signal. This may or may not be accompanied by
a baseline. Payload Type is Float</xs:documentation>
1539             </xs:annotation>
1540         </xs:enumeration>
1541         <xs:enumeration value="price">
1542             <xs:annotation>
1543                 <xs:documentation>Signal indicates the Price. Extended Price is the value multiplied by the number of units units
(denominated in Itembase or in the EMIX Product). Payload Type is emix:price</xs:documentation>
1544             </xs:annotation>
1545         </xs:enumeration>
1546         <xs:enumeration value="priceMultiplier">
1547             <xs:annotation>
1548                 <xs:documentation>Signal indicates the Price Multiplier. Extended Price is the computed price (as described in
EMIX) the value multiplied by the number of units units (denominated in Itembase or in the EMIX Product). Payload Type is
emix:priceMultiplier</xs:documentation>
1549             </xs:annotation>
1550         </xs:enumeration>
1551         <xs:enumeration value="priceRelative">

```

```

1552         <xs:annotation>
1553         <xs:documentation>Signal indicates the Relative Price. Extended Price is the computed price (as described in
EMIX) the value multiplied by the number of units units (denominated in Itembase or in the EMIX Product). Payload Type is
emix:priceRelative</xs:documentation>
1554     </xs:annotation>
1555 </xs:enumeration>
1556 <xs:enumeration value="product">
1557     <xs:annotation>
1558     <xs:documentation>Signal indicates the Product for each interval. Payload Type is an EMIX Product Description</
xs:documentation>
1559     </xs:annotation>
1560 </xs:enumeration>
1561 <xs:enumeration value="setpoint">
1562     <xs:annotation>
1563     <xs:documentation>Signal indicates a target amount of units (denominated in Itembase or in the EMIX Product).
Payload Type is Quantity</xs:documentation>
1564     </xs:annotation>
1565 </xs:enumeration>
1566 </xs:restriction>
1567 </xs:simpleType>
1568 <!-- 9.9.8 Report Type -->
1569 <xs:element name="reportType" type="ei:ReportTypeType"/>
1570 <xs:simpleType name="ReportTypeType">
1571     <xs:annotation>
1572     <xs:documentation>An enumerated value that gives the type of report being provided.</xs:documentation>
1573     </xs:annotation>
1574     <xs:union memberTypes="ei:ReportEnumeratedType ei:EiExtensionTokenType"/>
1575 </xs:simpleType>
1576 <xs:element name="reportEnumerated" type="ei:ReportEnumeratedType"/>
1577 <xs:simpleType name="ReportEnumeratedType">
1578     <xs:annotation>
1579     <xs:documentation>Enumerated Report types</xs:documentation>
1580     </xs:annotation>
1581     <xs:restriction base="xs:token">
1582     <xs:enumeration value="reading">
1583     <xs:annotation>
1584     <xs:documentation>Report indicates a Reading, as from a meter. Readings are moments in time--changes over
time can be computed from the difference between successive readings. Payload Type is Float</xs:documentation>
1585     </xs:annotation>
1586 </xs:enumeration>
1587 <xs:enumeration value="usage">
1588     <xs:annotation>
1589     <xs:documentation>Report indicates an amount of units (denominated in Itembase or in the EMIX Product) over a
period. Payload Type is Quantity. A typical ItemBase is Real Energy.</xs:documentation>
1590     </xs:annotation>
1591 </xs:enumeration>
1592 <xs:enumeration value="demand">
1593     <xs:annotation>
1594     <xs:documentation>Report indicates an amount of units (denominated in Itembase or in the EMIX Product).
Payload Type is Quantity. A typical ItemBase is Real Power.</xs:documentation>
1595     </xs:annotation>
1596 </xs:enumeration>
1597 <xs:enumeration value="setPoint">
1598     <xs:annotation>
1599     <xs:documentation>Report indicates the amount (denominated in Itembase or in the EMIX Product) currently set.
May be a confirmation/return of the setpoint control value sent from the VTN. Payload Type is Quantity. A typical ItemBase is
Real Power.</xs:documentation>
1600     </xs:annotation>
1601 </xs:enumeration>
1602 <xs:enumeration value="deltaUsage">
1603     <xs:annotation>
1604     <xs:documentation>Change in Usage as compared to the Baseline. See usage for more information</

```



```

xs:documentation>
1605     </xs:annotation>
1606     </xs:enumeration>
1607     <xs:enumeration value="deltaSetPoint">
1608         <xs:annotation>
1609             <xs:documentation>Changes in Setpoint from previous schedule.</xs:documentation>
1610         </xs:annotation>
1611     </xs:enumeration>
1612     <xs:enumeration value="deltaDemand">
1613         <xs:annotation>
1614             <xs:documentation>Change in Demand as compared to the Baseline. See Demand for more information</
xs:documentation>
1615         </xs:annotation>
1616     </xs:enumeration>
1617     <xs:enumeration value="baseline">
1618         <xs:annotation>
1619             <xs:documentation>Can be Demand or Usage, as indicated by ItemBase. Indicates what [measurement] would be
if not for the Event or Regulation. Report is of the format Baseline.</xs:documentation>
1620         </xs:annotation>
1621     </xs:enumeration>
1622     <xs:enumeration value="deviation">
1623         <xs:annotation>
1624             <xs:documentation>Difference between some instruction and actual state.</xs:documentation>
1625         </xs:annotation>
1626     </xs:enumeration>
1627     <xs:enumeration value="avgUsage">
1628         <xs:annotation>
1629             <xs:documentation>Average usage over the duration indicated by the Granularity. See usage for more information,
</xs:documentation>
1630         </xs:annotation>
1631     </xs:enumeration>
1632     <xs:enumeration value="avgDemand">
1633         <xs:annotation>
1634             <xs:documentation>Average usage over the duration indicated by the Granularity. See demand for more
information.</xs:documentation>
1635         </xs:annotation>
1636     </xs:enumeration>
1637     <xs:enumeration value="operatingState">
1638         <xs:annotation>
1639             <xs:documentation>Generalized state of a resource such as on/off, occupancy of building, etc. No ItemBase is
relevant. Requires an Application Specific Payload Extension.</xs:documentation>
1640         </xs:annotation>
1641     </xs:enumeration>
1642     <xs:enumeration value="upRegulationCapacityAvailable">
1643         <xs:annotation>
1644             <xs:documentation>Up Regulation capacity available for dispatch, expressed in EMIX Real Power. Payload is
always expressed as positive Quantity.</xs:documentation>
1645         </xs:annotation>
1646     </xs:enumeration>
1647     <xs:enumeration value="downRegulationCapacityAvailable">
1648         <xs:annotation>
1649             <xs:documentation>Down Regulation capacity available for dispatch, expressed in EMIX Real Power. Payload is
always expressed as positive Quantity.</xs:documentation>
1650         </xs:annotation>
1651     </xs:enumeration>
1652     <xs:enumeration value="regulationSetpoint">
1653         <xs:annotation>
1654             <xs:documentation>Regulation setpoint as instructed as part of regulation services</xs:documentation>
1655         </xs:annotation>
1656     </xs:enumeration>
1657     <xs:enumeration value="storedEnergy">
1658         <xs:annotation>

```

```
1659         <xs:documentation>Stored Energy is expressed as as Real Energy and Payload is expressed as a Quantity.</
xs:documentation>
1660     </xs:annotation>
1661 </xs:enumeration>
1662 <xs:enumeration value="targetEnergyStorage">
1663     <xs:annotation>
1664         <xs:documentation>Target Energy is expressed as as Real Energy and Payload is expressed as a Quantity.</
xs:documentation>
1665     </xs:annotation>
1666 </xs:enumeration>
1667 <xs:enumeration value="availableEnergyStorage">
1668     <xs:annotation>
1669         <xs:documentation>Capacity available for further energy storage, perhaps to get to Target Energy Storage</
xs:documentation>
1670     </xs:annotation>
1671 </xs:enumeration>
1672 <xs:enumeration value="price">
1673     <xs:annotation>
1674         <xs:documentation>Price per ItemBase at each Interval</xs:documentation>
1675     </xs:annotation>
1676 </xs:enumeration>
1677 <xs:enumeration value="level">
1678     <xs:annotation>
1679         <xs:documentation>Simple level from market at each Interval. Itemabse is not relevant.</xs:documentation>
1680     </xs:annotation>
1681 </xs:enumeration>
1682 </xs:restriction>
1683 </xs:simpleType>
1684 <!-- 9.99 Extension Types -->
1685 <xs:simpleType name="EiExtensionTokenType">
1686     <xs:annotation>
1687         <xs:documentation>Pattern used for extending string enumeration, where allowed</xs:documentation>
1688     </xs:annotation>
1689     <xs:restriction base="xs:token">
1690         <xs:pattern value="x-\S.*"/>
1691     </xs:restriction>
1692 </xs:simpleType>
1693 <xs:simpleType name="EiExtensionStringType">
1694     <xs:annotation>
1695         <xs:documentation>Pattern used for extending string enumeration, where allowed</xs:documentation>
1696     </xs:annotation>
1697     <xs:restriction base="xs:string">
1698         <xs:pattern value="x-\S.*"/>
1699     </xs:restriction>
1700 </xs:simpleType>
1701 </xs:schema>
1702
```