Energy Market Information Exchange (EMIX) Version 1.0

Committee Specification Draft 02

28 April 2011

Specification URIs:
This version:
> http://docs.oasis-open.org/emix/emix/v1.0/csd02/emix-v1.0-csd02.doc (Authoritative)
> http://docs.oasis-open.org/emix/emix/v1.0/csd02/emix-v1.0-csd02.html
> http://docs.oasis-open.org/emix/emix/v1.0/csd02/emix-v1.0-csd02.pdf

Previous version:
> http://docs.oasis-open.org/emix/emix/v1.0/csprd01/emix-v1.0-csprd01.doc (Authoritative)
> http://docs.oasis-open.org/emix/emix/v1.0/csprd01/emix-v1.0-csprd01.html
> http://docs.oasis-open.org/emix/emix/v1.0/csprd01/emix-v1.0-csprd01.pdf

Latest version:
> http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.doc (Authoritative)
> http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.html
> http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.pdf

Technical Committee:
> OASIS Energy Market Information Exchange (EMIX) TC

Chair(s):
> William Cox, Individual
> Edward Cazalet, Individual

Editor(s):
> Toby Considine, University of North Carolina at Chapel Hill

Related work:
> This specification is related to:
> - OASIS Specification WS-Calendar V1.0, in process
> - OASIS Specification Energy Interoperation V1.0, in process
> - XML schema(s): emix/v1.0/csd02/xsd/

Declared XML namespace(s):
> http://docs.oasis-open.org/ns/emix
> http://docs.oasis-open.org/ns/emix/power
> http://docs.oasis-open.org/ns/emix/power/resource

Abstract:
> The data models and XML vocabularies defined by this TC will address issues in energy markets and the Smart Grid, but are defined so as to support requirements for other markets. The TC will develop an information model and XML vocabulary to exchange prices and product definitions for transactive energy markets.

- Price information
- Bid information
- Time for use or availability
- Units and quantity to be traded
- Characteristics of what is traded

The definition of a price and of other market information exchanged depends on the market context in which it exists. It is not in scope for this TC to define specifications for markets, nor how prices are determined, nor the mechanisms for interoperation.

Status:

This document was last revised or approved by the OASIS Energy Market Information Exchange (EMIX) TC on the above date. The level of approval is also listed above. Check the "Latest Version" location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at http://www.oasis-open.org/committees/emix/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/emix/ipr.php).

Citation format:

When referencing this specification the following citation format should be used:

[EMIX-v1.0]

Energy Market Information Exchange (EMIX) Version 1.0. 28 April, 2011. OASIS Committee Specification Draft 02. http://docs.oasis-open.org/emix/emix/v1.0/csd02/emix-v1.0-csd02.doc

Notices

Copyright © OASIS Open 2011. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS" and "EMIX" are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/who/trademark.php for above guidance.

Table of Contents

Tables, Figures & Examples

Index to Figures

Index to Tables

1 Introduction

This document defines an information model to exchange Price and Product information for power and energy markets. Product definition includes quantity and quality of supply as well as attributes of interest to consumers distinguishing between power and energy sources. Energy Market Information Exchange (EMIX) is not intended as a stand-alone signal; rather, it is anticipated to be used for information exchange in a variety of market-oriented interactions.

The EMIX Technical Committee (TC) is developing this specification in support of the US Department of Commerce National Institute of Standards and Technology (NIST) NIST Framework and Roadmap for Smart Grid Interoperability Standards **[NIST]** and in support of the US Department of Energy (DOE) as described in the Energy Independence and Security Act of 2007 (EISA 2007) **[EISA]**.

This specification defines the following:

- The characteristics of power and energy that along with price define a product
- An **[XML Schema]** for Price and Product definition for products whose value varies with time of delivery.
- An **[XML Schema]** for Price and Product definition for Power-related products and services.
- An **[XML Schema]** describing the capabilities of resources that are being offered to the market.

Key to reading the document:

- **BOLD** terms are the names of referenced standards
- *Italic phrases* are quotes from external material.
- **[bracketed]** are references to the standards listed in listed in the normative or non-normative sections references sections.
- All examples and all Appendices are non-normative.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [**RFC2119**].

1.2 Process

This information exchange was developed primarily by integrating requirements and use cases for Price and Product definition developed by the North American Energy Standards Board (NAESB) as part of its response to NIST Priority Action Plan 03 (PAP03), "Develop Common Specification for Price and Product Definition" **[PAP03]**, which was driven by NIST, Federal Energy Regulatory Commission (FERC), and DOE priority items.

Where appropriate, semantic elements from the International Electrotechnical Commission (IEC) Technical Committee (TC) 57 Power systems management and associated information exchange Common Information Model (CIM) are used **[IEC]**. Business and market information was borrowed from the financial instruments Common Information Models as described in International Standards Organization (ISO) **[ISO20022]** standard and in the financial trading protocol, **[FIX]** (Financial Information eXchange).

Both the supply and the use of energy products, and therefore the market value, are time dependent, so precise communication of time of delivery is a significant component of product definition. EMIX incorporates schedule and interval communication interfaces from Web Services Calendar (**[WS-Calendar]**) to communicate schedule-related information.

43 Additional guidance was drawn from subject matter experts familiar with the design and implementation of
44 enterprise and other systems that may interact with smart grids.

1.3 Normative References

46	**RFC2119**	S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*,
47		http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.
48	**CEFACT**	*United Nations Centre for Trade Facilitation and Electronic Business*, Currency
49		codes, ISO 4217 3A - Code List Schema Module
50		http://www.unece.org/uncefact/codelist/standard/ISO_ISO3AlphaCurrencyCode_
51		20100407.xsd
52	**GML**	L van den Brink, C Portele, P. Vretanos *Geography Markup Language (GML)*
53		*simple features profile*, OpenGIS® Implementation Standard, GML 3.2 Profile,
54		Version 2.0, October 2010, http://schemas.opengis.net/gml/3.2.1/gml.xsd
55	**SOA-RM**	M MacKenzie, K Laskey, F McCabe, P Brown, R Metz, *OASIS Reference Model*
56		*for Service Oriented Architecture 1.0*, October 2006 http://docs.oasis-
57		open.org/soa-rm/v1.0/
58	**UML**	*Unified Modeling Language (UML), Version 2.2*, Object Management Group,
59		February, 2009, http://www.omg.org/spec/UML/2.2/
60	**URI**	T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifier (URI):*
61		*Generic Syntax,* http://www.ietf.org/rfc/rfc3986.txt, January 2005
62	**WS-Calendar**	T. Considine, M. Douglas, *OASIS WS-Calendar Public Review Draft 02*, April
63		2011, specification in progress, http://docs.oasis-open.org/ws-calendar/ws-
64		calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.pdf
65	**XML Schema**	H. Thompson, D Beech, M Maloney, N Mendelsohn, *XML Schema Part 1:*
66		*Structures Second Edition*, http://www.w3.org/TR/xmlschema-1/ October 2004
67		PV Biron, A Malhotra, *XML Schema Part 2: Datatypes Second Edition*,
68		http://www.w3.org/TR/xmlschema-2/ October 2004.

1.4 Non-Normative References

70	**EISA**	Energy Independence and Security Act (EISA 2007)
71		http://www.gpo.gov/fdsys/pkg/PLAW-110publ140/content-detail.html
72	**FIX**	Financial Information eXchange (FIX) Protocol,
73		http://www.fixprotocol.org/specifications/FIX.5.0SP2
74	**IEC TC57**	IEC TC 57 Power and Load Management, http://tc57.iec.ch/index-tc57.html
75	**NIEM**	NIEM Technical Architecture Committee (NTAC), *National Information Exchange*
76		*Model Naming and Design Rules v1.3*, October 2008,
77		http://www.niem.gov/pdf/NIEM-NDR-1-3.pdf
78	**TeMIX**	Transactional Energy Market Information Exchange [TeMIX] an approved White
79		Paper of the EMIX TC. Ed Cazalet et al. http://www.oasis-
80		open.org/committees/download.php/37954/TeMIX-20100523.pdf
81	**NAESB 03**	*Requirements Specification for Common Electricity Product and Pricing*
82		*Definition*, North American Energy Standards Board [NAESB], March, 2010
83		(Public Review Draft).
84		http://naesb.org/pdf4/weq_2010_ap6a_retail_2010_ap9a_rec.doc
85	**NIST Roadmap**	NIST Framework and Roadmap for Smart Grid Interoperability Standards,
86		Release 1.0, online.
87		http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final
88		.pdf
89	**PAP03**	Details of PAP03 may be found at http://collaborate.nist.gov/twiki-
90		sggrid/bin/view/SmartGrid/PAP03PriceProduct (link retrieved 06/23/2010)

91	**RFC5545**	B. Desruisseaux *Internet Calendaring and Scheduling Core Object Specification*
92		*(iCalendar)*, http://www.ietf.org/rfc/rfc5545.txt, IETF RFC 5545, September 2009.
93	**White Paper on WS-Calendar**	http://docs.oasis-open.org/ws-calendar/ws-
94		calendar/v1.0/CD01/WS-Calendar-Conceptual-Overview-CD01.pdf

1.5 Namespace

96 XML namespaces and prefixes used in this standard:

Prefix	Namespace
emix:	http://docs.oasis-open.org/ns/emix
power:	http://docs.oasis-open.org/ns/emix/power
resource:	http://docs.oasis-open.org/ns/emix/power/resource
xs	http://www.w3.org/2001/XMLSchema
gml:	http://www.opengis.net/gml/3.2
xcal:	urn:ietf:params:xml:ns:icalendar-2.0
clm5ISO42173A:	urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07

97 Namespace URIs resolve to a Resource Directory Description Language [RDDL 2.0] document
98 describing the namespace.

1.6 Naming Conventions

100 This specification generally follows the follows the National Information Exchange Model **[NIEM]** naming
101 and design rules for artifacts defining the specification, as follows:

102 For the names of elements and the names of attributes within XSD files, the names follow the lower
103 camelCase convention, with all names starting with a lower case letter. For example,

```
<element name="componentService" type="emix:ComponentServiceType"/>
```

105 For the names of types within XSD files, the names follow the Upper CamelCase convention with all
106 names starting with an upper case letter postfixed with "Type". For example,

```
<complexType name="ComponentServiceType">
```

1.7 Editing Conventions

109 For readability, element names in tables appear as separate words. The actual names are
110 lowerCamelCase, as specified above, and as they appear in the XML schemas.

111 The cardinality of each element can vary by transactive state. For clarity, cardinality for each element is
112 not indicated in the tables in the specification. Note: because of EMIX Inheritance (see section 16.1), a
113 :"missing" required element may be supplied through inheritance.

114 Information in the "Specification" column of the tables is normative. Information appearing in the note
115 column is explanatory and non-normative.

116 All sections explicitly noted as examples are informational and are not to be considered normative.

1.8 Semantics from WS-Calendar

Time semantics are critical to EMIX. An overview of EMIX semantics is in Appendix C for easy reference. Practitioners should read that specification or the [**White Paper on WS-Calendar].**

1.9 Market Semantics

Different energy markets have specific market terms and interaction patterns. This specification endorses none of them, but still needs to discuss the various stages of a market transaction. Without mandating the terminology used in any particular market, the EMIX specification uses the common market terms as defined in *Table 4-5: Transactive States Enumeration*.

You may want to turn ahead to have these definitions in mind as you read this document.

1.10 Security Approaches

EMIX is an information model, and thus security per se is out of scope for this specification. EMIX will normally be conveyed in messages as part of business processes. Each business process will have its own security needs, including different consequences for failure of security.

2 Overview

2.1 Introduction

Energy markets have been characterized by tariffs and embedded knowledge that make decision automation difficult. Different market segments use conflicting terms for similar attributes. Smart grids introduce rapidly changing products and product availability, with associated dynamic prices. A lack of a widely understood model conveying market information has been a barrier to development and deployment of technology to respond to changing market circumstances.

Price and product definition are *actionable information.* When presented with standard messages conveying price and product, automated systems can make decisions to optimize energy and economic results. In regulated electricity markets, price and products often are defined by complex tariffs, derived through not strictly economic processes. These tariffs convey the price and product information to make buying and selling decisions easier. The same information can be derived from market operations in non-tariffed markets. EMIX defines the information for use in messages that convey this actionable information.

An essential distinction between energy and other markets is that price is strongly influenced by time of delivery. Energy for sale at 2:00 AM, when energy use is low, is not the same product as energy for sale at the same location at 2:00 PM, during the working day. EMIX conveys time and interval by incorporating WS-Calendar into tenders, transactions, and performance calls.

Not all market information is available in real time. Present day markets, particularly wholesale markets, may have deferred charges (e.g. balancing charges) that cannot be determined at point of sale. Other markets may require additional purchases to allow the use of the energy purchased (e.g. same-time transmission rights or pipeline fees when accepting delivery on a forward contract). EMIX is useful for representing available price and product information.

2.2 Approach

The EMIX TC has prepared a white paper which provides a context for discussing the use of transactions in retail and wholesale energy markets. The Transactional Energy Market Information Exchange **(TeMIX)** white paper can be found in the non-normative references.

Transactive Energy Market Information Exchange (TeMIX) was developed as a specialization of work within the EMIX TC to address retail and wholesale transactions using approaches common in energy wholesale and financial transactions. The Energy Interoperation TC defines a TeMIX profile which is a subset of the EMIX information model and the Energy Interoperation TC services.

The TeMIX profile allows only specific tenders and transactions for block power on defined intervals of time. Tenders may be offered by any party to any other party, as market rules and regulations may allow. Any party can be a buyer, seller or both. Transactions may include call and put options. TeMIX Options perform a similar function to demand response contracts or ancillary service contracts where an operator has dispatch control over the exercise of the option. TeMIX products also include transmission and distribution (transport) products.

TeMIX tenders and transactions can support dynamic tariffs by retail providers to retail customers. TeMIX is designed for interval metering where delivery can be accurately measured. The simplified information model and services of the TeMIX profile also support increased automation of transactions using the computer and communications technology of the smart grid.

EMIX has adopted the much of the TeMIX terminology. EMIX supports current operating models of market operators, utilities, and demand response providers while at the same time supporting the TeMIX model and future transitions among the models.

Power is a commodity whose market value may be different based upon how it is produced or generated. After production, though, the commodity is commingled with production from other sources with which it is

176 fully fungible. Even so, some energy purchasers distinguish between sources of this product even as they
177 consume the commingled commodity.

178 Throughout this work, we refer to the intrinsic and extrinsic properties of an energy product. An intrinsic
179 property is one *"belonging to a thing by its very nature."* An extrinsic property is one *"not forming an
180 essential part of a thing or arising or originating from the outside."* In EMIX, the term intrinsic properties
181 refers to those that can be measured and / or verified at the point of delivery, i.e., electric power and
182 price. The term extrinsic properties refers to those that can only be known with prior knowledge, such as
183 the carbon cost, the energy source, or the sulfate load from generation.

184 EMIX artifacts communicate both intrinsic and extrinsic properties; extrinsic properties must be able to
185 clear in the market just as do intrinsic properties.

186 EMIX is not concerned with the processes whereby an actor provides the products and resources it
187 describes. EMIX is an information model that assumes conveyance within a service-based environment.
188 As defined in the OASIS Reference Model for Service Oriented Architecture 1.0 **[SOA-RM]**, service
189 requests access the capability of a remote system.

190   *The purpose of using a capability is to realize one or more real world effects. At its core, an*
191   *interaction is "an act" as opposed to "an object" and the result of an interaction is an effect (or a*
192   *set/series of effects). This effect may be the return of information or the change in the state of*
193   *entities (known or unknown) that are involved in the interaction.*

194   *We are careful to distinguish between public actions and private actions; private actions are*
195   *inherently unknowable by other parties. On the other hand, public actions result in changes to the*
196   *state that is shared between at least those involved in the current execution context and possibly*
197   *shared by others. Real world effects are, then, couched in terms of changes to this shared state*

198 A request for the delivery of a product is a request for specific real world effects. For EMIX, these effects
199 are expected to occur during a given period. Consider two sellers that offer the same product. One must
200 start planning an hour or more in advance. The second may be able to deliver the product in five minutes.
201 The service start time is the time when product delivery begins. Because this service start time and
202 service period are all that matters, different providers using quite different technologies can provide
203 equivalent product as specified in EMIX.

204 Time semantics are critical to EMIX. EMIX uses semantics from **[WS-Calendar]** to describe time,
205 duration, and schedule. WS-Calendar also defines an information model wherein services or products
206 that vary over time can be efficiently and unambiguously communicated using inheritance. Lastly, WS-
207 Calendar describes an approach wherein an incompletely specified sequence of information can be
208 completed using minimal re-definition and remote invocation. EMIX uses these semantic and
209 conformance rules throughout this specification.

2.3 Information Structure

211 As a conceptual aid, we discuss the information structure using the metaphor of an *envelope containing*
212 *warrants*. The intrinsic properties and the price are on the face of the envelope, easy to read by all. The
213 contents of the envelope are the supporting information and various warrants about the extrinsic qualities.

214 On the face of the envelope, EMIX lists the intrinsic qualities of the energy product. In the simplest model,
215 the intrinsic qualities are limited to the price and the information a meter can provide. In a market of
216 homogenous energy sources and commodity energy, only the intrinsic qualities are actionable. In postal
217 handling, information on the face is meant for high-speed automated processing. The simplest devices,
218 including the proverbial smart toaster, may understand only the intrinsic qualities. The phrase "prices to
219 devices" is used in energy policy discussions to describe a market model in which energy use decisions
220 are distributed to each device that uses energy. Under this model, decisions about whether to use energy
221 now or delay energy use until later are best made where the value is received for that energy use, the
222 end device. The smart toaster is shorthand for the smallest, least capable device that can receive such a
223 message. The Committee anticipates that the information on the face of the envelope will be sufficient for
224 many if not most energy decisions.

225 The envelope contents are the supporting documents that explain and support the price for the intrinsic
226 qualities on the face of the envelope. These extrinsic qualities are separable from the intrinsic transaction

227 and perhaps can be traded in secondary markets. The contents can include Warrants about the source
228 and the environmental attributes which provide information about the energy, but they are not the energy.
229 The extrinsic qualities enable traceability and auditing, increasing public trust in energy markets and on
230 energy differentiation. The simplest gateways and devices may ignore the warrants, that is, they can
231 forward or process messages without opening the envelope.

232 Extrinsic information conveyed within the envelope includes supporting information. For example, a
233 purchaser may opt to buy energy from a particular supplier with advertised rates. Transport loss may
234 reduce the quantity delivered. Markets may add congestion charges along the way. Such supporting
235 information can explain why the delivered cost, on the face of the envelope, is different than the purchase
236 cost.

2.4 EMIX Time and Schedules

238 Time is an important component of energy products. An energy product produced in one Interval of time
239 may or may not be able to be stored for delivery at a later Interval of time. Thus the same product in
240 different Intervals of time may have different prices. EMIX uses **[WS-Calendar]** to apply prices and
241 products to time Intervals.

242 WS-Calendar defines a mechanism to apply a schedule to a Sequence of time Intervals. WS-Calendar
243 further defines how to use a process analogous to inheritance to apply a single information artifact to
244 each Interval in the Sequence, allowing elements of that artifact to be over-ridden within any given
245 Interval. WS-Calendar also defines a schedule entry point, defining how specific performance can be
246 contracted and scheduled.

247 This document assumes that the reader has a clear understanding of WS-Calendar and its interfaces.
248 The non-normative white paper on the use of the WS-Calendar specification published by that committee
249 is a good place to start.

2.5 Tenders and Transactions for Power Products and Resource Capabilities

252 The focus of EMIX is on price and product communication in support of commercial transactions. The
253 messaging and interaction patterns for commercial transactions are out of scope for EMIX but worth a
254 brief discussion here to provide context.

255 EMIX is intended for commercial transactions in all types of markets including ISO/RTO markets,
256 exchange markets, regulated markets, regulated retail tariffs, open markets, and wholesale and retail
257 bilateral markets. The commercial practices that determine prices vary in these markets but all markets
258 can benefit from interoperable communication of price and product.

259 Transactions in most markets begin with Tenders (offers to buy or sell) by a Party to another Party. Once
260 an agreement among Parties is reached, the parties Agree to a Transaction (contract or award). The
261 parties to the Transaction then must perform on the Transaction by arranging for supply, transport,
262 consumption, settlement and payment. At every stage in this process, clear communication of the terms
263 (price, quantity, delivery schedule and other attributes) of the tender or transaction is essential. Section 4,
264 *"Overview of the Information Elements"* describes EMIX Base Type, the core of EMIX information models.

265 In many electricity markets Operators are offered electrical products based on specific resources, i.e.,
266 generators, load curtailment, and other energy resources. EMIX uses EMIX Resource Descriptions to
267 describe the responsiveness, capacity, and other aspects of these Resources. EMIX Resource Offers
268 combine an EMIX Resource Description with a multi-part offer. A Party can use EMIX Resource Offers to
269 tender to an Operator one or more EMIX Products. Similarly, an EMIX Load Curtailment Offer combines a
270 Load Curtailment Resource Description with a multi-part offer.

2.6 Transport

272 Product transport costs vary over time. Delivery costs come in two general forms. Congestion charges
273 apply to each unit of product that passes through a particular point in the distribution system. Congestion

274 charges increase the cost of the Product delivered in a particular Interval. Loss reduces the product
275 delivered as it passes from the purchase point to the delivery point. Loss may reduce the amount of
276 product received or a loss charge may be applied to purchase replacement energy for the energy loss.

277 If the Product is priced for Delivery to the consumer, transport charges may not apply. Product
278 descriptions for Transport charges are discussed in Section 14, *Power Transport Product* Descriptions.

279 2.7 Verification

280 Many products, particularly those transacted for Demand Response, are distinguished by particular
281 Verification Methods. In a pure transactive energy market, the meter would be the only Verification
282 mechanism. In today's markets, Verification can be more complex.

283 Verification is out of scope for this document. Verification is fully specified under NAESB Business
284 Practices for Verification. This specification does not describe verification.

3 Guide to the Schema Structures

The EMIX information exchange model defines common structures that can be used to define products whose value varies with the time of delivery. Because the future of smart energy markets its not known, there is an emphasis on extensibility and composition to allow EMIX to be suitable for markets known and unknown, and for easy evolution.

The EMIX 1.0 Specification consists of three schemas.

- The EMIX schema defines the framework and extensibility as well as agreement types common to many markets.

- The Power schema defines the specific information exchanges, based on the EMIX framework, needed for markets in power and energy.

- The Resource schema defines how load and generation, specific capabilities of devices and systems to affect power and energy markets, can be described irrespective of the underlying technologies.

Other markets, particularly other products for energy markets, share the characteristic that value is closely linked to time of delivery. Power and Resource provide examples of extension and conformance with the EMIX model. Specifications that wish to claim conformance with EMIX use should follow the same approaches. Information exchanges based on specifications that conform to the EMIX specification, can be used within any business process or specification that uses or exchanges EMIX payloads.

Table 3-1: EMIX Schemas

Schema	Definition
EMIX	The EMIX schema has target namespace http://docs.oasis-open.org/ns/emix and consists of three files—emix.xsd, emix-requirements.xsd, and emix-warrants.xsd
Power	The Power schema as target namespace http://docs.oasis-open.org/ns/emix/power consists of three files—power.xsd, power-product.xsd, and power-quality.xsd.
Resource	The Resource schema has target namespace http://docs.oasis-open.org/ns/emix/power and consists of one file—resource.xsd

3.1 Core Extension Elements

The core extension elements are the Product Description Type and the EMIX Base Type. These types include the abstract types Item (Item Base), and the Interface (EMIX Interface). Almost all of EMIX using these four abstract types.

The abstract Product Description Type is the basis for all static descriptions of EMIX products. Product Descriptions are static in that they refer to a particular instance in time. Most of the elements in the Power and Resource schemas are creating Product Descriptions for Power Markets.

The abstract EMIX Base Type defines a Product Description Type to a Schedule. That Schedule may be as simple as a single 5 minute interval on a particular day, or as complex and repeating you can find in your own personal calendar. Any type derived from the EMIX Base Type can hold any Product Description. Information elements derived from the EMIX Base include Products, Options, TEMIX, and Delivery (Metered Information).

The Item Base is the basis for the lowest level description of each Product and its aspects. The term Item is in common business use for that thing on a line of a purchase order, or of a receipt, or on a bill of

319 lading. Item Base derived types have at least a name, a unit of measure, and a scale factor. The Power
320 schema defines 3 power types and three energy types derived from the Item Base Type.

321 All product descriptions include the EMIX Interface. The EMIX Interface is where something transfers
322 ownership for the market. In Power, it can be a node or meter, and aggregation of nodes or meters, a pair
323 of nodes, or a geographic area. Other specifications can derive from the base type to support their own
324 needs. Any type derived from the Interface can be used in any of the EMIX Base derived types.

3.2 Extensibility

326 EMIX supports a modular model in which extensions to EMIX can easily be propagated into standards
327 that communicate EMIX. There are multiple EMIX envelopes to participate in different roles; each
328 includes a set of EMIX Base Types that describe what is tendered or transacted.

329 New efforts could specify novel Product Descriptions by extending the EMIX Product Description Type.
330 These new Product Descriptions would define new conformant EMIX Products merely by application the
331 EMIX Base Type. Such conforming Products could then be transported on any EMIX Envelope. Any
332 Specification that communicates EMIX Products can exchange market information about these new
333 Product Descriptions. A new committee can extend EMIX into new products without re-considering any
334 aspects of the EMIX specification itself.

335 A similar logic applies to the warrants, which outside the scope of this specification. If the warrant
336 information varies over time, the warrant information can be applied to a WS-Calendar Sequence just as
337 if it were a Product Description.

338 Extensibility mechanisms supported in EMIX are discussed in Appendix B.

3.3 Power and Resource Schemas

340 The Power and Resource schemas are, in effect, the first extensions to the EMIX Schema. This
341 specification includes two schemas that extend EMIX. The Power schema extends the EMIX schema to
342 define products for Power markets. The Resource schema extends the EMIX schema to define the
343 capabilities of systems in ways that allow market participants to make buying decisions.

4 Overview of the Information Elements

EMIX describes the market communications (EMIX Base type) of tenders and transactions for products whose market value varies with time of delivery. An energy product typically is delivered over time at a specific location. Five kW at 2:00 AM does not provide the same energy services as five kW at 2:00 PM. EMIX describes the terms of tenders and transactions for which time and location are essential characteristics. For example, the price and quantity (rate of delivery) of energy in each time Interval of a Sequence of Intervals may vary for energy transactions made in a Sequence of Intervals.

EMIX Base derived types are created by applying Product Descriptions to WS-Calendar Sequences. WS-Calendar Sequences embody the same calendaring standards used by most business and personal calendaring systems. This enables greater interoperation between grid systems and business and personal systems. An EMIX Product Description describes the elements of an energy product at a location for a one Interval or a Sequence of Intervals. An EMIX Product Description for a constant rate of delivery power product over a single Interval comprises a (1) start time, (2) duration, (3) rate of delivery, (4) price and (5) location. If the rate of delivery (kW) and price ($/kWh) have been exchanged in advance, the information exchanged to deliver the product is simply "start (reference **[URI]** to product) at 3:00 AM for 0.75 hours."
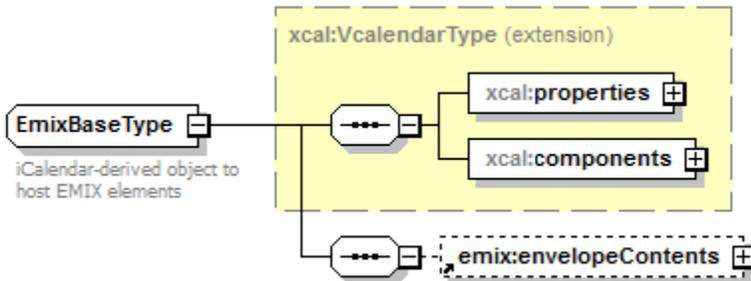
Figure 4-1: EMIX Base Type

A Product Description included in each Interval in a Sequence could describe the same elements again and again. Only a few elements, perhaps only price, or quantity, may change per Interval. EMIX uses the WS-Calendar Sequence to specify product elements once, and then specifies which elements may vary by the time Intervals of a Sequence.

For example, a resource representing a responsive load may state that 15 minutes lead time is required between notification and load reduction. This characteristic may hold true whether the response requested is for a run-time of 10 minutes or for one of 10 hours. EMIX specifies invariant characteristics as part of a product description or resource, while offering the variable run-time to the market.

EMIX Base types using EMIX Product Descriptions applied to WS-Calendar Sequence provide a flexible information model for describing any energy tender or transaction. New or specialized energy products can offered and transacted without changing the EMIX standard.

EMIX Base types minimize the size of information exchanged by efficiently describing how information elements of a tender or a transaction may or may not vary over time. This reduces communication overhead.

4.1 The Intrinsic Elements: EMIX Products

The following table Table 4-1: Elements of the EMIX Product specifies the Intrinsic Elements in the EMIX information model. Intrinsic elements make up the face of the envelope. EMIX-based transactions are based on the exchange of these envelopes. There are four types of envelopes defined in EMIX. These are Product, Option, TeMIX, and Delivery, each envelope with its own requirements.

381 Central to each is the Base Product Description Type. The Base Product Description Type is the abstract
382 class from which all Product Descriptions are derived. A Product is a description of the product or service
383 applied to a delivery schedule. Product Descriptions as concrete classes, make up most of the sections
384 after this one. However, as no envelope is complete without a Product Description, we define them here.

4.1.1 Intrinsic Elements of the EMIX Product Type
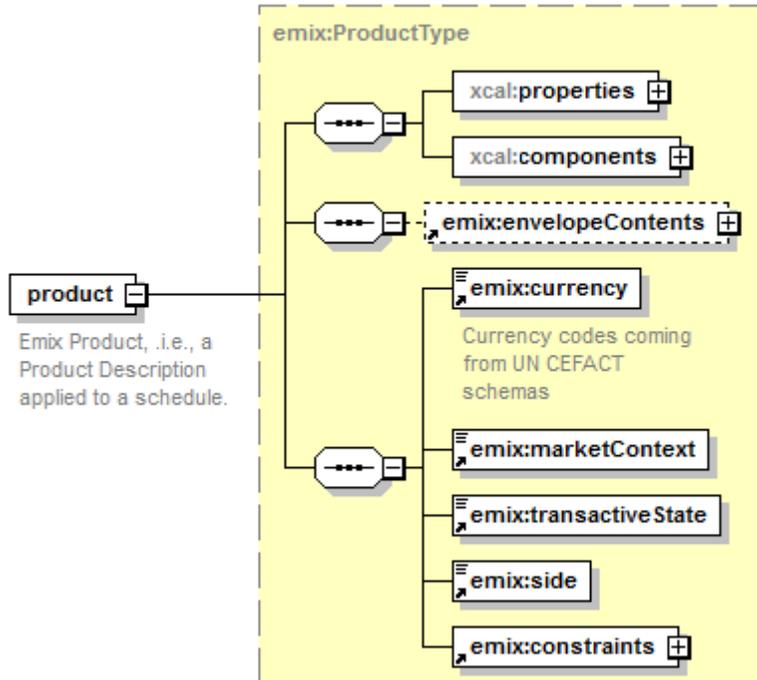
386

387 *Figure 4-2: EMIX Product Type*

388 The EMIX Product is the commonest of the envelopes. It is used for simple tenders, and agreements. It
389 describes specific product delivery.

390 *Table 4-1: Elements of the EMIX Product*

Product Element	Definition
UID	Identifier of this artifact. In many (if not most) markets the UID is required to be globally unique.
Transactive State	Used to aid parsing and conformance, e.g., to distinguish between different purposes for EMIX communications See *Table 4-5: Transactive States Enumeration*
EMIX Product	EMIX Products are created by applying a Product Description to a Schedule using the Base Product abstract class.
Market Context	An identification of the market in which the Product is offered. This may include standard financial and energy exchanges, markets managed by system operators, markets managed by or for aggregators and distributors, or an identification of the microgrid in which the Product is priced
Party Type	Identifies whether information originator is Buyer or Seller
Currency	A code that indicates the currency used, as specified in **[CEFACT]**

Product Element	Definition
Constraints	A collection of business and performance rules that constrain the option agreement. See Constraints at Section 16
Envelope Contents	As defined in section 4.3 *Inside the Envelope – the Extrinsic Items*

4.1.2 Intrinsic Elements of the EMIX Option

391

392 The EMIX Option is an elaboration of the EMIX Product described above. An option is a financial
393 instrument that gives the buyer the right, but not the obligation, to buy or sell a product at a set price
394 during given time windows. Many typical energy agreements, including demand response and reserves,
395 include elements that would give them the name Option in any other market.
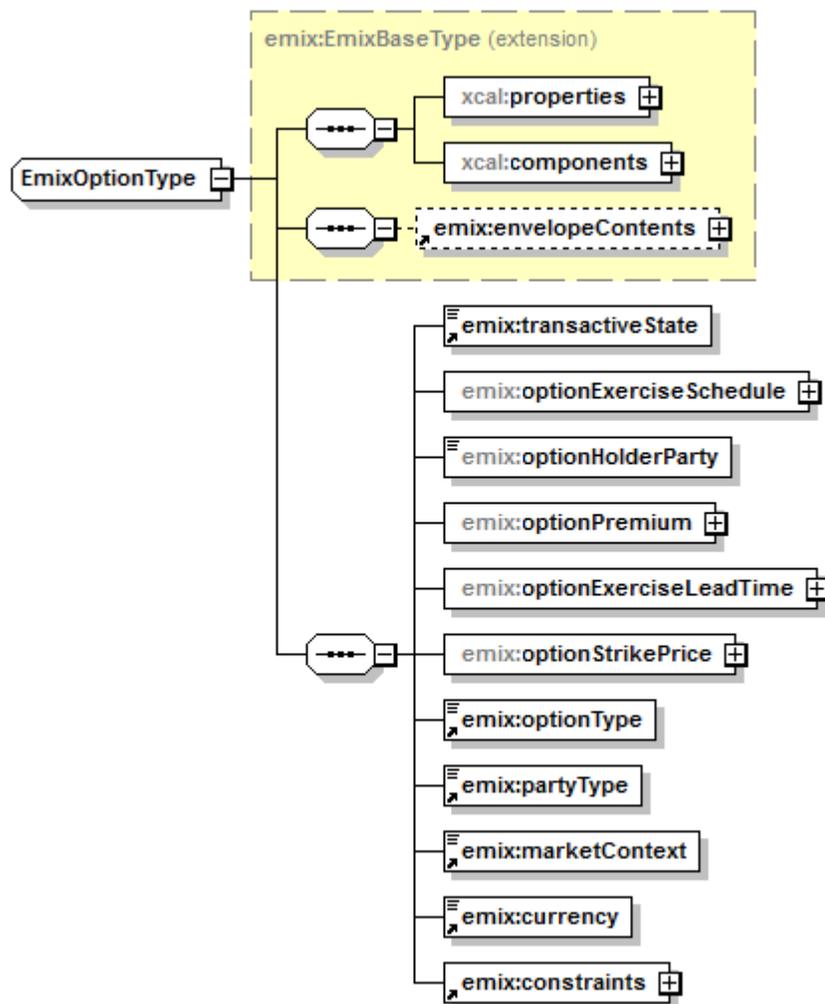
396

397 *Figure 4-3: EMIX Option Type*

398 The EMIX option also specifies specific availability and performance. The "face of the envelope" displays
399 additional information to support these requirements.

400 *Table 4-2: Option Elements – another "Face of the Envelope"*

Option Element	Specification

Option Element	Specification
UID	Identifier of this artifact. In many (if not most) markets the UID is required to be globally unique.
Transactive State	Used to aid parsing and conformance, e.g., to distinguish between different purposes for EMIX communications
EMIX Product	EMIX Products are created by applying a Product Description to a Schedule using the Base Product abstract class.
Market Context	An identification of the market in which the Product is offered. This may include standard financial and energy exchanges, markets managed by system operators, markets managed by or for aggregators and distributors, or an identification of the microgrid in which the Product is priced
Currency	A code that indicates the currency used, as specified in **[CEFACT]**
Envelope Contents	As defined in section 4.3 *Inside the Envelope – the Extrinsic Items*
Option Exercise Schedule	Uses the Availability Schedule Constraint to specify the period or periods in which the option is available for exercise. For example, a reserve power option could specify a schedule of afternoons in July excluding the 4th
Option Holder Side	The side which enjoys the benefit of choosing whether or not to exercise the terms specified in the option. Sometimes referred to as the Promisee
Option Premium	The price paid to the Promisor for the rights involved
Option Strike Price	The price at which an option holder (Promisee) has the right to require the option writer (Promisor) to deliver
Exercise Lead Time	The minimum notification time required by the Promisor to to be able to perform. Uses the Minimum Notification Duration Constraint. The Promisor is not responsible for performance in less than the Exercise Lead Time.
Strike Price	The price the Promisee will pay the Promisor delivery of Product under the option. May be fixed or relative to a specified market.
Side	Identifies whether information originator is on the Buy or Sell side
Option Type	An enumerated list of Option types
Constraints	A collection of business and performance rules that constrain the option agreement. See Constraints at Section 16

401   4.1.3 Intrinsic Elements of the TeMIX
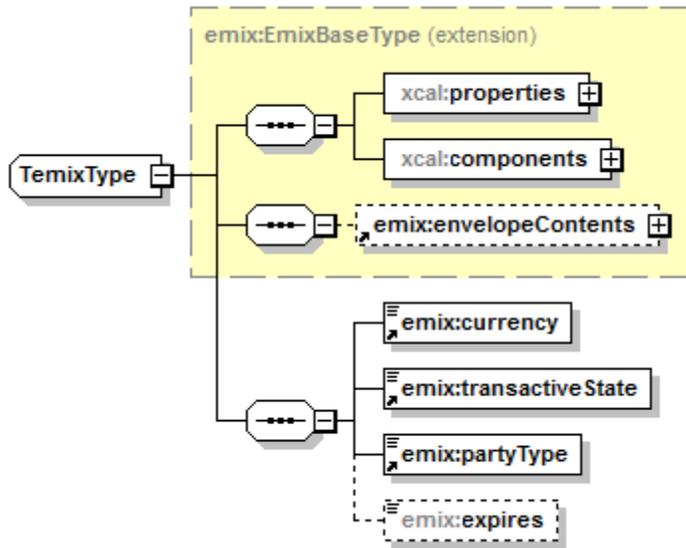
402

403   *Figure 4-4: The TEMIX Product*

404   The TEMIX (Transactional Energy Market Information Exchange) is a model for balancing energy markets
405   with pure economic trading. As such, it is the simplest of the EMIX Envelopes.

406   *Table 4-3: Elements of the TeMIX*

TEMIX Element	Specification
UID	Identifier of this artifact. In many (if not most) markets the UID is required to be globally unique.
Transactive State	Used to aid parsing and conformance, e.g., to distinguish between different purposes for EMIX communications See *Table 4-5: Transactive States Enumeration*
EMIX Product	EMIX Products are created by applying a Product Description to a Schedule using the Base Product abstract class.
Currency	A code that indicates the currency used, as specified in **[CEFACT]**
Constraints	A collection of business and performance rules that constrain the option transaction. See Constraints at Section 16. The permitted list of constraints for TeMIX is constrained to those discussed in section 11.
Expiration Date	For Tenders only, the date and time when this Tender expires.
Envelope Contents	As defined in section 4.3 *Inside the Envelope – the Extrinsic Items*

407   See Section 11 for a discussion putting TeMIX products in context.

4.1.4 Intrinsic Elements of Delivery

409
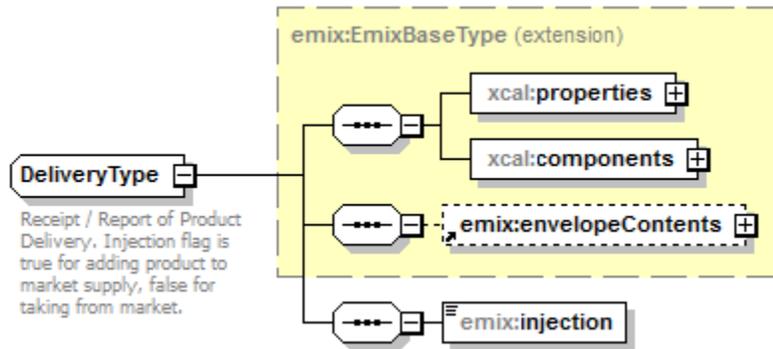
Figure 4-5: Delivery

In any market, order must be matched to delivery. EMIX Delivery reports the historical delivery of product over time.

Table 4-4: Elements of the EMIX Delivery

Delivery Element	Specification
UID	Identifier of this artifact. In many (if not most) markets the UID is required to be globally unique.
EMIX Product	EMIX Products are created by applying a Product Description to a Schedule using the Base Product abstract class.
Injection	True means positive Delivery is injection into the grid. False means positive Delivery is extraction from Grid
Envelope Contents	As defined in section 4.3 *Inside the Envelope – the Extrinsic Items*

4.1.5 Other Envelopes and Information Elements

EMIX anticipates that further elements will be defined, and an EMIX envelope containing elements not defined herein is fully compliant.

4.2 Transactive States

As parties use EMIX to come to an agreement and transact energy, the information required changes. An initial offer may not have a price. An agreement may not yet have a performance date. It is necessary for both parties in any communication to understand the Transactive State, i.e. what level of agreement defines the current communication.

Table 4-5: Transactive States Enumeration

Transactive State	Description
Indication of Interest	Indication of Interest is a non-binding offer or request for offer for a transaction that may or may not indicate price and quantity and other terms.
Tender	A Tender is a binding offer or bid for a Transaction by a party that when accepted by a counter party will result in a binding Transaction. ISOs use the term Bids to describe offers and bids into their markets.

Transactive State	Description
Transaction	A Transaction between two parties is a binding agreement
Exercise	Exercise applies to two-part Transactions such as Ancillary Services Dispatch by ISOs, Call and Put Options and DR event dispatch that have an initial agreement that includes a second step to that results in another Transaction.
Transport Commitment	Transport Commitment is what the ISOs call "Transmission Scheduling" which is a Transport product and not an energy product transaction. Since the distribution grid may require such transactions or schedules in the future we use the term "Transport"
Delivery	Delivery, which includes both Production and Usage, is the act of actually generating and consuming power that is measured by meters and reported for settlement to the parties. Delivery also names the enumeration of the Delivery.
Publication	Publication is the act of general announcement or posting of appropriate prices and other information concerning products. Publications are not Tenders or Indications of Interest.

4.3 Inside the Envelope – the Extrinsic Items

While energy markets deliver a blended commodity, the customer may value the product differently based upon indistinguishable characteristics of the commodity. Often this distinction is based upon the origin of the product. The product may come with attached credits that may have re-sale value. The buyer may contract for, and the supplier may need to report specific quality of product delivery. In other circumstances, it may be necessary to deliver supporting detail to explain the prices delivered.
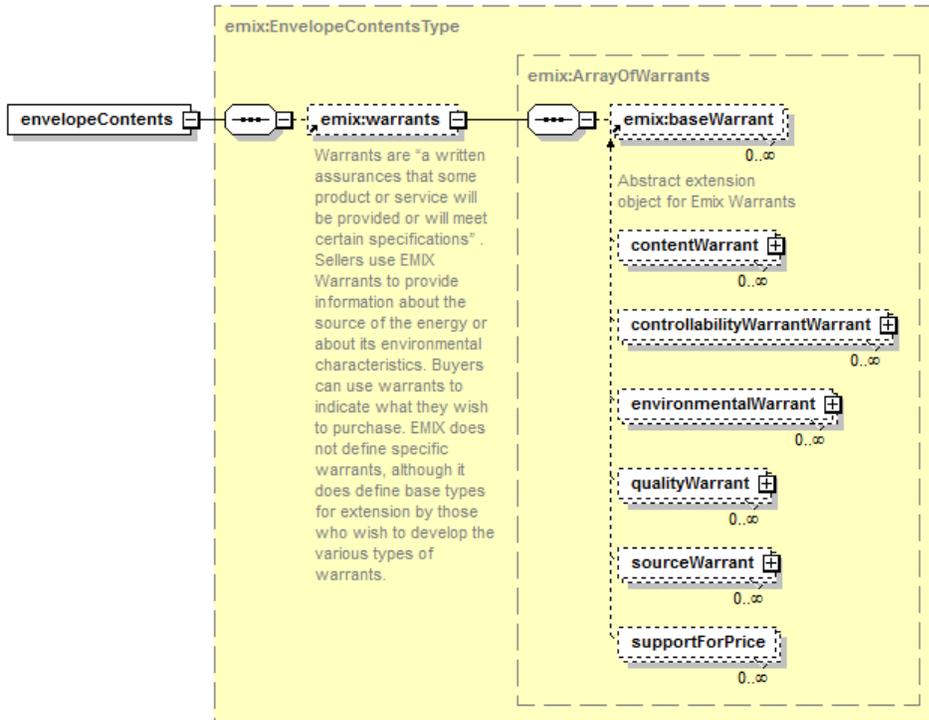
Figure 4-6: Envelope Contents

431 The definition of a warrant is "a written assurance that some product or service will be provided or will
432 meet certain specifications". Sellers use EMIX Warrants to provide information about the source of the
433 energy or about its environmental characteristics. Buyers can use warrants to indicate what they wish to
434 purchase. It seems a fundamental market rule that a middleman cannot sell more wind power than he has
435 bought. Such rules are beyond the scope of EMIX, but EMIX-based information exchanges support such
436 market rules.

437 EMIX Warrants are assertions about the EMIX Product.

4.4 Summary of the EMIX Base Derivations

438
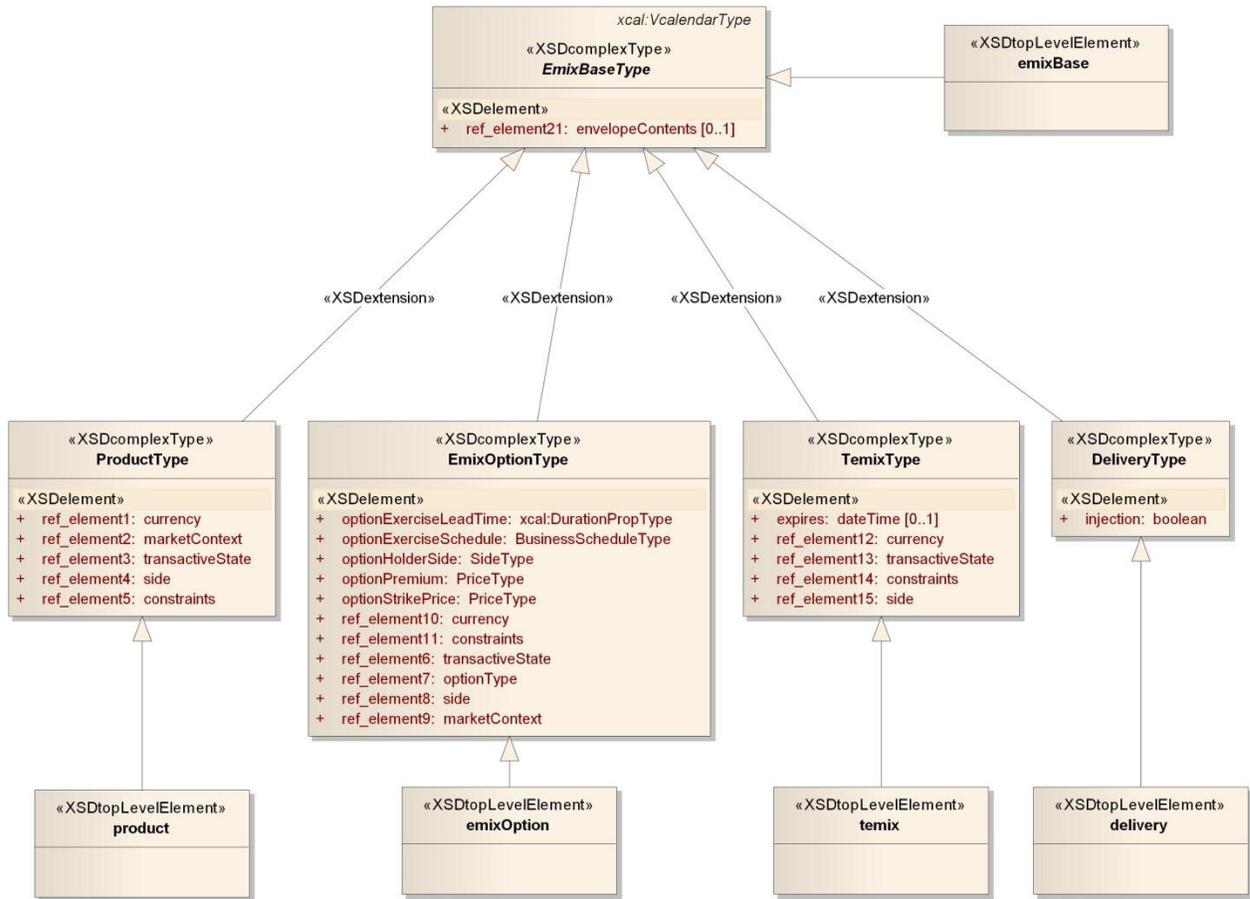
439

440 *Figure 4-7: UML of EMIX Base and its Extensions*

5 Constraints and Market Requirements

As noted on each of the elements above, EMIX Products can be subject to a number of Constraints and Market Requirements. These Constraints and Requirements can apply at each transactive state. Both Constraints and Requirements are extensible, so additional schemas, specifications, and standards can extend the lists while remaining in conformance.

Neither the EMIX Constraints nor Requirements are tied to any particular kind of Product or Resource (See section 10 for a discussion of Resources).

5.1 EMIX Constraints

Constraints are extrinsic to the product delivery but affect how a partner may request performance of a service. Performance constraints may originate in the basic mechanical needs of the Resource or to the business needs of the source. These constraints can affect the market value of the resource or the repeated invocation of a resource. It is possible for a given underlying resource to be offered to the market with different constraints and therefor different values.

Table 5-1: Constraints

Constraint	Definition
Minimum Response Duration	The shortest Duration for which the resource will accept a request to maintain a response before returning to pre-request levels.
Maximum Response Duration	The longest Duration for which the resource will accept a request.
Minimum Recovery Duration	The minimum Duration that the Resource requires after the end of a response the resource has is ready to respond to a new request.
Minimum Duration Between Invocations	The minimum Duration that the Resource requires after receiving a request before the resource has is ready to respond to a new request.
Minimum Notification Duration	The minimum Duration that the Resource requires for Notification before initiating a response to a request.
Maximum Notification Duration	The maximum Duration in advance of a response that the Resource will accept a Notification.
Response Time	Duration required from receipt of a request to supplying the requested level of response by the resource
Maximum Invocations Per Duration	Maximum number of invocations of service during a given duration
Maximum Consecutive Durations	Maximum consecutive durations in which service can be invoked, e.g., it will not accept requests on more than 3 consecutive days.
Minimum Starts Per Duration	The fewest Requests that the resource will accept during any duration. This constraint is typically used in market rather than in resource descriptions
Maximum Run Duration	The Maximum duration for which a resource will accept a request
Minimum Run Duration	The Minimum duration for which a resource will accept a request

Constraint	Definition
Availability Schedule	A schedule of time for which a resource will accept requests. The schedule may include multiple availability windows, i.e., an availability in May, can include weekday mornings and Thursday afternoons. The scheduled duration must be entirely within a single instance of an availability window.
Notification Schedule	A schedule of time during which a resource will accept requests. The schedule may include multiple availability windows which may be tied to business process, i.e., must receive notifications between 8:00 and 9:30 on business days. The notification must be received within a notification window.
Unavailability Schedule	A schedule of time during which a resource will accept requests. The schedule may include multiple availability windows, i.e., an availability in May, can include weekday mornings and Thursday afternoons. The scheduled duration must be entirely within a single instance of an availability window.

5.2 Market Requirements

456 Market Requirements are the market portion of Constraints, i.e., they are used to state the offeror's
457 expectations about a tender. It is possible for a given underlying resource to be offered to the market with
458 different Requirements and therefore different values.

459 *Table 5-2: Market Requirements for EMIX Products*

Market Requirement	Definition
Minimum Economic Requirement	Minimum net remuneration this resource requires from a total response
Required Startup Cost	Minimum remuneration required from start-up of this service.
Minimum Resource Cost	Resource requires this amount per period, i.e., a minimum requirement for $100 / hour at whatever rate

460 There is a wide variety of warrant types, issuing authorities, and characteristics described by warrants.
461 For bilateral agreements, there may be self-issued warrants. In larger markets, there may be a
462 requirement that Warrants be traceable through multiple levels of transactions.

463 Warrants are discussed in Section 15.

6 Interfaces and Items – Components for Constructing Product Descriptions

EMIX Product Descriptions applied to business schedules define EMIX Products. The EMIX Products were defined in section 4.1, including Products, Options, TEMIX, and Delivery. All product descriptions include an EMIX Interface and one or more elements derived from the EMIX Item Base.

6.1 EMIX Interfaces

Every market transaction occurs at an interface, where beneficial rights to or use of a product are transferred between buyer and seller. This is often the point at which the flow of product is measured although it may not be.

In power Markets, described in the sections below, it can be a node or meter, and aggregation of nodes or meters, a pair of nodes, or a geographic area. Only the geographic area is defined within the EMIX schema.

Other specifications can derive from the base type to support their own needs. Any type derived from the Interface can be used in any of the EMIX Base derived Product Descriptions.

6.2 Item Base

In common business usage, the item is that thing on each line of the Purchase Order, n each line of the Invoice, and on each line of the Shipping Document. Common aspects of the item is the name and the unit of measure. For general use, EMIX also defines the

EMIX references the International System of Units (SI) to specify a set of unit prefixes known as SI prefixes or metric prefixes. An SI prefix is a name that precedes a basic unit of measure to indicate a decadic multiple or fraction of the unit. The SI prefixes are standardized by the International Bureau of Weights and Measures (IBWM)in resolutions dating from 1960 to 1991. EMIX enumerates the Si prefixes in the SiScale enumeration. EMIX requires that conforming specifications use the SiScale to indicate the size of the unit of measure.

The Item Base is used not only to quantify the Item, but potential attributes of the Item as well.

Items do not include quantity or precise. The same Item definition may be used in every transactive state, and prices and quantities are not known for all.

6.2.1 Example of use of Item Base

The Item Base is used in many derived types. This illustration shows the POWER Energy Item Type, from which Real, Apparent, and Reactive Energy are derived.
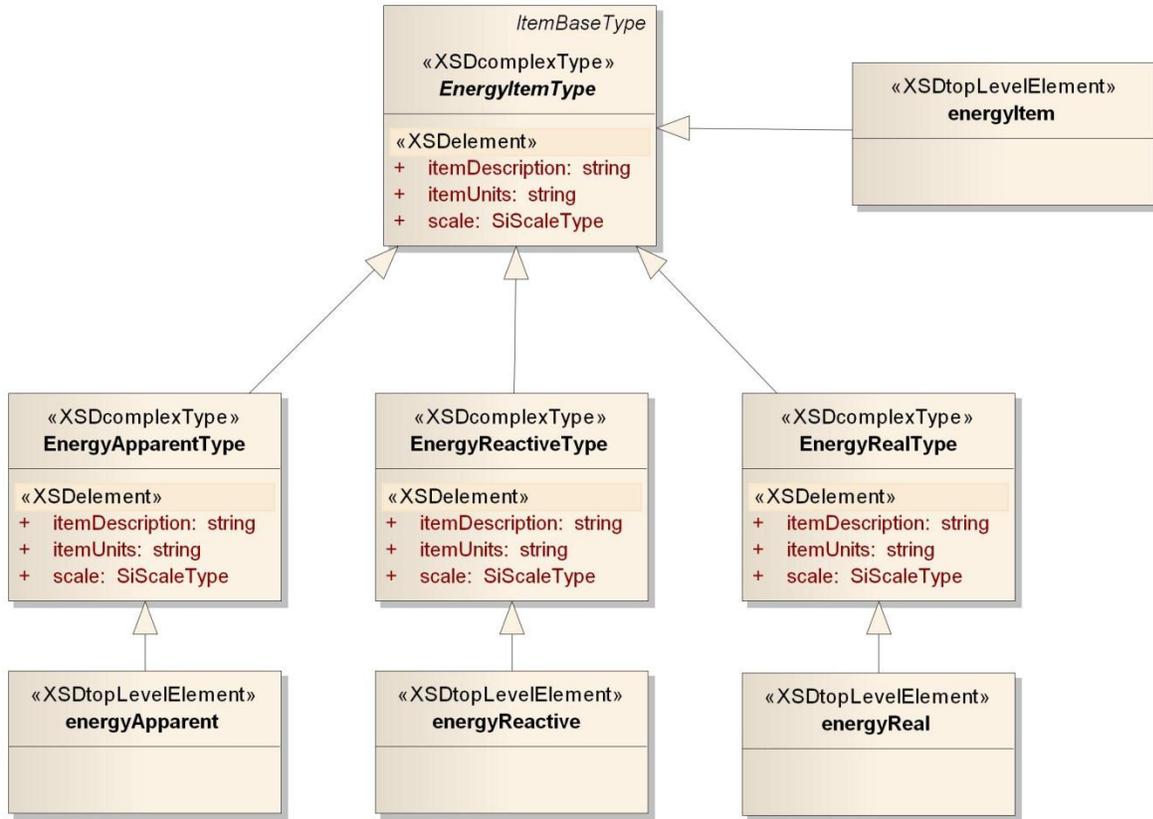
494

495   *Figure 6-1: UML showing use of Item Base in Energy Types*

7 The Schedule in the EMIX Product: Gluons and Intervals.

The EMIX Base Product is an abstract class that defines how all Product Descriptions are assembled with a schedule to be brought to market. The Base Product also defines an inheritance model whereby a fixed description of a product is refined with additional information as it becomes actionable.

While a product can be fully defined within an Interval, energy markets often consist of many consecutive intervals throughout the day. The intervals can be as short as minutes, or even seconds. A day's worth of intervals, each described separately, would consist of much duplicate information. For this reason, it is desirable to define product information in the Gluon, and place only those bits of information that change over time in each interval. Sometimes, the information in a particular interval takes precedence over the inherited information. The rules of inheritance are discussed below in EMIX Conformance with WS-Calendar,

7.1 The EMIX Gluon

The Base Product incorporates structures and inheritance patterns from **[WS-Calendar]** that are applied to and through the schedule. Table 7-1 and Table 4-1describe the key elements of the semantics of the Base Product. **[WS-Calendar]** defines the Gluon as a way to convey information relating to an entire Schedule. Those unfamiliar with WS-Calendar may wish to refer to Appendix C for an overview,

Table 7-1: EMIX Base Product – the Gluon

Gluon Element	Definition
Product Description	An EMIX ProductDescription describes the energy or services, the location and the price and quantity variables that can be set as a default in the Gluon and inherited by the Intervals in the Sequence. With the possibility of lineage of multiple gluons, product description in the interval, this is nor required.
Gluon Duration	A Duration set in a Gluon can be inherited by a Sequence of Intervals, subject to the inheritance rules. Not present in all Gluons.
Gluon Quantity	A Quantity set in a Gluon can be inherited by a Sequence of Intervals, subject to the inheritance rules. Not present in all Gluons.
Gluon Unit Price	A Price set in a Gluon can be inherited by a Sequence of Intervals, subject to the inheritance rules. Not present in all Gluons.
Sequence	A sequence is a set of intervals and the Gluons associated with them. A Gluon influences a Sequence through Inheritance.
Starting Date-Time	A Price set in a Gluon can be inherited by the Designated Interval in the Sequence to define the schedule for all Intervals in the Sequence. Not present in all Gluons
Designated Interval	The Interval in a sequence which has a direct relation with a Gluon (or chain of Gluons).
Availability	When present in a tender, defines when the product is available for delivery.

7.2 The EMIX Sequence and Intervals

[WS-Calendar] defines a Sequence is a temporally related set of Intervals. An interval is a period when something is done or delivered. Because of the temporal relation, Scheduling one Interval in the Sequence schedules them all. For this reason, EMIX Intervals are normally brought to market through one or more Gluons, each able to schedule its Sequence.

Table 7-2: EMIX Base Product - the Interval

Interval Element	Definition
Product	Elements of the Product Description that can be inherited without change from the Gluon need not be expressed in the Interval.
Duration	Can be inherited from the Gluon Lineage. Local expression supersedes inheritance.
Quantity	Can be inherited from the Gluon Lineage. Local expression supersedes inheritance.
Unit Price	Can be inherited from the Gluon Lineage. Local expression supersedes inheritance.
Starting Date-Time	Within a Sequence, is computed from the Starting Date Time of a single member of the Sequence. The Designated Interval can inherit this from the Gluon Lineage. Local expression supersedes inheritance.
Relation	Link from one Interval to other that specifies the relationship in time between Intervals in a Sequence.

7.3 EMIX Product Model

The illustration below provides a model demonstrating a sequence of three intervals, and the successive application of Gluons to bring them to market.
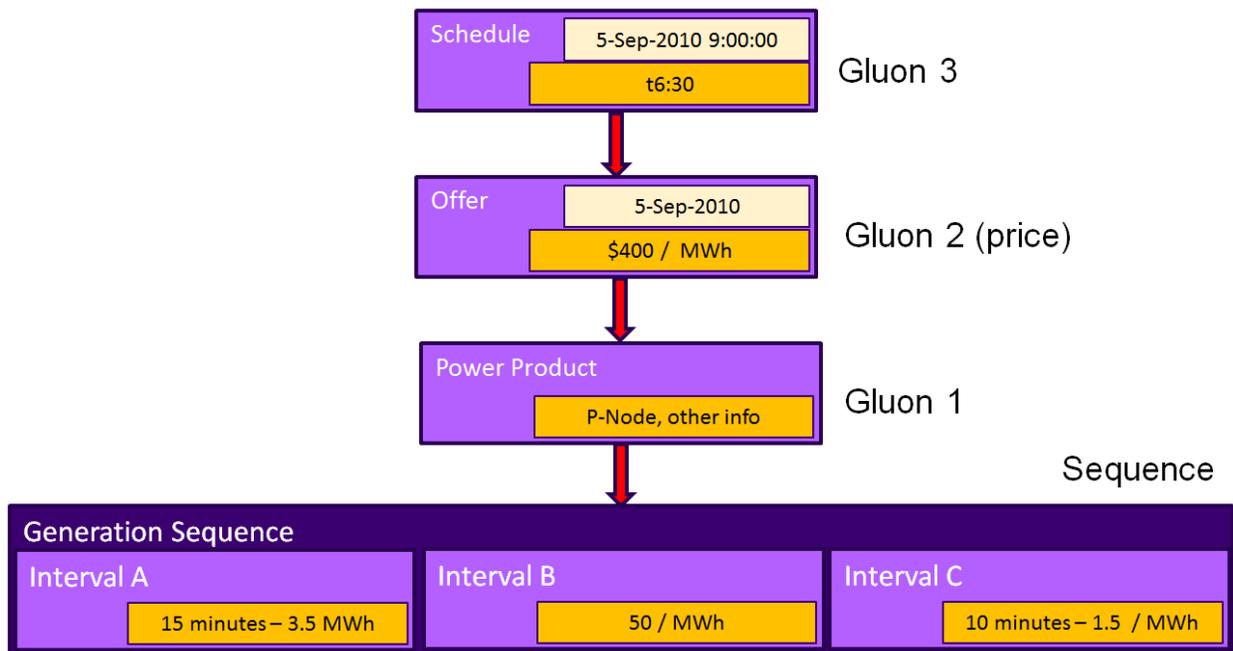
Figure 7-1: EMIX Model

525   1. *Power source defines product to market (Sequence and Gluon 1).*

526   2. *Product is offered to market on a particular day ([1] and Gluon 2) (Date but not time, required*
527   *price specified)*

528   3. *Transaction specifies start time (9:00) and duration (6:30) (Gluon 3), inherited by Sequence*
529   *through Gluons 2 and 1. Interval B (linked to Gluon 1) is the Interval that starts at 9:00.*

8 EMIX Power Product Descriptions

Electrical Energy and must be described precisely as it comes to market. Different products can provide total power, real power, or reactive power. Products delivering the same Power at a different voltage, or in DC rather than AC, may be valued differently. For the convenience of the readers, terms associated with electrical power and energy, and the relationships between them, are reviewed in Appendix E.

EMIX Provides a general model for exchanging product and market information about products whose value is tied closely to the time of delivery. EMIX Power defines specific EMIX Products for Power delivery. EMIX Resources define capabilities that could be brought market and the performance characteristics those resources will have, and thus enable a buyer to determine which resources to seek agreements with.

EMIX Products consist of Product Descriptions applied to the EMIX Base Product. The sections ahead discuss three classes of Product Description:

1) Power Product Descriptions

2) Resource Offer Descriptions

3) Transport Product Descriptions

EMIX Electrical Power Products are defined using standard attribute definitions from the Power and Load Management Common Information Model (CIM). The canonical definitions are in the IEC TC57 CIM.

8.1 Power Product Descriptions

Power can be bought under terms that

a) Specify the rate of delivery over a duration of an interval. Duration times power = energy

b) Specify the amount of energy over an interval with no restrictions on the rate of delivery at any instant with in the interval.

c) Made available as Full Requirements Power (the same as b) except that the amount of energy transacted is measured after delivery.

Power Products are the subject of tenders and transactions, i.e., they are what is actually bought and sold. Depending upon the market, Power can be bought under terms that specify the energy and its rate of delivery (power), or made available for use up to the maximum amount deliverable by the in-place infrastructure (also known as "Full-requirements Power") Power Products in Section 9, *Power Product Descriptions.*

8.2 Resource Offer Descriptions

Resources include generators that can produce power and other services, storage devices that can consume, store and then produce power, and load that produce a power through load curtailment.

A Resource Offer describes both the characteristics of the resource and the prices and quantities of products and services offered as described in Section 10, *Energy Resources*

8.3 Transport Product Descriptions

Product transport incurs specific costs that vary over time. Transport costs include congestion charges that apply to each unit of Product that passes through a particular point in the distribution system, and loss, .which reduces the Product delivered. If the Product is priced for Delivery to the consumer, transport charges may not apply. Product descriptions for Transport charges are discussed in Section 14, *Power Transport Product* Descriptions *Descriptions.*

9 Power Product Descriptions

571 The information model in this section is described in POWER-PRODUCTS.XSD

572 All Power Products are based on core abstract class, the Power Product Description. The Power
573 Products also share core semantic elements, used throughout the Descriptions and their associated
574 charges. Not all elements are in all classes; these are the recurring elements.

575 *Table 9-1: Semantic Elements common to Multiple Power Products*

Name	Definition
EMIX Interface	An EMIX Interface is any of a number of metering points (as defined below), an aggregate point, or a geographic area at which a product exchanges ownership
Attributes	Essential characteristics (Voltage, Hz, AC or DC) of delivered electricity.
Voltage	One of three elements hereafter referred to as the Power Attributes.
Hertz	One of three elements hereafter referred to as the Power Attributes. Always 0 for DC
AC	One of three elements hereafter referred to as the Power Attributes.
Power Units	Enumeration of Power Units, e.g., total power (VA), real power (W), and reactive power (VAR)
Energy Units	Enumeration of Energy Units, e.g., including real energy (Wh), reactive energy, (VARh), and apparent energy (VAh)
Voltage Units	Enumeration of Voltage Units, e.g., MV
VAR Units	Enumeration of volt amperes reactive (var) units, e.g., Kvar
Meter Asset	Identifier for an actual or virtual meter
Node	Grid Location identifier
Price	A fixed price.
Price Multiplier	A multipler relative to a market. It consists of a multiplier, which could be more or less than 1.0 and of a reference to a market context. PriceMultiplier can also be used to set a price now to match market at a forward period in time
Price Relative	A price to add or subtract from the pre-existing market price. It consists of a price, which could be positive or negative, and of a reference to a market context.

576

577 ## 9.1 Base Power Contract
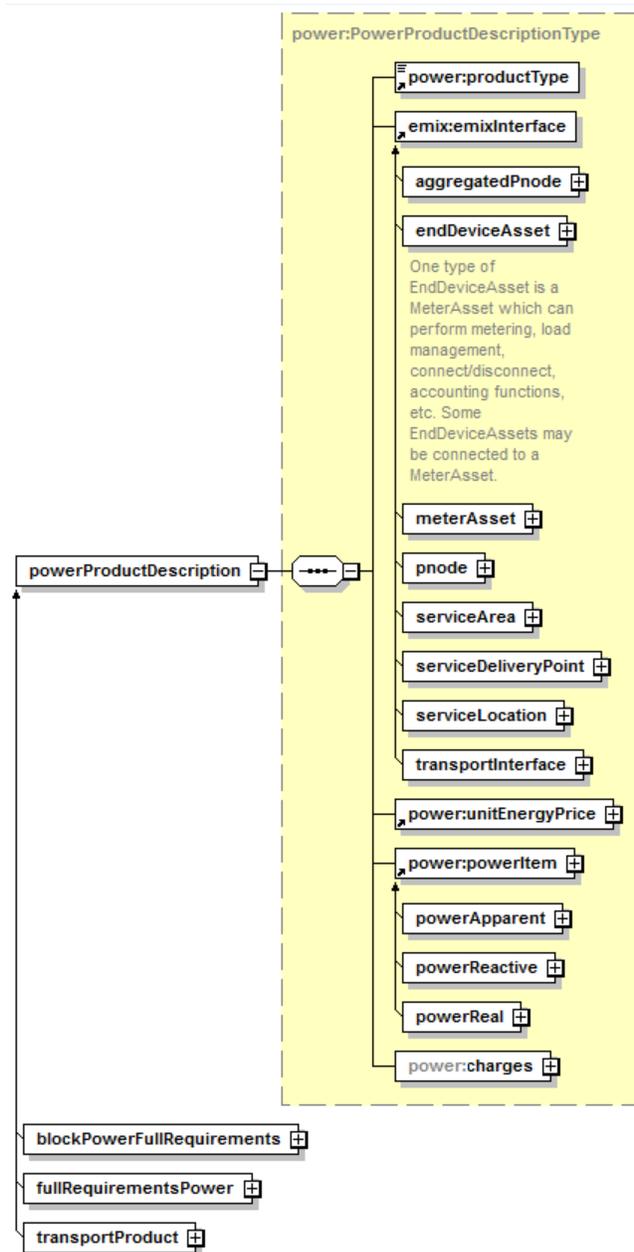
578

579 *Figure 9-1: Base Power Product*

580 The Base Power Contract is the foundation for all the other Power Contracts. Each of them has the
581 characteristics of the Base Power Contract plus their own additional elements:

582 *Table 9-2: Base Power Product Description*

Name	Definition
Power Product Type	Enumerated type of Power Product. Used to determine conformance requirements.

Name	Definition
EMIX Interface	An EMIX Interface is any of a number of market exchange points including a point, an aggregate point, or a geographic area at which a product exchanges ownership
Unit Energy Price	Price per Unit of Energy
Power Item	Can indicate Real, Apparent, or Reactive Power
Charges	Charges affect the power product in addition to the cost of the product delivered. Charges are defined below.

583 Each Power Product is applied to the EMIX Base Product before it is fully described. Because each
584 element can be set for the while Sequence, or applied to individual intervals, each can vary over time.

9.2 Full Requirements Power

586 Full Requirements Power products are the traditional "all-you-can-eat" electrical contract. Maximum
587 delivery is limited by the physical infrastructure. Demand Charges may apply. This type of product often
588 appears in Residential markets.

589 As well as the attributes in the base Power Contract, the Full Requirements Product has:

590 *Table 9-3: Full Requirements Power Product Description*

Name	Definition
Price	EMIX Price is a choice one of a Price, a Price Multiplier, or a Price Relative.
Energy Units	Denominates the units that the Price applies to.
Attributes	Essential characteristics (Voltage, Hz, AC or DC) of delivered electricity.
Maximum Power	Denominates the most power available for transacting during the period.
Minimum Power	Denominates the least power that must be transacted during the Interval. Buyer is responsible for making up the difference if the stated value is not reached.

9.3 Block Power Full Requirements

592 Block Power Full Requirements products provide for full buyer requirement, but prices the power in
593 "blocks". Price is constant within a block, but changes as each block is used during a period. Demand
594 Charges MAY be included. Often used in retail residential rates.

595
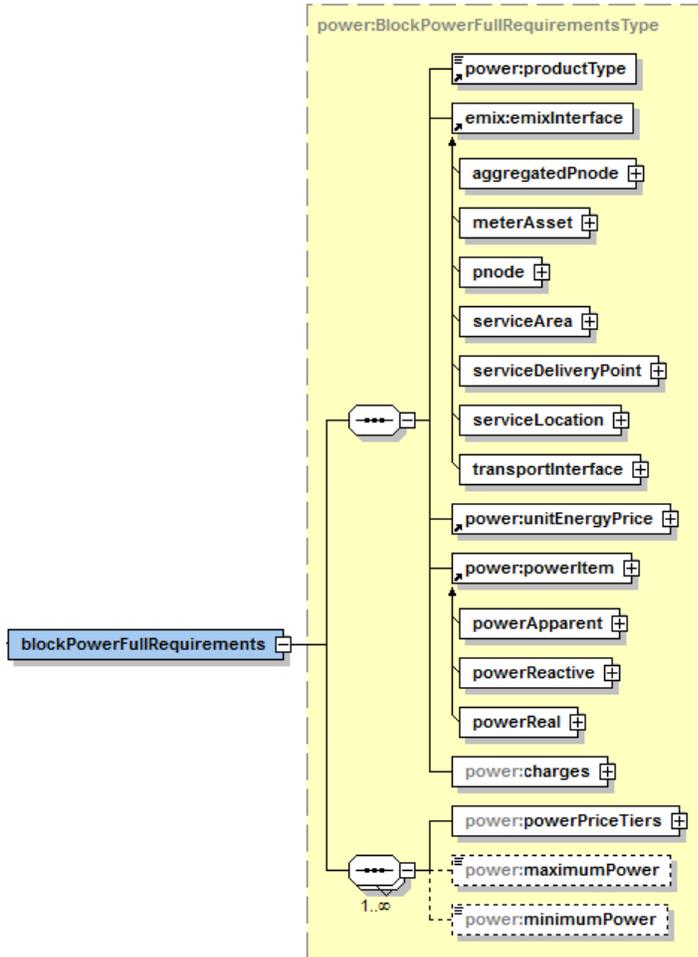
Figure 9-2 Block Power Full Requirements

As well as the attributes in the base Power Contract, the Block Power Full Requirements Product has these additional elements:

Table 9-4: Block Power Full Requirements

Block Power Element	Definition
Block Energy Price	Blocks are sorted in order of Maximum Energy Quantity and price for next block starts when last block is used. Blocks can be confined within an interval to create different tiers at different times of day.
Energy Units	Denominates the units that the Price applies to.
Attributes	For residential, usually 60 Hz, 220V AC
Maximum Power	Denominates the most power available for transacting during the period.
Minimum Power	Denominates the least power that must be transacted during the Interval. Buyer is responsible for making up the difference if the stated value is not reached.

9.4 TEMIX Power Product

601
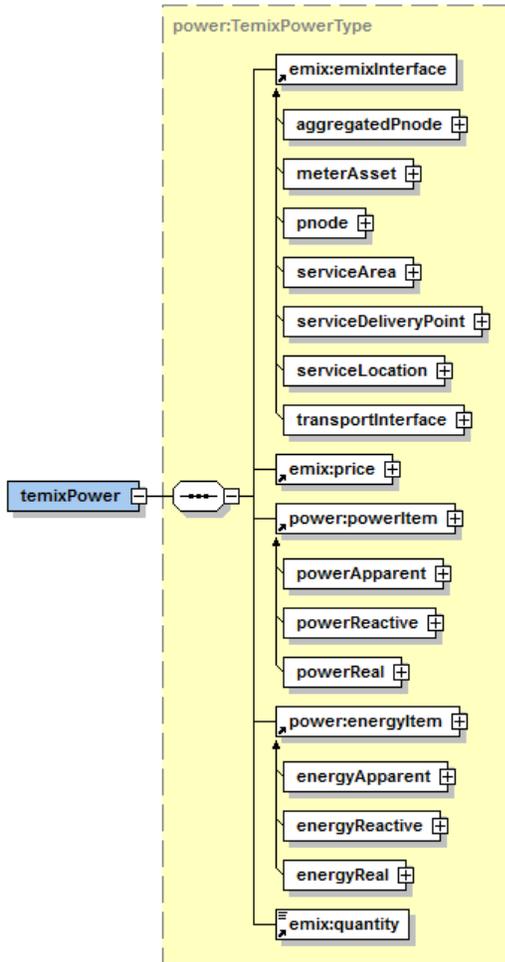
Figure 9-3: TeMIX Power

TEMIX Products are specified by the power (rate of delivery of energy) over an interval. . TEMIX Products
are obligations in that a TeMIX Product is a commitment by the seller to deliver and the buyer to take the
power (energy) over the interval., When the interval includes more than one measurement or metering
interval, the TeMIX product is defined as a constant rate over those metering intervals. An example is the
sale of 1 MW tomorrow between 3 and 5 PM that may be measured every 15 minutes (The energy is 1
MWh). The power in each 15 minute intervals is 1 MW and the energy in each 15 minute interval is 0.25
MWh. A position in a TEMIX product may be resold or added to. Depending on local market rules
differences between the power purchased and the actual delivery may be delivered from or to spot
markets at spot market prices

Table 9-5: TEMIX Power Product Description

TEMIX Element	Definition
Power Product Type	Enumerated type of Power Product. Used to determine conformance requirements.
EMIX Interface	An EMIX Interface is any of a number of market exchange points including a point, an aggregate point, or a geographic area at which a product exchanges ownership

TEMIX Element	Definition
Price	Price per Unit of Energy. For TeMIX, this is always the actual price and not an offset.
Energy Item	Total Energy (Power * Time), Real, Apparent, or Reactive, in the block purchase
Power Item	Rate of Delivery of Energy. Can be Real, Apparent, or Reactive Power, and must match type of Energy Item.

613 TeMIX Product-based information exchanges are a little different than those for other products; they are
614 discussed by themselves in Section 11.

9.5 Power Product Charges

616 Each of the products above, with the exception of TEMIX, can be subject to one or more Power Market
617 Charges. All charges are based on the BasePowerCharge abstract interface, meaning markets the define
618 new charges have the means to define new compliant charges. See the Appendices for a discussion of
619 extensibility in EMIX.

620 Many of the charges defined are specific to Transport Products, although they can be applied to each of
621 the Product herein. See Section 14, *Power Transport Product* Descriptions, for a discussion of those
622 charges.

623 One charge can be applied to each of the Products as above, so is defined here. That charge is the
624 Demand Charge.

625 *Table 9-6: Elements of Power Demand Charges*

Demand Charge Element	Definition
Demand Charge Units	Units upon which Demand Charges will be computed
Demand Charge Floor	Below this floor, demand charges are not applied
Demand Charge Rate	Incremental charge applied if floor is exceeded.
Measurement Interval	Granularity or Power Use readings. For example the demand charge may be incurred of the Power is above the floor for 5 minutes.
Collection Interval	Period during which power usage is summed for comparison to Demand Floor.
Collection Period	Period during which the Demand Charge applies
Charge Duration	Period during which Demand Charges will be applied after incurred.

9.6 Enumerated Power Product Types

627 Because different Power Product Descriptions use the same informational elements, and because
628 different transaction states may not require all elements be present in every exchange, each Power
629 Product Description includes a Power Contract Type. Different Power Contract Types MAY have different
630 conformance requirements in different market contexts. The Power Contract Type MAY be extended per
631 the extensibility rules.

632 The following Power Product Types are enumerated:

633 *Table 9-7: Requirements Power Products*

Power Contract Type	Note

Power Contract Type	Note
Energy	Used in TeMIX for simple block of Energy agreement
Transport	Used in TeMIX for simple transport agreement
Energy Option	Used in TeMIX for Option to transact simple block of Energy
Transport Option	Used in TeMIX for Option to acquire rights to Transport
Full Requirements Power	Traditional power Product to provide all power used. Often used in retail residential rates. Demand Charges
Full Requirements Power with Demand Charge	Similar to Full Requirements except specific and perhaps recurring charges are incurred for exceeding set limit(s)
Full Requirements Power with Maximum and Minimum	Customer must draw energy at least the minimum rate (power) and no more than the maximum rate during any measurement Interval.
Hourly Day Ahead Pricing	Same Full requirements power but prices potentially change each day.
Ex-Ante Real Time Price	Used to report prices after the fact.
Time of Use Pricing	Similar to Hourly day-ahead pricing but prices may change seasonally and not be at hourly Intervals
Transport Service	Product to acquire Transport including factors for congestion, loss, charges, fees, etc.
Congestion Revenue Rights	Hedge product against future Transport / Congestion costs

634　Power products such as these can be described using the Power Product Descriptions

10 Energy Resources

636 The information model in this section is described in RESOURCE.XSD

637 Resources describe potential services to offer to others in a smart grid. Resource tenders are either
638 requesting services or offering services. In a pure transactive market, these tenders might be identical to
639 the services provided, i.e., they could be fully described using the same language used to transact
640 execution and performance.

641 Resources often enter or are called to enter the market to meet specific needs. These needs can include
642 a range of performance requirements; Resources might be able to perform a range of capabilities. These
643 performance capabilities are described using the information in Resource Offers. Resource Offers are
644 less specific than a single transactive request, and may thereby present the Resource to more than a
645 single market.

646 When making a tender for products and services, it is useful to describe the capabilities of a resource, so
647 the counter party can determine if a resource can meet the requirements. A notice of interest MAY specify
648 performance expectations. A Resource MAY compare its own capabilities to those requirements before
649 submitting a bid.

650 Resource Capabilities may describe a ramp rate, or maximum run time, or any number of elements useful
651 to energy schedulers. A Resource Offer associates offers for power produces with a Resource Capability.

10.1 Resource Capabilities

653 Resources have capabilities rather than schedules. Resource descriptions describe what could be done,
654 as distinguished from a transaction in which specific performance is requested or agreed to.
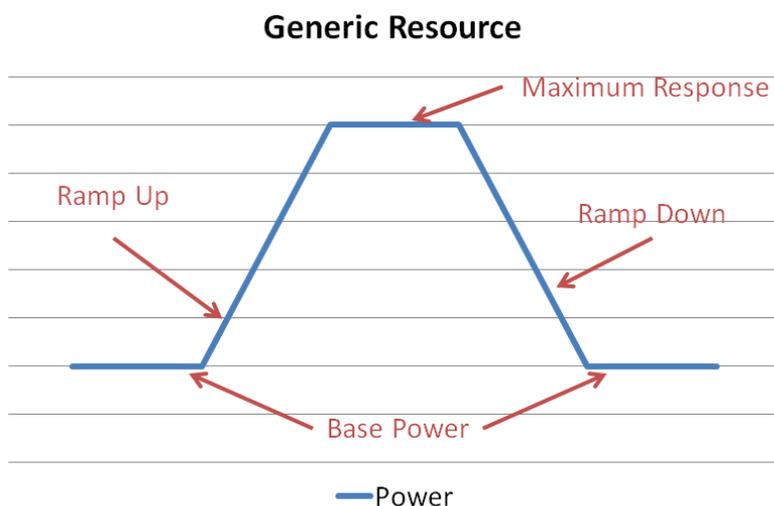
656 *Figure 10-1: Attributes of a Generic Resource*

657 In the Resource illustration above, there is some base level of energy, a *status quo ante*. When invoked,
658 the resource takes some period of time to change to a different level. If the response is binary, then it can
659 only go up to the maximum response, and that ramp rate takes a fixed time. If a resource is able to
660 provide several layers of response, then the ramp time also varies. The ramp time can be computed from
661 the ramp rate and the difference between the base power and the maximum response.

662 As electricity is fungible, a critical key element of power resources is that generation, that is the
663 production of power, and load shedding, the reduction of power use are similar products with similar
664 value.

Equivalence of Load & Generation
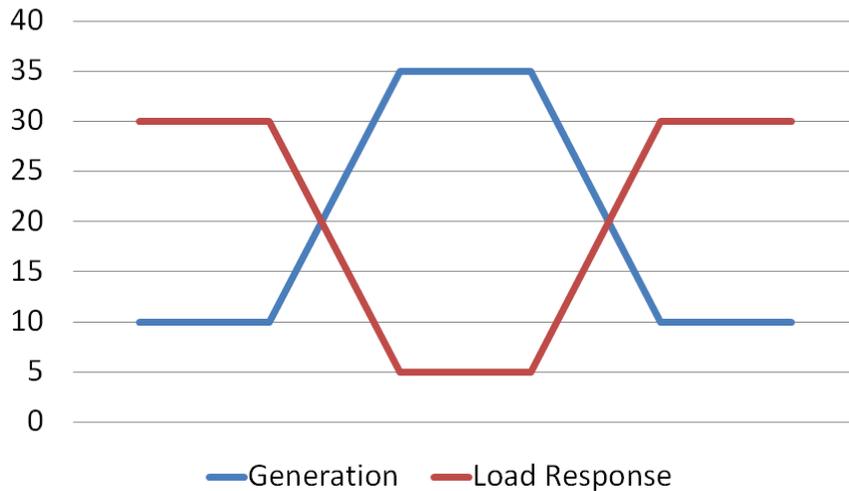
Figure 10-2: Equivalence of Load Shed and Generation

As shown above, generation and load response are similar and can be described using the same language.

Many Resources have capabilities that change over the range of response. A generator may have one ramp speed until it gets up to half speed, and then another as it goes to full. Load response can have similar characteristics. Such resources can be described by combining simple response characteristics.

Generic 2-Level Resource
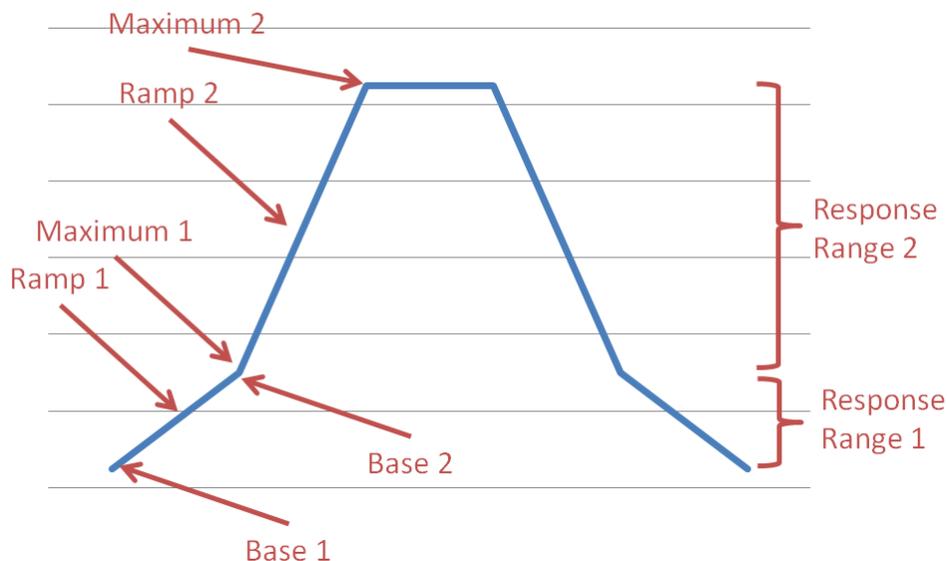
Figure 10-3: Combining Response Capabilities

Resources as in Figure 10-3 can be communicated as an array of ramp up rates, a maximum power offered, and an array of ramp down rates. Between the Base 1 and Maximum 1, expressed in MW, the resource can ramp up at Ramp 1 expressed in MW/min. Between the Base 2 and Maximum 2, expressed in MW, the resource can ramp up at Ramp 1 expressed in MW/min.

678   With capabilities expressed as above, to capabilities of a Resource can be found by the time indicated
679   (moving along the X axis) between Base 1 and wherever the ramp up line passes through desired output
680   level.

681   Users of the IEC TC57 CIM express this with a Ramp Rate Curve. Figure 10-4 expresses similar
682   information as does Figure 10-3, showing Base1 at 50 MW of power and Maximum 1 at 100 MW with a
683   ramp rate of 10 MW/minute. Ramp 2, at 15 MW/minute goes from 100MW to 180 MW.
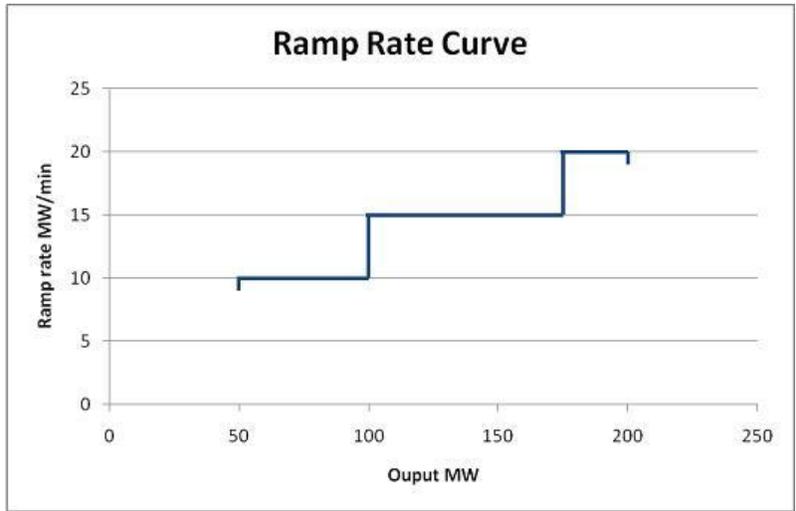
684

685   *Figure 10-4: Ramp Rate Curve—CIM Style*

686   By expressing Resources in terms of capabilities and ramp rates, a potential purchaser can determine if a
687   Resource meets his or her needs, tendering a single resource to a variety of purchase scenarios.

688   Picture several Resources each able to generate 10 MW of additional power. One can increase power at
689   1 MW/minute, one at 2 MW/minute, one at 5 MW/minute. The latter two each can enter into an
690   Agreement to supply 10 MW in 5 minutes. Only the last can Agree to supply an increase of 10 MW within
691   2 minutes. All three can Agree to supply an increase of 10 MW within 15 minutes.

10.2 Resource Description Semantics

693   EMIX Resource Descriptions are an extension of the EMIX Product Description. As an extension of the
694   Product Description, resources can be applied inside any EMIX schedule.
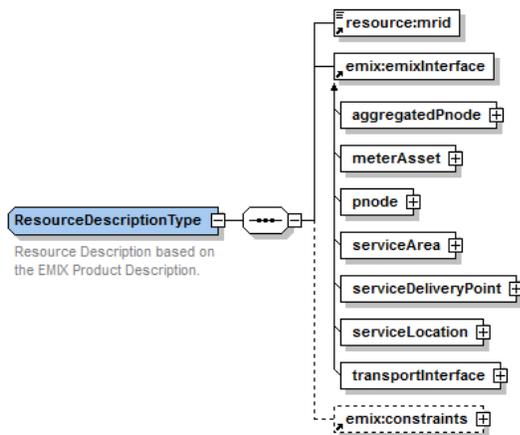
695

696   *Figure 10-5: Resource Description base*

697   The only aspects of a Resource that matters to the energy market are the effects it can provide, the
698   likelihood it will be able adequately to provide what it promises, and the financial incentives required to

699 acquire them. The technology and process control details are many, and new ones may be required for
700 each new power technology. Unless the market for the Resource requires direct control, such details are
701 irrelevant. The limited semantic set herein is sufficient to describe the capabilities of a Resource.

702 The EMIX Resource Description base consists of:

703 *Table 10-1: Resource Description Elements*

Resource Description Element	Note
MRID	The Multi-part resource id as defined in the [**IEC TC57**] CIM uniquely identifies each resource.
EMIX Interface	The Interface is where the Resource injects or extracts power. Note: for many transactions, reduced extraction is equivalent to injection.
Constraints	As well as all of the constraints listed for Product performance, Resources have additional constraints, listed below.

704 Power Resources descriptions can use any of the constraints or requirements defined in EMIX. Power
705 Resource descriptions can also use additional constraints that are specific to Power:

706 *Table 10-2: Constraints unique to Power Resources*

Power Constraint	Note
Minimum Load	Constraint on Minimum Load that a Resource can maintain
Maximum Power	Constraint on Maximum Power available from a resource
Maximum Energy	Constraint on Maximum Energy available from a resource
Minimum Load Reduction	Constraint on Minimum Load Reduction resource can make

10.3 Generic Power Resource

707

708 The Generic Power Resource description is used both for generation and for load Resources. The
709 common Resource model is as follows:

710 *Table 10-3: Generic Power Response Resource*

Generic Resource Element	Note
Staging Ramp	An array Power Ramp Segments describing a Resource's ability to change level at the initiation of a Response
Minimum Response	The least Response for which this resource will accept a request.
Maximum Response	The greatest Response for which this resource will accept a request
Recovery Ramp	An array Power Ramp Segments describing how a Resource's returns to its original state following a response.

711 A Power Response Description MAY be accompanied by an Offer Curve *(described in section 10.3.1)*.

712 Each Ramp consists of zero to many Power Ramp Segments (see figure Figure 10-3: Combining
713 Response Capabilities). Each Power Ramp Segment Rate describes a change up or down in
714 units/duration, from the Power Quantity of the Begin Ramp to the Power Quantity of the End Ramp. The
715 rate of change is assumed to be constant between the Begin Ramp and the End Ramp.

716 Power Ramp Segments consist of the following elements:

717 *Table 10-4: Power Ramp*

Power Ramp Element	Note
Rate	Power Units for the Ramp
Begin Ramp	Power Quantity at the beginning of the Segment
End Ramp	Power Quantity at the end of the Segment
Duration	The time to get between the begin ramp and the end ramp.
Integral Only	If true, one can't stop between the begin and end rates.

718 While Power Ramps are generic, specific instances within derived Resource Descriptions are subject to
719 different conformance rules.

720 For a Generation Resource, Staging Ramps are processed in order of increasing End Power. The
721 quantity of End Power MUST be greater than the quantity of the Begin Power for each Ramp in the
722 Staging Ramp. Recovery Ramps are processed in order of decreasing End Power. The quantity of End
723 Power MUST be less than the quantity of Begin Power for each Ramp in the Recovery Ramp.

724 For a Load Resource, Staging Ramps are processed in order of decreasing End Power. The quantity of
725 End Power MUST be less than the quantity of Begin Power for each Ramp in the Staging Ramp.
726 Recovery Ramps are processed in order of increasing End Power. The quantity of End Power MUST be
727 greater than the quantity of the Begin Power for each Ramp in the Recovery Ramp.

728 Load Resources and Power Resources are conformed instances of the Generic Power Resource.

10.3.1 Offer Curves

730 When a Power Resource is offered to the market, it may be accompanied by an Offer Curve. An Offer
731 Curve is comprised of a number of Offer Segments. An Offer Segment defines the minimum requirements
732 (as expressed in EMIX Requirements) of the Offeror for each block of response without which the Offeror
733 will withdraw the Resource from the market.

734 *Table 10-5: Resource Offer Segment*

Resource Offer Element	Note
Price	Price required for this Segment
Maximum Response	Enumerator for the Power rate at the beginning of this Rampaximum Power change in this segment
Units	Power Quantity at which the Ramp Ends

735 Because an Offer Curve is always figured in terms of the block size of the response, it is always sorted in
736 order of increasing response. In many markets, they Offer Curves are then processed as a series of bids.

10.4 Reactive Power Resources

738 In addition, voltage regulation services have their own semantics.

739 *Table 10-6 Semantics for Voltage Regulation Services*

Voltage Regulation Element	Note
VMin	VMin is the IEEE 1547 minimum voltage level of 88% of nominal voltage where the PV inverter must disconnect. Also defined as the minimum Reactive Power of the Resource.

Voltage Regulation Element	Note
VMax	VMax is the IEEE 1547 maximum voltage level of 110% of nominal voltage where the PV inverter must disconnect. Also defined as the Maximum reactive power of the Resource
QMax	Qmax is the inverter's current var capability and may be positive (capacitive) or negative (inductive). It is the VA capability left after supporting the W demand.
voltVar	Reactive Power

10.5 Summary of Resource Types
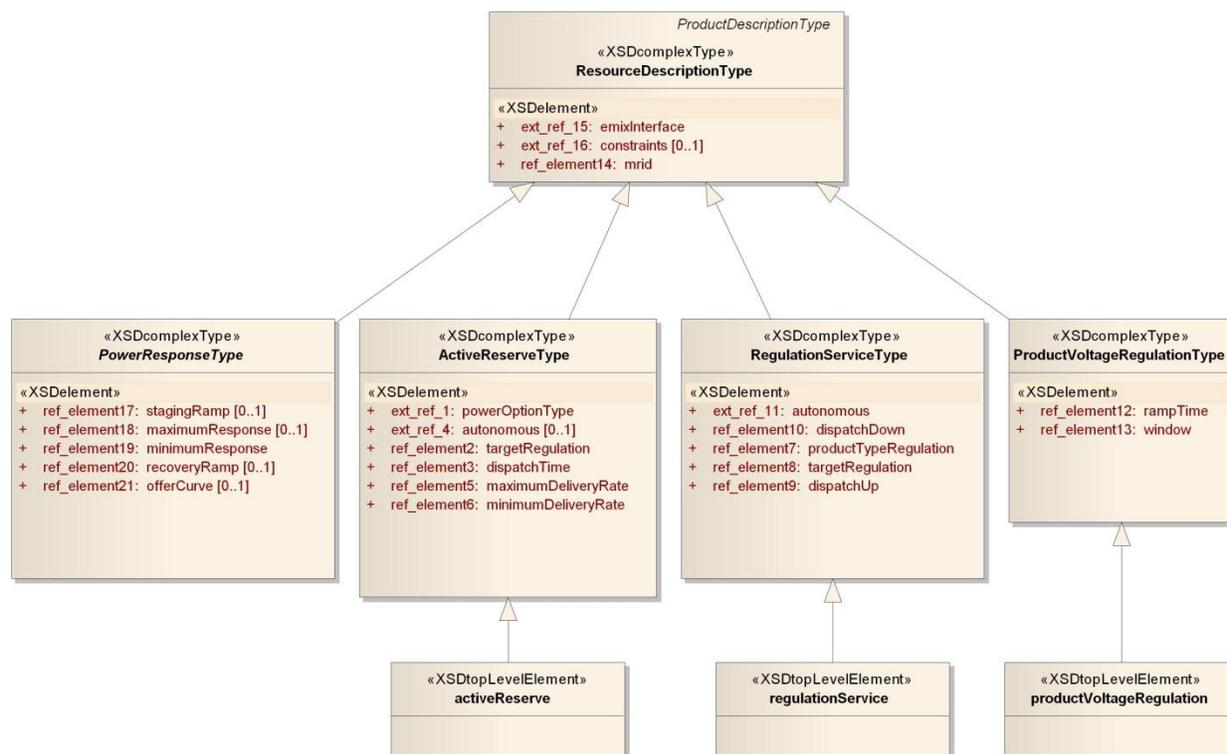
Figure 10-6: UML Summary of Resource Types

11 Transactive Energy (TeMIX) Products

TeMIX products use transactive interactions to acquire blocks of power. It emphasizes simple interactions and requires minimal knowledge of one's trading partner. All TeMIX Products are subscriptions for power over a single Delivery Interval. Subscriptions impose an obligation on the buyer to purchase and the seller to deliver a TeMIX Power Product. This simplicity reduces the number of products and interactions.

There are only four types of TeMIX Products:

1. TeMIX Power Product
2. TeMIX Transport Product
3. TeMIX Option Power Product
4. TeMIX Option Transport Product

The Transactive States for a TeMIX Product are:

- Indication of Interest
- Tender
- Transaction
- Delivery
- Price Publishing

A TeMIX Delivery Interval is specified by a Duration and Start Time. When TeMIX Product is specified for a set of Delivery Intervals, then elements that do not vary by Delivery Interval may be specified in a Gluon. However each TeMIX Delivery Interval is transacted independently of the others in the set.

A TeMIX Power Product defines a subscription for Power (energy = power * duration) over a Delivery Interval. The subscribed power of TeMIX Power Product is constant over all measured (metered) intervals within a TeMIX Delivery Interval.

For example, 1 MW of power subscribed for delivery tomorrow for 2-hours between 3 and 5 PM provides 1 MWh of energy over each hour and 2-MWh over the two hours. If delivery is measured every 15-minutes, then the power subscribed in each 15 minute interval is 1 MW. The energy subscribed in each 15-minute interval is 0.25 MWh. If the energy delivered in each 15-minute interval is greater or less than 0.25 MWh then the balance (positive or negative) will typically be sold or purchased in a subsequent balancing transaction.

The Price of a TeMIX Product is expressed in energy units. For the example above, when the price is $80 per MWh of energy, the extended price (cost) of 1 MW of Power for 2- hours between 3 and 5 PM is $160 and the extended price for 1 MW of Power in each 15-minute interval of the 2-hours is $20.

A TeMIX Transport Product is a subscription for Transport (transmission or distribution) to transport a TeMIX Power Product from one EMIX Interface to another. A TeMIX Transport Product is a subscription for power transport at a constant rate over the delivery interval.

A TeMIX Option Product is a subscription for optionality applied to a TeMIX Power or Transport Product. A TeMIX Option Product is a subscription that provides the Option Holder a right to instruct the Option Writer to deliver (call) or take (put) a TeMIX Power or Transport Product up to the subscribed quantity (rate of delivery) of the Option at a Strike Price.

TeMIX Options are either Call or Put Options on TeMIX Power and Transport Products. A TeMIX Option can be exercised during the Delivery Interval of the Option for any subinterval not smaller than the Option Interval Granularity.

For example a TeMIX Option for 10 MW for a Day and an Option Interval Granularly of 1-hour and an Option Lead Time of 30 minutes would allow the Holder to exercise the option for any or all hours of the Day at the Strike Price by giving notice 30 minutes before each hour.

787 The elements of a TeMIX Power and Transport Product are shown in **Error! Reference source not**
788 **found.**. When the Product Description (from section 9) is applied to the EMIX Base types, the TeMIX
789 elements are:

790   *Table 11-1: TeMIX Power Product Description*

TeMIX Element	Definition
Power Product Type	Enumerated type of Power Product. Used to determine conformance requirements.
EMIX Interface	The TeMIX Interface where the transaction occurs. Generally the Interface for a Power Product has one node and the Interface for a Transport Product has two nodes.
Price	Price per Unit of Energy. For TeMIX, this is always the actual price and not an offset.
Start Date and Time	When the Interval begins
Duration	The length of time of the Interval
Price	The Unit Energy Price for the interval. TeMIX does not allow Relative Prices or Price Multipliers.
Energy Item	Total Energy (Power * Time), Real, Apparent, or Reactive, in the block purchase
Power Item	Units for the Rate of Delivery of Energy for the Delivery Interval. Includes Power Attributes.
Power Quantity	Rate of Delivery of Energy for the Delivery Interval.
Transactive State	TeMIX Transactive state is conformed to Indication of Interest, Tender, Transaction, Delivery or Publish.
Currency	Currency for the exchange
Side	Indicates which side of the agreement the information originator is on. Buy or Sell
Expires Date	Date and Time Tender expires. Not present if the Transactive State is anything other than Tender.
Envelope	As defined in Section 4.1.5

791 The TeMIX Option extends the TeMIX Product by adding these additional elements:

792   *Table 11-2: TeMIX Power Option Product Description*

TeMIX Element	Definition
Option Holder	The side (buy or sell side of the option) which enjoys the benefit of choosing whether or not to exercise the option. The other side is the Option Writer
Strike Price	The price at which the Option Holder can require Option Writer to deliver.
Option Lead Time	The Minimum Notification Duration constraint
Option Schedule	The Availability Schedule constraint

TeMIX Element	Definition
Minimum Option Call	The shortest duration for which the Option can be called. Uses the Minimum Run Duration constraint
Granularity	If present, expresses the temporal granularity of requests as a duration. For example, if the Duration is 15 Minutes, the option can be called at10:00, 10:15, 10:30, or 10:45.

12 Ancillary Services

Ancillary Services are typically products provided by a Resource Capability and historically were contracted to stand by for a request to deliver changes in power to balance the grid on very short notice. Ancillary services include Regulation Up, Regulation Down, Spinning Reserve, and Non-Spinning Reserve. These Ancillary services are different from other power products in that they are paid for availability, whether or not they perform. Of course, they must also perform when called.

In general, Ancillary services are a promise to perform, usually within tight constraints, i.e., within five minutes of notification in one market, or within one minute of notification for another. The promisee pays for this offer to perform, whether or not the promise is called. When the performance call comes, the promisor is then paid again, often at a premium over the market rate at the time of the performance call.

In EMIX, Reserves are modeled as simple Options using the EMIX Option type, which is one of the EMIX Base-derived types. Performance expectations are expressed using constraints, which can appear on either side of a Tender. Strike prices and the penalty for non-performance are part of the option agreement.

Because it is useful to have a short-hand to refer to these services, they are enumerated in the Power Option Type enumeration which is incorporated into the Power Product Types.

The enumerated Power Option Types are: Spinning Reserve, Non Spinning Reserve, Operating Reserve, and Demand Response. Because the exact definitions vary from market to market, and will continue to vary over time, EMIX does not define these terms. All definitions and performance requirements SHALL be expressed through the constraints.

13 Power Quality

814 The information model in this section is described in POWER-QUALITY.XSD

815 Higher quality power can obtain a market premium. A buyer willing to accept lower quality power may be
816 able to obtain inexpensive power. Power Qualities must be measurable, discrete, and on a spectrum
817 allowing the buyers to make choices. They must also be verifiable, measurable by defined protocols, so
818 performance can be compared to promise.

13.1 Electrical Power Quality

820 *Table 13-1: AC Power Quality*

Name	Specification
Measurement Protocol	A string containing an identification of the standard or other protocol used to measure power quality
Power Frequency	A floating point number describing the measured Power frequency. Users who wish to describe how the frequency varies over time will need to derive their own measure from the base Powr Quality type.
Supply Voltage Variations	An unsigned integer count of Supply Voltage Variations during the period
Rapid Voltage Changes	An unsigned integer count of Rapid Voltage Change events during the period
Flicker	An unsigned integer count of Flicker events during the period
Supply Voltage Dips	An unsigned integer count of Supply Voltage Dip events during the period
Short Interruptions	An unsigned integer count of Short Interruption events during the period
Long Interruptions	An unsigned integer count of Long Interruption events during the period
Temp Overvoltage	An unsigned integer count of Temporary Overvoltage events during the period
Supply Voltage Imbalance	An unsigned integer count of Supply Voltage Imbalance events during the period. Not meaningful for DC.
Harmonic Voltage	A floating point number for the Harmonic Voltage during the period. For DC, distortion is with respect to a signal of 0 Hz
Mains Voltage	A floating point number Mains [Signaling] Voltage

821

14 Power Transport Product Descriptions

823 The information model in this section is described in POWER-PRODUCTS.XSD

824 Transport costs affect the delivery of energy in all markets. Today's electrical power markets use different
825 terms in transmission and delivery, but the underlying elements are the same. Future markets, including
826 those for microgrids and virtual service providers, may not make the same distinctions between
827 transmission and distribution as have been made in the past. Distributed Energy Resources (DER) may
828 create new business models for use of the existing distribution networks.

829 The information model below merges the charges and approaches used in the respective transmission
830 and distribution networks today. It anticipates that potential source selection markets may result in
831 passage through multiple networks. The resulting EMIX Base can support either stand-alone transport
832 products, or price support information conveyed within the Envelope, in support of Locational Marginal
833 Pricing (LMP).

834 *Table 14-1: Transport Description*

Transport Product Element	Definition
Point of Receipt	Where power enters a network or changes ownership
Point of Delivery	Where power exits a network or changes ownership
Transport Access Fee	Fixed Charge (not dependent on congestion) to access transport system
Transport Congestion Fee	Congestion fee per unit of energy for energy flowing from receipt to delivery point. Can be a positive or negative price.
Marginal Loss Fee	Marginal Loss Fee
Transport Loss Factor	Reduction in amount delivered due to loss during transport. (Loss Factor * purchase amount) = delivered amount
Conversion Loss Factor	Reduction in amount delivered as product voltage is changed or as converted from AC to DC or DC to AC. (Loss Factor * purchase amount = delivered amount)

835 There MAY be multiple instances of the above Artifacts in a single Price instance. For example, in a given
836 transaction, power may pass through multiple distribution nodes and congestion points.

837 The items listed in the table above are each derived from the base charge type. All other charges,
838 previously described, are available for inclusion within a Transport Product.

15 EMIX Warrants

840   The information model in this section is described in EMIX-WARRANTS.XSD

841   Warrants are specific assertions about the extrinsic characteristics of EMIX Products that may affect
842   market pricing. Warrants are in effect Product artifacts as defined in EMIX. Warrants are extensions of the
843   Product Descriptions type that are applied the Intervals in a Schedule. There may be zero Intervals in a
844   Product if the unchanged product description applies to all.

845   The Intervals in a warrant may differ from those of the Product on the outside of the envelope.

846   Some warrants may be applicable only in certain jurisdictions. For example, in today's energy markets
847   (2011) energy warranted as renewable in the Pacific Northwest can include hydropower. Energy markets
848   in California exclude hydropower from their definition of renewable power. Credits or mandates for
849   renewable energy in California are not met by Products warranted as renewable in the Northwest.

850   Some warrants may be separable from the underlying energy. For example, a warrant that energy is
851   generated by a source that is certified as "green" by an authority, may be issued a "green certificate". In
852   some markets, such a certificate can be traded separately. The detailed specification of warrants is not
853   part of version 1.0 of this specification.

15.1 Warrant List Definition

855   Warrant Types are abstract types defined in this specification for extension and definition elsewhere.
856   Conforming information exchanges can include schema types derived from these types.

857   *Table 15-1: Warrant Types*

Warrant Type	Note
Product Quality	If during an offer, can be a promise of quality. If during verification, and be actual measurements. If during an indication of interest, might be a minimum standard.
Environmental Warrant	Quantifies the environmental burden created during the generation of the electric power.
Content Warrant	The proportion of the product defined that is from non-fossil fuel sources, including but not limited to "hydroelectric", "nuclear", "solar", and "wind".
Source Warrant	In aggregate may be the same as a Warrant Content
Controllability Warrant	Assertion that a Resource referenced on the face of the envelope can be controlled and/or operated by or to some standard.

858

16 Conformance and Rules for EMIX and Referencing Specifications

This section specifies conformance related to the semantic model of EMIX. EMIX is heavily dependent upon WS-Calendar, and repeatedly incorporates WS-Calendar-based information models.

EMIX artifacts can be exchanged at any of several stages of a transaction. Necessarily, a tender must be able to accept an incomplete information model while a call for execution must fully define the performance expected. Specifications referencing EMIX SHALL define conformance rules by transaction type and market context.

EMIX Conformance necessarily occurs in two stages. EMIX uses WS-Calendar to communicate similar intervals that occur over time, each containing an EMIX artifact. Portions of that artifact may be expressed within the Lineage of the sequence. Applications MUST apply WS-Calendar Inheritance and then EMIX Inheritance to Compose the information exchange for each interval. Only after Composition, can the EMIX artifact within each Interval of the Sequence be evaluated for conformance and completeness.

16.1 EMIX Conformance with WS-Calendar

EMIX Base are EMIX Products and Resources instantiated through the schedule model of WS-Calendar. As such, EMIX Base SHALL follow WS-Calendar Conformance rules. These rules include the following conformance types

- Conformance to the *inheritance rules* in WS-Calendar, including the direction of inheritance
- *Specific attributes* for each type that MUST or MUST NOT be inherited.
- *Conformance rules* that Referencing Specifications MUST follow
- Description of *Covarying attributes* with respect to the Reference Specification
- *Semantic Conformance* for the information within the artifacts exchanged.

EMIX Products and Resources also extend the Inheritance patterns of WS-Calendar to include the EMIX information model. We address each of these in the following sections.

16.1.1 Inheritance in EMIX Base

In this section we recapitulate the rules that define inheritance including direction in WS-Calendar.

I1: Proximity Rule Within a given lineage, inheritance is evaluated though each Parent to the Child before what the Child bequeaths is evaluated.

I2: Direction Rule Intervals MAY inherit attributes from the nearest gluon subject to the Proximity Rule and Override Rule, provided those attributes are defined as Inheritable.

I3: Override Rule If and only if there is no value for a given attribute of a Gluon or Interval, that Gluon or Interval SHALL inherit the value for that attribute from its nearest Ancestor in conformance to the Proximity Rule

I4: Comparison Rule Two Sequences are equivalent if a comparison of the respective Intervals succeeds as if each Sequence were fully Bound and redundant Gluons are removed.

I5: Designated Interval Inheritance [To facilitate composition of Sequences] the Designated Interval in the ultimate Ancestor of a Gluon is the Designated Interval of the composed Sequence. Special conformance rules for Designated Intervals apply only to the Interval linked from the Designator Gluon.

I6: Start Time Inheritance When a start time is specified through inheritance, that start time is inherited only by the Designated Interval; the start time of all other Intervals are computed through the durations

899 and temporal; relationships within the Sequence. The designated Interval is the Interval whose parent is
900 at the end of the lineage.

16.1.2 Specific Attribute Inheritance in EMIX

902 This section refers to EMIX Products, agreements, and Resources as Artifacts. In general, if an artifact of
903 a particular type blocks inheritance of a complete artifact of that type down the lineage.

904 If an Artifacts of the same type exist in both the parent and in the child, the prototypical argument can be
905 discussed two-dimensional tree with branches. Blended inheritance consists of deciding when to graft a
906 branch onto the root.

907 The root node of parent and the child must match for blended inheritance to occur, that is, the roots must
908 be of the same type. The exception is if there are no roots the child's Artifact, then the root and all its
909 branches are inherited by the child.

910 If matching roots are found in both the parent and in the child, then each tree should be navigated to
911 determine blended inheritance. The child's artifact may be mostly unpopulated. Within any branch in the
912 child, the first node that is populated blocks all further inheritance on that branch. All nodes deeper in the
913 Artifact than that populated node, are determined by the child. When a branch is inherited from the child,
914 it blocks the inheritance of any deeper nodes within that branch.

915 Specific artifacts may declare rules that break this inheritance pattern. As of now, the exceptions are:

916   -   There are no exceptions.

917 Inheritance creates a virtual artifact at each level of processing. That virtual artifact is the basis for
918 inheritance for any child artifact.

919 In EMIX the following attributes MUST NOT be inherited

920   • UID (Gluons and Intervals)

921   • Temporal Relationships

922 Some elements of EMIX are may be **covarying**, meaning that they change together. Such elements are
923 treated as a single element for inheritance, they are either inherited together or the child keeps its current
924 values intact. This becomes important if one or more of a covarying set have default values. In that case,
925 if any are present, then inheritance should deem they are all present, albeit some perhaps in their default
926 values.

16.2 Miscellaneous Business Rules not yet dealt with.

928 If the first Interval in a series has a price only, all Intervals in the Sequence have a price only and there is
929 no price in the Product

930 If the first Interval in a series has a quantity only, all Intervals in the Sequence have a quantity only and
931 there is no quantity in the Product

932 If the first Interval in a series has a price & quantity, all Intervals in the Sequence MUST have a Price and
933 Quantity and there is neither Price not Quantity in the Product

934 All Intervals in a Sequence may be restricted to single service location. What are the rules?

935

A. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Bruce Bartell, Southern California Edison
Timothy Bennett, Drummond Group Inc.
Edward Cazalet, Individual
Toby Considine, University of North Carolina at Chapel Hill*
William Cox, Individual
Sean Crimmins, California Independent System Operator
Phil Davis, Schneider Electric
Sharon Dinges, Trane
Pim van der Eijk, Sonnenglanz Consulting
Girish Ghatikar, Lawrence Berkeley National Laboratory
Todd Graves, Microsoft Corporation
Anne Hendry, Individual
David Holmberg, NIST*
Gale Horst, Electric Power Research Institute (EPRI)
Ali Ipakchi, Open Access Technology International Inc. (OATi)
Perry Krol, TIBCO Software Inc.
Derek Lasalle, JPMorganChase
Jeremy Laundergren, Southern California Edison
Alex Levinson, Lockheed Martin*
Dirk Mahling, CPower
Scott Neumann, Utility Integration Solutions Inc.
Robert Old, Siemens AG
John Petze, Individual
Donna Pratt, ISO/RTO Council
Ruchi Rajasekhar, Midwest Independent Transmission System Operator, Inc.
Carl Reed, Open Geospatial Consortium, Inc. (OGC)*
Jeremy Roberts, LonMark International*
Anno Scholten, Individual
Aaron F. Snyder, EnerNex
Pornsak Songkakul, Siemens AG
Bill Stocker, ISI/RTO Council (IRC)
David Sun, Alstom Power Inc.
Jake Thompson, EnerNOC
Matt Wakefield, Electric Power Research Institute (EPRI)
David Webber, Individual
Leighton Wolffe, Individual
Brian Zink, New York Independent System Operator (NYISO)

977 B. Extensibility and EMIX

978 Extensibility was a critical design constraint for EMIX. Extensibility allows the EMIX specification to be
979 used in markets and in interactions that were not represented on the Technical Committee. Formal
980 extensibility rules also create a set of complaint extensions for incorporation into later versions that are
981 already compliant.

982 B.1 Extensibility in Enumerated values

983 EMIX defines a number of enumerations. Some of these, such as measurements of power, are
984 predictably stable. Others, such as market contracts or energy sources, may well have new elements
985 added. In general, these accept any string beginning with "x-" as a legal extension. In particular, these
986 are defined using the following mechanism in the formal schemas (XSD's).

987 In emix.xsd, the extensibility pattern is defined. This pattern looks like:

```
988  <xs:simpleType name="EMIXExtensionType">
989     <xs:annotation>
990         <xs:documentation>Pattern used for extending string enumeration,
991  where allowed</xs:documentation>
992     </xs:annotation>
993     <xs:restriction base="xs:string">
994         <xs:pattern value="x-\S.*"/>
995     </xs:restriction>
996  </xs:simpleType>
```

997 Non-extensible enumerated types look like this:

```
998  <xs:simpleType name="PowerOptionTypeEnumeratedType">
999     <xs:annotation>
1000         <xs:documentation>Power Reserve Options</xs:documentation>
1001     </xs:annotation>
1002     <xs:restriction base="xs:string">
1003         <xs:enumeration value="SpinningReserve"/>
1004         <xs:enumeration value="NonSpinningReserve"/>
1005         <xs:enumeration value="OperatingReserve"/>
1006         <xs:enumeration value="DemandResponse"/>
1007     </xs:restriction>
1008  </xs:simpleType>
```

1009 The enumerations used in the specifications look like this.

```
1010     <xs:element name="powerOptionType" type="power:PowerOptionTypeType"/>
1011     <xs:simpleType name="PowerOptionTypeType">
1012         <xs:union memberTypes="power:PowerOptionTypeEnumeratedType
1013  emix:EmixExtensionType"/>
1014     </xs:simpleType>
```

1015 This pattern has been followed throughout EMIX, allowing any string beginning "X-" to be a legal
1016 extension numeration for EMIX enumerated strings.

1017 Some extensible enumerated types are planned for extension. For example, the means of measurement
1018 for power quality are defined specific testing protocols. As of this writing, there are only two testing
1019 protocols in the specification.

```
1020     <xs:simpleType name="MeasurementProtocolEnumeratedType">
1021         <xs:restriction base="xs:string">
1022             <xs:enumeration value="EN 50160"/>
1023             <xs:enumeration value="IEEE 1549-2009"/>
1024         </xs:restriction>
```

```
1025        </xs:simpleType>
```

We anticipate that other protocols will be used. In this case, we use the suffix "EnumeratedType" to allow
for the possibility of other Measurement Protocols that are not enumerated. Actual compliance, though, is
based upon the type:

```
1029        <xs:simpleType name="MeasurementProtocolType">
1030             <xs:union memberTypes="power:MeasurementProtocolEnumeratedType
1031        emix:EMIXExtensionType"/>
1032        </xs:simpleType>
```

That is, valid values for the measurement protocol are the enumerated values, and any that match the
extension pattern "x-*"

EMIX defines extensibility for the following values:

- [Quality] Measurement Protocol
- Contract Type
- Option Type
- Power Option Type
- Resource Type

B.2 Extension of Structured Information Collective Items

EMIX anticipates adding some information structures that are more complex than simple strings can be
extended as well. A challenge for these items is that they are more complicated and so require formal
definition. Formal definitions, expressed as additions to schema, could require changes to the
specification. Without formal definition, it is difficult for trading partners to agree on valid information
exchanges.

EMIX uses abstract classes for many information exchanges. For example, trading partners could agree
on the exchange of larger or smaller lists of quality measures. Many measures of power quality are
defined in power-quality.xsd. Quality consists of an array of elements that are derived from the abstract
base quality element.

```
1051    <xs:complexType name="PowerQualityType">
1052      <xs:annotation>
1053             <xs:documentation>Power Quality consists of a number of measures,
1054      based on contract, negotiation, and local regulation. Extend Power Quality to
1055      incorporate new elements by creating additional elements based on
1056      PowerQualityBaseType</xs:documentation>
1057      </xs:annotation>
1058      <xs:sequence>
1059      <xs:element name="measurementProtocol"
1060    type="power:MeasurementProtocolType"/>
1061             <xs:element name="constraints" type="power:ArrayOfPowerQualities"/>
1062      </xs:sequence>
1063    </xs:complexType>
```

A practitioner who wanted to add an additional quality type would need to develop a description and
instantiation of that type based on the abstract base, similar to that used below. The implementation
refers to the substitution group:

```
1067        <xs:element name="supplyVoltageVariations"
1068        type="power:SupplyVoltageVariationsType"
1069        substitutionGroup="power:basePowerQualityMeasurement"/>
```

and the type extends the abstract base class BasePowerQualityMeaurementType:

```
1071        <xs:complexType name="SupplyVoltageVariationsType" mixed="false">
1072            <xs:complexContent mixed="false">
```

```
1073            <xs:extension base="power:BasePowerQualityMeasurementType">
1074                <xs:sequence>
1075                    <xs:element name="count" type="xs:int"/>
1076                </xs:sequence>
1077            </xs:extension>
1078        </xs:complexContent>
1079    </xs:complexType>
```

1080 The resulting schema, which references the approved EMIX schemas, but does not change them, can
1081 then be distributed to business partners to validate the resulting information exchanges. The core EMIX
1082 types, which are used throughout the specifications herein, can be extended this way, including:

- **EMIX Base Type**: iCalendar-derived object to host EMIX Product Descriptions

1084 - **Product Description Type**: In EMIX, the Product Description is the basis for all Resources and
1085 Product Descriptions.

1086 - **Item Base**: Abstract base class for units for EMIX Product delivery, measurement, and warrants.
1087 Item does not include Quantity or Price, because a single product description or transaction may
1088 have multiple quantities or prices associated with a single item.

1089 - **EMIX Interface**: Abstract base class for the interfaces for EMIX Product delivery, measurement,
1090 and/or pricing.

1091 The following additional abstract types are among those designed with extension by practitioners in mind:

1092 - **BasePowerQualityMeaurementType**: the basis for exchanging measurements of power quality

1093 - **BaseConstraintType**: used to express constraints on the performance of equipment exposed to
1094 the market as Resources

1095 - **BaseRequirementType**: used to express the market or business requirements of a trading
1096 partner.

1097 - **BaseWarrantType**: the root for all warrants delivered with the energy product.

1098 C. Semantics from WS-Calendar

1099 Certain terms appear throughout this document that are defined in **[WS-Calendar]**. This section provides
1100 summary definitions for the convenience of the reader and reviewer. Nothing in this table replaces or
1101 over-rides the normative definitions in that specification.

1102 *Table C-16-1: WS-Calendar Foundational Semantics*

Time Segment	Definition
Duration	Duration is the length of an event scheduled using iCalendar or any of its derivatives.
Interval	The Interval is the core component of duration and sequence. Parties make Agreements for delivery of EMIX-described products during an Interval.
Sequence	A Sequence is a set of Intervals with defined temporal relationships. Sequences may have gaps between Intervals, or even simultaneous activities. A Sequence may be re-locatable, i.e., it does not require a specific date and time. A Sequence may consist of a single Interval. A Sequence MAY include a Lineage.
Partition	A Partition is a set of consecutive Intervals. The Partition includes the trivial case of a single Interval. Many energy negotiations apply an EMIX product to a partition, e.g., consecutive fifteen minute Intervals.
Gluon	A Gluon is influences the serialization of Intervals in a Sequence, though inheritance and through schedule setting. The Gluon is similar to the Interval, but has no service or schedule effects until applied to an Interval or Sequence. A Gluon also defines a handle for invoking a sequence within a service.
Artifact	An Artifact is the thing that occurs during an Interval. The contents of the Artifact are not specified in WS-Calendar, rather the Artifact provides an extension base for the use of WS-Calendar in other specifications. EMIX product and performance Artifacts may inherit elements as do Intervals within a Sequence.

1103 Much of EMIX defines the payloads that are delivered in the artifact. WS-Calendar defines how schedule-
1104 related information, although incomplete in an Interval and Sequence can be modified and completed.
1105 WS-Calendar calls this process Inheritance and specifies a number of rules that govern Inheritance.
1106 EMIX artifacts define Inheritance in manner compliant with WS-Calendar. Table C-16-2 defines the terms
1107 used to describe inheritance.

1108 *Table C-16-2: WS-Calendar Semantics of Inheritance*

Term	Definition
Parent	A Gluon that points to a sequence is known as the sequence's Parent. A Gluon may alternately reference another Gluon, i.e., it is that other Gluon's Parent.
Lineage	Lineage refers to the full ordered set of Parents of a Sequence

Term	Definition
Inheritance	Parents bequeath information to Children that inherit them. If a child does not already possess that information, then it accepts the inheritance. Information specified in one informational object is considered present in another that is itself lacking expression of that information. This information is termed the Inheritance of that object.
Bequeath	A Parent Bequeaths attributes (Inheritance) to its Children
Inherit	A Child Inherits attributes (Inheritance) from its Parent
Availability	Availability expresses the range of times in which an Interval or Sequence can be Scheduled. Availability can overlay or be overlaid by Busy. Availability can be Inherited
Busy	Busy expresses the range of times in which an Interval or Sequence cannot be Scheduled. Busy can overlay or be overlaid by Availability. Busy can be Inherited

1109 As Intervals are processed, as Intervals are assembled, and as inheritance is processed, the information
1110 conveyed about each element changes. EMIX artifacts may pass through several stages in which the
1111 information is not yet complete or actionable, but is still a conforming expression of time and Sequence.
1112 Table 16-3 defines the terms used when discussing the processing or processability of Intervals and
1113 Sequences.

1114 *Table 16-3: WS-Calendar Semantics of Information Processing*

Term	Definition
Bound	As in mathematical logic where a metasyntactic variable is called "bound", an Interval, Sequence, or Partition is said to be Bound when the values necessary to execute it (as a service) are completely filled in.
Partially Bound	A Partially Bound Interval is one that is still not Bound after receiving its Inheritance. A Sequences or Partitions is Partially Bound if it contains at least one Interval that is Partially Bound.
Unbound	An Unbound Interval or Sequence is not itself complete, but must still receive inheritance to be fully specified. A Sequences or Partitions is Unbound if it contains at least one Interval that is Unbound.
Fully Bound	A synonym for Bound
Scheduled	A Sequence or Partition is said to be Scheduled when it is Anchored, Fully Bound, and service performance has been requested.
Unscheduled	An Interval is Unscheduled if its neither its begin date and time nor its end date and time have been set. A Sequence or Partition is Unscheduled if none of its Intervals, after when Fully Bound, is Scheduled.
Designated Interval	In a Sequence the Designated `Interval` is either (a) (if there are no Gluons related to the Sequence) one of the Earliest Interval(s), or (b) (if there is at least one Gluon related to the Sequence) the single Interval referenced by a Gluon as Parent.
Composed Interval	A Composed Interval is the virtual Interval specified by applying inheritance through the entire lineage and into the Sequence in accord with the inheritance rules. A Composed Interval may be Bound or Unbound.

Term	Definition
Composed Sequence	A Composed Sequence is the virtual Sequence specified by applying inheritance through the entire lineage and into the Sequence in accord with the inheritance rules. A Composed Sequence may be Bound or Unbound.

1115 The WS-Calendar defines more terms, and in greater detail, but the tables above are sufficient to be able
1116 to discuss schedule, sequence, and inheritance in EMIX.

1117

1118 D. Electrical Power and Energy

1119 Each type of Electrical Power and Energy Product has its own definitions and its own descriptive
1120 parameters. These artifacts are the specific descriptions relevant to defining the potential utility of the
1121 power and energy Product. The Power and Energy Artifacts describe the intrinsic information. There may
1122 be cases when an Artifact is held in the envelope contents, perhaps as informational support for the
1123 intrinsic prices.

1124 To put the terms "Power" and "Energy" into the proper context for this specification, the following
1125 definitions will be used:

1126 • Apparent Energy: the production or consumption of Apparent Power over time; unit: volt-ampere
1127   hours, VAh

1128 • Apparent Power (S): mathematical product of root-mean-square voltage and root-mean-square
1129   current, vector sum of Real Power and Reactive Power, square root of sum of squares of Real
1130   Power and Reactive Power; unit: volt-ampere, VA

1131 • Current: flow of electric charge, or rate of flow of electric charge; unit: ampere, A

1132 • Energy: the production or consumption of Power over time.

1133 • Power Factor (p.f.): ratio of Real Power to Complex Power, cosine of the phase angle between
1134   Current and Voltage, expressed as a number between 0 and 1, expressed as a percentage (i.e.,
1135   50% = 0.5); unit: dimensionless

1136 • Reactive Energy: the production or consumption of Reactive Power over time; unit: volt-ampere-
1137   reactive hours, VARh, VA-rh, varh

1138 • Reactive Power (Q): mathematical product of the root-mean-square voltage and root-mean-
1139   square current multiplied by the sine of the angle between the voltage and current; unit: volt-
1140   amperes reactive, VAR, VA-r, var

1141 • Real Energy: the production or consumption of Real Power over time; unit: Watt-hour, Wh

1142 • Real Power (P): rate at which electricity is produced or consumed, mathematical product of
1143   Voltage and Current; unit: Watt, W

1144 • Voltage: difference in electric potential between two points; unit: volt, V

1145 • Generically, the use of the term "Power" refers to "Real Power" and is expressed in Watts.
1146   Otherwise, one talks of Apparent Power in VA, or Reactive Power in VARs. Generically, the use
1147   of the term "Energy" refers to "Real Energy" and is expressed in Watt-hours. Otherwise, on talks
1148   of Apparent Energy in VAh, or Reactive Energy in VARh.

1149 Generically, the use of the term "Power" refers to "Real Power" and is expressed in Watts. Otherwise, one
1150 talks of Apparent Power or Complex Power in VA, or Reactive Power in VARs.

E. Mapping between NAESB PAP03 work and this specification

Under the **[NIST]**-led smart grid interoperability process, the North American Energy Standards Board (NAESB) provided a minimal scope and requirements for this specification, specifically in its work to address the Priority Action Plan 03 (PA03), Price and Product definition. This section maps the specific requirements from NAESB to the work in this specification.

Tariff Rate Type	Description
block rate	In Power-Contracts.xsd, addressed by the Block Power Full Requirements Contract.
critical peak price	Addressed by both Price Relative and Price Multiplier when applies to a business schedule.
demand rate	Demand charges can be applied to all Product types in EMIX.
day ahead market rate	Either TEMIX or a Block Power agreement applied to a day-ahead schedule addresses this need.
market clearing price for energy	TEMIX addresses this use case directly.
peak time rebate	
real time price rate	Either TEMIX or a Block Power agreement applied to a day-ahead schedule addresses this need.
time of use rate	Either TEMIX or a Block Power agreement applied to a day-ahead schedule addresses this need. EMIX applied alongside any of the standard agreements can support variable peak pricing.
variable peak pricing	TEMIX applied alongside any of the standard products can support variable peak pricing.

1160 F. Schemas (Non-Normative)

1161 In OASIS, when there are external, machine-readable artifacts, they are always normative. These are
1162 placed here as a convenience to the reviewer.

1163 If you are tracing inheritance, and the construction of EMIX information through the schemas, recall that
1164 every EMIX Product is derived from EMIX Base which is a Business Schedule applied to a Product
1165 Description. All transactions occur at the EMIX Interface. Products are described and enumerated using
1166 extensions of the Item Base.

1167 F.1 EMIX Schemas

1168 The EMIX Schema is in three parts

1169 F.1.1 EMIX.XSD

```
1170    <?xml version="1.0" encoding="UTF-8"?>
1171    <!-- emix.xsd
1172    Schema Set for OASIS EMIX 1.0 WD23 (20110411)
1173    Set includes:
1174        EMIX, EMIX-Requirements, EMIX-Warrants (emix)
1175        Power, Power-Contracts, Power-Quality (power)
1176        Resource (resource)
1177
1178    This set built on the WS-Calendar v1.0 PRD02 Schemas.
1179    -->
1180    <!-- 1.0 EMIX: Energy Market Information Exchange-->
1181    <xs:schema xmlns:emix="http://docs.oasis-open.org/ns/emix"
1182    xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
1183    xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
1184    xmlns:gml="http://www.opengis.net/gml/3.2"
1185    xmlns:gmlsf="http://www.opengis.net/gmlsf/2.0"
1186    xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://docs.oasis-
1187    open.org/ns/emix" elementFormDefault="qualified" attributeFormDefault="unqualified">
1188        <xs:include schemaLocation="emix-requirements.xsd"/>
1189        <xs:include schemaLocation="emix-warrants.xsd"/>
1190        <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
1191    schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
1192    spec/v1.0/csprd02/xsd/iCalendar.xsd"/>
1193        <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
1194    schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
1195    spec/v1.0/csprd02/xsd/iCalendar-wscal-extensions.xsd"/>
1196        <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
1197    schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
1198    spec/v1.0/csprd02/xsd/iCalendar-availability-extension.xsd"/>
1199        <xs:import namespace="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
1200    schemaLocation="http://www.unece.org/uncefact/codelist/standard/ISO_ISO3AlphaCurrencyCod
1201    e_20100407.xsd"/>
1202        <xs:import namespace="http://www.opengis.net/gml/3.2"
1203    schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
1204        <!-- 1.0 Core EMIX objects-->
1205        <xs:annotation>
1206            <xs:appinfo
1207    source="http://schemas.opengis.net/gml/3.2.1/profiles/gmlsfProfile/2.0/gmlsfLevels.xsd">
1208                <gmlsf:ComplianceLevel>0</gmlsf:ComplianceLevel>
1209
1210        <gmlsf:GMLProfileSchema>http://schemas.opengis.net/gml/3.2.1/profiles/gmlsfProfile/2.
1211    0/gmlsf.xsd</gmlsf:GMLProfileSchema>
1212            </xs:appinfo>
1213        </xs:annotation>
1214        <!-- 1.1 EMIX Product -->
1215        <xs:element name="product" type="emix:ProductType">
1216            <xs:annotation>
1217                <xs:documentation>Emix Product, .i.e., a Product Description applied
1218    to a schedule.</xs:documentation>
```

```
1219                        </xs:annotation>
1220              </xs:element>
1221              <xs:complexType name="ProductType" mixed="false">
1222                    <xs:annotation>
1223                          <xs:documentation>EMIX Product Type, i.e. a Product Description
1224         applied to a Schedule</xs:documentation>
1225                    </xs:annotation>
1226                    <xs:complexContent mixed="false">
1227                          <xs:extension base="emix:EmixBaseType">
1228                                <xs:sequence>
1229                                      <xs:element ref="emix:currency"/>
1230                                      <xs:element ref="emix:marketContext"/>
1231                                      <xs:element ref="emix:transactiveState"/>
1232                                      <xs:element ref="emix:side"/>
1233                                      <xs:element ref="emix:constraints"/>
1234                                </xs:sequence>
1235                          </xs:extension>
1236                    </xs:complexContent>
1237              </xs:complexType>
1238              <!-- 1.2 EMIX Option -->
1239              <xs:element name="emixOption" type="emix:EmixOptionType">
1240                    <xs:annotation>
1241                          <xs:documentation>Option to buy an Emix Product</xs:documentation>
1242                    </xs:annotation>
1243              </xs:element>
1244              <xs:complexType name="EmixOptionType" mixed="false">
1245                    <xs:complexContent mixed="false">
1246                          <xs:extension base="emix:EmixBaseType">
1247                                <xs:sequence>
1248                                      <xs:element ref="emix:transactiveState"/>
1249                                      <xs:element name="optionExerciseSchedule"
1250         type="emix:BusinessScheduleType"/>
1251                                      <xs:element name="optionHolderSide"
1252         type="emix:SideType"/>
1253                                      <xs:element name="optionPremium"
1254         type="emix:PriceType"/>
1255                                      <xs:element name="optionExerciseLeadTime"
1256         type="xcal:DurationPropType"/>
1257                                      <xs:element name="optionStrikePrice"
1258         type="emix:PriceType"/>
1259                                      <xs:element ref="emix:optionType"/>
1260                                      <xs:element ref="emix:side"/>
1261                                      <xs:element ref="emix:marketContext"/>
1262                                      <xs:element ref="emix:currency"/>
1263                                      <xs:element ref="emix:constraints"/>
1264                                </xs:sequence>
1265                          </xs:extension>
1266                    </xs:complexContent>
1267              </xs:complexType>
1268              <!-- 1.3 EMIX TEMIX -->
1269              <xs:element name="temix" type="emix:TemixType">
1270                    <xs:annotation>
1271                          <xs:documentation>minimalist Energy Market Information Exchange (EMIX)
1272         Type</xs:documentation>
1273                    </xs:annotation>
1274              </xs:element>
1275              <xs:complexType name="TemixType" mixed="false">
1276                    <xs:complexContent mixed="false">
1277                          <xs:extension base="emix:EmixBaseType">
1278                                <xs:sequence>
1279                                      <xs:element ref="emix:currency"/>
1280                                      <xs:element ref="emix:transactiveState"/>
1281                                      <xs:element ref="emix:constraints"/>
1282                                      <xs:element ref="emix:side"/>
1283                                      <xs:element name="expires" type="xs:dateTime"
1284         minOccurs="0" maxOccurs="1"/>
1285                                </xs:sequence>
1286                          </xs:extension>
1287                    </xs:complexContent>
1288              </xs:complexType>
1289              <!-- 1.4 Delivery -->
```

```xml
          <xs:element name="delivery" type="emix:DeliveryType"/>
          <xs:complexType name="DeliveryType" mixed="false">
                <xs:annotation>
                        <xs:documentation>Receipt / Report of Product Delivery. Injection flag
       is true for adding product to market supply, false for taking from
       market.</xs:documentation>
                </xs:annotation>
                <xs:complexContent mixed="false">
                        <xs:extension base="emix:EmixBaseType">
                                <xs:sequence>
                                        <xs:element name="injection" type="xs:boolean"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
          </xs:complexType>
      <!-- 2.0 EMIX Components-->
      <!-- 2.1 Envelope -->
      <xs:element name="envelopeContents" type="emix:EnvelopeContentsType"/>
      <xs:complexType name="EnvelopeContentsType">
                <xs:sequence>
                        <xs:element ref="emix:warrants" minOccurs="0" maxOccurs="1"/>
                </xs:sequence>
      </xs:complexType>
      <!-- 8.0 Supporting Information Structures-->
      <!-- 8.2 Market defintions -->
      <!-- 8.2.1 Market Context -->
      <xs:element name="marketContext" type="emix:MarketContextType"/>
      <xs:simpleType name="MarketContextType">
                <xs:restriction base="xs:anyURI"/>
      </xs:simpleType>
      <!-- 8.2.2 Transactive State -->
      <xs:element name="transactiveState" type="emix:TransactiveStateType"/>
      <xs:simpleType name="TransactiveStateType">
                <xs:restriction base="xs:string">
                        <xs:enumeration value="IndicationOfInterest"/>
                        <xs:enumeration value="Tender"/>
                        <xs:enumeration value="Transaction"/>
                        <xs:enumeration value="Exercise"/>
                        <xs:enumeration value="Delivery"/>
                        <xs:enumeration value="TransportCommitment"/>
                        <xs:enumeration value="Publication"/>
                </xs:restriction>
      </xs:simpleType>
      <!-- 8.2.3 Currency -->
      <xs:element name="currency" type="clm5ISO42173A:ISO3AlphaCurrencyCodeContentType">
                <xs:annotation>
                        <xs:documentation>Currency codes coming from UN CEFACT
       schemas</xs:documentation>
                </xs:annotation>
      </xs:element>
      <!-- 8.2.4 Enumeration for Side -->
      <xs:element name="side" type="emix:SideType"/>
      <xs:simpleType name="SideType">
                <xs:restriction base="xs:string">
                        <xs:enumeration value="Buy"/>
                        <xs:enumeration value="Sell"/>
                </xs:restriction>
      </xs:simpleType>
      <!-- 8.3 Price -->
      <xs:element name="priceBase" type="emix:PriceBaseType" abstract="true">
                <xs:annotation>
                        <xs:documentation>Abstract base for EMIX Prices</xs:documentation>
                </xs:annotation>
      </xs:element>
      <xs:complexType name="PriceBaseType" abstract="true">
                <xs:annotation>
                        <xs:documentation>Type of Abstract base for EMIX
       Prices</xs:documentation>
                </xs:annotation>
      </xs:complexType>
      <!-- 8.3.1 Absolute Price  -->
```

```
1361          <xs:element name="price" type="emix:PriceType" substitutionGroup="emix:priceBase"/>
1362          <xs:complexType name="PriceType" mixed="false">
1363                  <xs:annotation>
1364                          <xs:documentation>Simple Price</xs:documentation>
1365                  </xs:annotation>
1366                  <xs:complexContent mixed="false">
1367                          <xs:extension base="emix:PriceBaseType">
1368                                  <xs:sequence>
1369                                          <xs:element ref="emix:value" minOccurs="1"
1370   maxOccurs="1"/>
1371                                  </xs:sequence>
1372                          </xs:extension>
1373                  </xs:complexContent>
1374          </xs:complexType>
1375          <!-- 8.3.2 Multiplier Price - multiplier on base amount  -->
1376          <xs:element name="priceMultiplier" type="emix:PriceMultiplierType"
1377   substitutionGroup="emix:priceBase"/>
1378          <xs:complexType name="PriceMultiplierType" mixed="false">
1379                  <xs:annotation>
1380                          <xs:documentation>Multiplier times market price, 1 for same as
1381   market</xs:documentation>
1382                  </xs:annotation>
1383                  <xs:complexContent mixed="false">
1384                          <xs:extension base="emix:PriceBaseType">
1385                                  <xs:sequence>
1386                                          <xs:element name="multiplier" type="xs:float"
1387   minOccurs="1" maxOccurs="1"/>
1388                                          <xs:element ref="emix:marketContext" minOccurs="0"
1389   maxOccurs="1">
1390                                                  <xs:annotation>
1391                                                          <xs:documentation>Market Context for
1392   base price. If blank, Market Context from hosting artifact.</xs:documentation>
1393                                                  </xs:annotation>
1394                                          </xs:element>
1395                                  </xs:sequence>
1396                          </xs:extension>
1397                  </xs:complexContent>
1398          </xs:complexType>
1399          <!-- 8.3.4 Price Offset (additive or subtractive) over base amount  -->
1400          <xs:element name="priceRelative" type="emix:PriceRelativeType"
1401   substitutionGroup="emix:priceBase"/>
1402          <xs:complexType name="PriceRelativeType" mixed="false">
1403                  <xs:annotation>
1404                          <xs:documentation>Price Relative is a fixed charge (positive or
1405   negative) apllied to base price</xs:documentation>
1406                  </xs:annotation>
1407                  <xs:complexContent mixed="false">
1408                          <xs:extension base="emix:PriceBaseType">
1409                                  <xs:sequence>
1410                                          <xs:element ref="emix:value" minOccurs="1"
1411   maxOccurs="1"/>
1412                                          <xs:element ref="emix:marketContext" minOccurs="0"
1413   maxOccurs="1">
1414                                                  <xs:annotation>
1415                                                          <xs:documentation>Market Context for
1416   base price. If blank, Market Context from hosting artifact.</xs:documentation>
1417                                                  </xs:annotation>
1418                                          </xs:element>
1419                                  </xs:sequence>
1420                          </xs:extension>
1421                  </xs:complexContent>
1422          </xs:complexType>
1423          <!-- 8.3.6 Simple Price  -->
1424          <xs:element name="value" type="emix:valueType"/>
1425          <xs:simpleType name="valueType">
1426                  <xs:restriction base="xs:decimal"/>
1427          </xs:simpleType>
1428          <!-- 8.5 Quantity -->
1429          <xs:element name="integralOnly" type="emix:IntegralOnlyType"/>
1430          <xs:simpleType name="IntegralOnlyType">
1431                  <xs:annotation>
```

```
1432                        <xs:documentation>integralOnly is an element used in many EMIX objects
1433    distinguishing between an (amount, response, ramp) that is all (true) or nothing
1434    (false)</xs:documentation>
1435                    </xs:annotation>
1436                    <xs:restriction base="xs:boolean"/>
1437        </xs:simpleType>
1438        <xs:element name="autonomous" type="emix:AutonomousType"/>
1439        <xs:simpleType name="AutonomousType">
1440                    <xs:annotation>
1441                        <xs:documentation>An autonomous resource or service (true) is able to
1442    respond or maintain service independently. A non autonomous service (false) must await
1443    dispatch.</xs:documentation>
1444                    </xs:annotation>
1445                    <xs:restriction base="xs:boolean"/>
1446        </xs:simpleType>
1447        <!-- 8.7 Enumeration for Option Types -->
1448        <xs:element name="optionType" type="emix:OptionTypeType"/>
1449        <xs:simpleType name="OptionTypeType">
1450                    <xs:union memberTypes="emix:OptionTypeEnumeratedType
1451    emix:EmixExtensionType"/>
1452        </xs:simpleType>
1453        <xs:simpleType name="OptionTypeEnumeratedType">
1454                    <xs:annotation>
1455                        <xs:documentation>Enumerated Option Types</xs:documentation>
1456                    </xs:annotation>
1457                    <xs:restriction base="xs:string"/>
1458        </xs:simpleType>
1459        <!-- 8.8 Performance Constraints -->
1460        <!-- 9.2 Abstract EMIX Base(product applied to a schedule) -->
1461        <xs:element name="emixBase" type="emix:EmixBaseType"/>
1462        <xs:complexType name="EmixBaseType" abstract="true">
1463                    <xs:annotation>
1464                        <xs:documentation>iCalendar-derived object to host EMIX
1465    elements</xs:documentation>
1466                    </xs:annotation>
1467                    <xs:complexContent>
1468                        <xs:extension base="xcal:VcalendarType">
1469                            <xs:sequence>
1470                                <xs:element ref="emix:envelopeContents" minOccurs="0"
1471    maxOccurs="1"/>
1472                            </xs:sequence>
1473                        </xs:extension>
1474                    </xs:complexContent>
1475        </xs:complexType>
1476        <!-- 9.3 Abstract Product  Description -->
1477        <xs:element name="productDescription" type="emix:ProductDescriptionType"
1478    substitutionGroup="xcal:artifactBase"/>
1479        <xs:complexType name="ProductDescriptionType" abstract="true">
1480                    <xs:annotation>
1481                        <xs:documentation>In EMIX, the Product Description is placed in the
1482    Interval or Gluon attachment. The respective product schemas extend this abstract
1483    class.</xs:documentation>
1484                    </xs:annotation>
1485                    <xs:complexContent>
1486                        <xs:extension base="xcal:ArtifactBaseType"/>
1487                    </xs:complexContent>
1488        </xs:complexType>
1489        <!-- 9.4 Interfaces  -->
1490        <xs:element name="serviceArea" type="emix:ServiceAreaType"
1491    substitutionGroup="emix:emixInterface"/>
1492        <xs:complexType name="ServiceAreaType">
1493                    <xs:annotation>
1494                        <xs:documentation>The Service Area is the geographic region that is
1495    affected by the EMIX market condition</xs:documentation>
1496                    </xs:annotation>
1497                    <xs:complexContent>
1498                        <xs:extension base="emix:EmixInterfaceType">
1499                            <xs:sequence>
1500                                <xs:element ref="emix:geographicArea"/>
1501                            </xs:sequence>
1502                        </xs:extension>
```

```
1503                    </xs:complexContent>
1504            </xs:complexType>
1505            <xs:element name="emixInterface" type="emix:EmixInterfaceType"/>
1506            <xs:complexType name="EmixInterfaceType" abstract="true" mixed="false">
1507                    <xs:annotation>
1508                            <xs:documentation>Abstract base class for the interfaces for EMIX
1509    Product delivery, measurement, and/or pricing</xs:documentation>
1510                    </xs:annotation>
1511            </xs:complexType>
1512            <!-- 9.5 Geographic Area -->
1513            <xs:element name="geographicArea" type="emix:GeographicAreaType"
1514    substitutionGroup="gml:AbstractFeature"/>
1515            <xs:complexType name="GeographicAreaType" mixed="false">
1516                    <xs:annotation>
1517                            <xs:documentation>A service area is a geographic region that may be
1518    affected by the same EMIX market condition.</xs:documentation>
1519                    </xs:annotation>
1520                    <xs:complexContent mixed="false">
1521                            <xs:extension base="gml:AbstractFeatureType">
1522                                    <xs:sequence>
1523                                            <xs:element name="foo"
1524    type="gml:GeometryPropertyType"/>
1525                                    </xs:sequence>
1526                            </xs:extension>
1527                    </xs:complexContent>
1528            </xs:complexType>
1529            <!-- 9.6 Business Schedule  -->
1530            <xs:element name="businessSchedule" type="emix:BusinessScheduleType"/>
1531            <xs:complexType name="BusinessScheduleType" mixed="false">
1532                    <xs:annotation>
1533                            <xs:documentation>iCalendar-derived business schedule, more variant
1534    than allowed in sequences</xs:documentation>
1535                    </xs:annotation>
1536                    <xs:complexContent mixed="false">
1537                            <xs:extension base="xcal:VavailabilityType"/>
1538                    </xs:complexContent>
1539            </xs:complexType>
1540            <xs:element name="duration" type="emix:DurationType"/>
1541            <xs:complexType name="DurationType" mixed="false">
1542                    <xs:annotation>
1543                            <xs:documentation>iCalendar-derived duration. THis brings in
1544    xcal:duration and xcal:parameters. </xs:documentation>
1545                    </xs:annotation>
1546                    <xs:complexContent mixed="false">
1547                            <xs:extension base="xcal:DurationPropType"/>
1548                    </xs:complexContent>
1549            </xs:complexType>
1550            <!-- 9.7 emix Items -->
1551            <xs:element name="measurement" type="emix:MeasurementType"
1552    substitutionGroup="emix:productDescription"/>
1553            <xs:complexType name="MeasurementType">
1554                    <xs:annotation>
1555                            <xs:documentation>Type of Measurement</xs:documentation>
1556                    </xs:annotation>
1557                    <xs:complexContent>
1558                            <xs:extension base="emix:ProductDescriptionType">
1559                                    <xs:sequence>
1560                                            <xs:element ref="emix:quantity"/>
1561                                            <xs:element ref="emix:itemBase"/>
1562                                    </xs:sequence>
1563                            </xs:extension>
1564                    </xs:complexContent>
1565            </xs:complexType>
1566            <xs:element name="emixGranularity" type="emix:EmixGranularityType"/>
1567            <xs:complexType name="EmixGranularityType" mixed="false">
1568                    <xs:annotation>
1569                            <xs:documentation>Abstract base class used for graularity of market
1570    indications of interest and tenders</xs:documentation>
1571                    </xs:annotation>
1572                    <xs:sequence>
1573                            <xs:element ref="emix:quantity"/>
```

```
1574                                        <xs:element ref="emix:itemBase"/>
1575                                </xs:sequence>
1576                        </xs:complexType>
1577                        <xs:element name="itemBase" type="emix:ItemBaseType" abstract="true"/>
1578                        <xs:complexType name="ItemBaseType" abstract="true" mixed="false">
1579                                <xs:annotation>
1580                                        <xs:documentation>Abstract base class for units for EMIX Product
1581            delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item,
1582            Lading Item. Item does not include Quantity or Price, because a single product
1583            description or transaction may have multiple quantities or prices associated with a
1584            single item.</xs:documentation>
1585                                </xs:annotation>
1586                        </xs:complexType>
1587                        <!-- 9.8 Units and Measurement Abstractions -->
1588                        <xs:element name="quantity" type="emix:QuantityType"/>
1589                        <xs:simpleType name="QuantityType">
1590                                <xs:annotation>
1591                                        <xs:documentation>Base type for all quanties in
1592            EMIX.</xs:documentation>
1593                                </xs:annotation>
1594                                <xs:restriction base="xs:float"/>
1595                        </xs:simpleType>
1596                        <xs:element name="scale" type="emix:SiScaleType"/>
1597                        <xs:simpleType name="SiScaleType">
1598                                <xs:annotation>
1599                                        <xs:documentation>Scale based on representations of SI scale as
1600            expressed in the unit multipliers defined for the CIM</xs:documentation>
1601                                        <xs:documentation xml:lang="en">enumeration</xs:documentation>
1602                                </xs:annotation>
1603                                <xs:restriction base="xs:string">
1604                                        <xs:enumeration value="n">
1605                                                <xs:annotation>
1606                                                        <xs:documentation>Nano 10**-9</xs:documentation>
1607                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1608                                                </xs:annotation>
1609                                        </xs:enumeration>
1610                                        <xs:enumeration value="micro">
1611                                                <xs:annotation>
1612                                                        <xs:documentation>Micro 10**-6</xs:documentation>
1613                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1614                                                </xs:annotation>
1615                                        </xs:enumeration>
1616                                        <xs:enumeration value="m">
1617                                                <xs:annotation>
1618                                                        <xs:documentation>Milli 10**-3</xs:documentation>
1619                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1620                                                </xs:annotation>
1621                                        </xs:enumeration>
1622                                        <xs:enumeration value="c">
1623                                                <xs:annotation>
1624                                                        <xs:documentation>Centi 10**-2</xs:documentation>
1625                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1626                                                </xs:annotation>
1627                                        </xs:enumeration>
1628                                        <xs:enumeration value="d">
1629                                                <xs:annotation>
1630                                                        <xs:documentation>Deci 10**-1</xs:documentation>
1631                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1632                                                </xs:annotation>
1633                                        </xs:enumeration>
1634                                        <xs:enumeration value="k">
1635                                                <xs:annotation>
1636                                                        <xs:documentation>Kilo 10**3</xs:documentation>
1637                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1638                                                </xs:annotation>
1639                                        </xs:enumeration>
1640                                        <xs:enumeration value="M">
1641                                                <xs:annotation>
1642                                                        <xs:documentation>Mega 10**6</xs:documentation>
1643                                                        <xs:documentation xml:lang="en">enum</xs:documentation>
1644                                                </xs:annotation>
```

```
1645                            </xs:enumeration>
1646                            <xs:enumeration value="G">
1647                                    <xs:annotation>
1648                                            <xs:documentation>Giga 10**9</xs:documentation>
1649                                            <xs:documentation xml:lang="en">enum</xs:documentation>
1650                                    </xs:annotation>
1651                            </xs:enumeration>
1652                            <xs:enumeration value="T">
1653                                    <xs:annotation>
1654                                            <xs:documentation>Tera 10**12</xs:documentation>
1655                                            <xs:documentation xml:lang="en">enum</xs:documentation>
1656                                    </xs:annotation>
1657                            </xs:enumeration>
1658                            <xs:enumeration value="none">
1659                                    <xs:annotation>
1660                                            <xs:documentation xml:lang="en">enum</xs:documentation>
1661                                    </xs:annotation>
1662                            </xs:enumeration>
1663                            <xs:enumeration value="p">
1664                                    <xs:annotation>
1665                                            <xs:documentation>Pico 10**-12</xs:documentation>
1666                                            <xs:documentation xml:lang="en">enum</xs:documentation>
1667                                    </xs:annotation>
1668                            </xs:enumeration>
1669                    </xs:restriction>
1670            </xs:simpleType>
1671            <!-- 9.9 Extension Type  -->
1672            <xs:simpleType name="EmixExtensionType">
1673                    <xs:annotation>
1674                            <xs:documentation>Pattern used for extending string enumeration, where
1675    allowed</xs:documentation>
1676                    </xs:annotation>
1677                    <xs:restriction base="xs:string">
1678                            <xs:pattern value="x-\S.*"/>
1679                    </xs:restriction>
1680            </xs:simpleType>
1681    </xs:schema>
```

F.1.2 EMIX-Requirements

```
1683    <?xml version="1.0" encoding="UTF-8"?>
1684    <!-- emix.xsd
1685    Schema Set for OASIS EMIX 1.0 WD23 (20110411)
1686    Set includes:
1687        EMIX, EMIX-Requirements, EMIX-Warrants (emix)
1688        Power, Power-Contracts, Power-Quality (power)
1689        Resource (resource)
1690
1691    This set built on the WS-Calendar v1.0 PRD02 Schemas.
1692    -->
1693    <!-- 8.9 Constraints & Requirements -->
1694    <xs:schema xmlns:emix="http://docs.oasis-open.org/ns/emix"
1695    xmlns:xs="http://www.w3.org/2001/XMLSchema"
1696    xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
1697    xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
1698    xmlns:gml="http://www.opengis.net/gml/3.2" targetNamespace="http://docs.oasis-
1699    open.org/ns/emix" elementFormDefault="qualified" attributeFormDefault="unqualified">
1700        <xs:include schemaLocation="emix.xsd"/>
1701        <xs:element name="constraints" type="emix:ConstraintsType"/>
1702        <xs:complexType name="ConstraintsType">
1703                <xs:annotation>
1704                        <xs:documentation>Constraints are extrinsic to the product delivery
1705    but effect how a partner may request performance of a service. Performance constraints
1706    may be tied to the basic mechanical needs of the resource or to the business needs of
1707    the source. These constraints can affect the market value of the resource or the
1708    repeated invocation of a resource. It is possible for a given underlying resource to be
1709    offered to the market with different constraints and therefor different values. It is
1710    possible for a given underlying resource to be offered to the market with different
1711    constraints and therefor different values.</xs:documentation>
1712                </xs:annotation>
```

```
1713                    <xs:sequence>
1714                            <xs:element name="constraints" type="emix:ArrayOfConstraints"/>
1715                            <xs:element name="requirements" type="emix:ArrayOfRequirements"/>
1716                    </xs:sequence>
1717            </xs:complexType>
1718            <!-- 8.9.1 Core EMIX Constraints-->
1719            <xs:element name="baseConstraint" type="emix:BaseConstraintType" abstract="true">
1720                    <xs:annotation>
1721                            <xs:documentation>Abstract extension object for Emix
1722    Constraints</xs:documentation>
1723                    </xs:annotation>
1724            </xs:element>
1725            <xs:complexType name="ArrayOfConstraints">
1726                    <xs:annotation>
1727                            <xs:documentation>Collection of Emix Constraints</xs:documentation>
1728                    </xs:annotation>
1729                    <xs:sequence>
1730                            <xs:element ref="emix:baseConstraint" minOccurs="0"
1731    maxOccurs="unbounded"/>
1732                    </xs:sequence>
1733            </xs:complexType>
1734            <xs:complexType name="BaseConstraintType" abstract="true">
1735                    <xs:annotation>
1736                            <xs:documentation>Type of Abstract extension object for Emix
1737    Constraints</xs:documentation>
1738                    </xs:annotation>
1739            </xs:complexType>
1740            <xs:element name="minimumResponseDuration" type="emix:MinimumResponseDurationType"
1741    substitutionGroup="emix:baseConstraint">
1742                    <xs:annotation>
1743                            <xs:documentation>The shortest Duration for which the resource will
1744    accept a request  to maintain a response before returning to pre-request
1745    levels.</xs:documentation>
1746                    </xs:annotation>
1747            </xs:element>
1748            <xs:element name="maximumResponseDuration" type="emix:MaximumResponseDurationType"
1749    substitutionGroup="emix:baseConstraint">
1750                    <xs:annotation>
1751                            <xs:documentation>The longest Duration for which the resource will
1752    accept a request.</xs:documentation>
1753                    </xs:annotation>
1754            </xs:element>
1755            <xs:element name="minimumRecoveryDuration" type="emix:MinimumRecoveryDurationType"
1756    substitutionGroup="emix:baseConstraint">
1757                    <xs:annotation>
1758                            <xs:documentation>The minimum Duration that the Resource requires
1759    after the end of a response the resource has is ready to respond to a new
1760    request.</xs:documentation>
1761                    </xs:annotation>
1762            </xs:element>
1763            <xs:element name="minimumDurationBetweenInvocations"
1764    type="emix:MinimumDurationBetweenInvocationsType"
1765    substitutionGroup="emix:baseConstraint">
1766                    <xs:annotation>
1767                            <xs:documentation>The minimum Duration that the Resource requires
1768    after receiving a request before the resource has is ready to respond to a new
1769    request.</xs:documentation>
1770                    </xs:annotation>
1771            </xs:element>
1772            <xs:element name="minimumNotificationDuration"
1773    type="emix:MinimumNotificationDurationType" substitutionGroup="emix:baseConstraint">
1774                    <xs:annotation>
1775                            <xs:documentation>The minimum Duration that the Resource requires for
1776    Notification before initiating a response to a request.</xs:documentation>
1777                    </xs:annotation>
1778            </xs:element>
1779            <xs:element name="maximumNotificationDuration"
1780    type="emix:MaximumNotificationDurationType" substitutionGroup="emix:baseConstraint">
1781                    <xs:annotation>
1782                            <xs:documentation>The maximum Duration in advance of a requested
1783    response that the resource is willing to accept a request.</xs:documentation>
```

```
                    </xs:annotation>
            </xs:element>
            <xs:element name="responseTime" type="emix:ResponseTimeType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>Duration required from receipt of a request to
    initiation of a response by the resource</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="maximumInvocationsPerDuration"
    type="emix:MaximumInvocationsPerDurationType" substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>Maximum number of invocations of service during a
    given duration</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="maximumConsecutiveDurations"
    type="emix:MaximumConsecutiveDurationsType" substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>Maximim consecutive durations in which service can
    be invoked, e.g., it will not accept requests on more than 3 consecutive
    days.</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="minimumStartsPerDuration" type="emix:MinimumStartsPerDurationType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>The fewest Requests that the resource will accept
    during any duration. This constraint is typically used in market rather than in resource
    descriptions.</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="maximumRunDuration" type="emix:MaximumRunDurationType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>The Maximum duration for which a resource will
    accept a request</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="minimumRunDuration" type="emix:MinimumRunDurationType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>The Minimum duration for which a resource will
    accept a request</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="availabilitySchedule" type="emix:AvailabilityScheduleType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>A schedule of times for which which a resource will
    accept requests. The schedule may include multiple availability windows. The scheduled
    duration must be entirely within an availability window.</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="notificationSchedule" type="emix:NotificationScheduleType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>A schedule of time during which a resource will
    accept requests. The schedule may include multiple availability windows. THe
    notification must occur within an availability window.</xs:documentation>
                    </xs:annotation>
            </xs:element>
            <xs:element name="unavailabilitySchedule" type="emix:UnavailabilityScheduleType"
    substitutionGroup="emix:baseConstraint">
                    <xs:annotation>
                            <xs:documentation>A schedule of times for which which a resource will
    not accept requests. The schedule may include multiple unavailability windows. The
    scheduled duration must not occur within or overlap an unavailability
    window.</xs:documentation>
                    </xs:annotation>
            </xs:element>
```

```
1855            <!-- 8.9.1.1 Minimum Response-->
1856        <xs:complexType name="MinimumResponseDurationType" mixed="false">
1857              <xs:annotation>
1858                    <xs:documentation>Type of the shortest Duration for which the resource
1859    will accept a request  to maintain a response before returning to pre-request
1860    levels.</xs:documentation>
1861              </xs:annotation>
1862              <xs:complexContent mixed="false">
1863                    <xs:extension base="emix:BaseConstraintType">
1864                          <xs:sequence>
1865                                <xs:element ref="emix:duration" minOccurs="1"
1866    maxOccurs="1"/>
1867                          </xs:sequence>
1868                    </xs:extension>
1869              </xs:complexContent>
1870        </xs:complexType>
1871        <!-- 8.9.1.2 Maximum Response-->
1872        <xs:complexType name="MaximumResponseDurationType" mixed="false">
1873              <xs:annotation>
1874                    <xs:documentation>Type of the longest Duration for which the resource
1875    will accept a request.</xs:documentation>
1876              </xs:annotation>
1877              <xs:complexContent mixed="false">
1878                    <xs:extension base="emix:BaseConstraintType">
1879                          <xs:sequence>
1880                                <xs:element ref="emix:duration" minOccurs="1"
1881    maxOccurs="1"/>
1882                          </xs:sequence>
1883                    </xs:extension>
1884              </xs:complexContent>
1885        </xs:complexType>
1886        <xs:complexType name="MinimumRecoveryDurationType" mixed="false">
1887              <xs:annotation>
1888                    <xs:documentation>Type of the minimum Duration that the Resource
1889    requires after the end of a response the resource has is ready to respond to a new
1890    request.</xs:documentation>
1891              </xs:annotation>
1892              <xs:complexContent mixed="false">
1893                    <xs:extension base="emix:BaseConstraintType">
1894                          <xs:sequence>
1895                                <xs:element ref="emix:duration" minOccurs="1"
1896    maxOccurs="1"/>
1897                          </xs:sequence>
1898                    </xs:extension>
1899              </xs:complexContent>
1900        </xs:complexType>
1901        <xs:complexType name="MinimumDurationBetweenInvocationsType" mixed="false">
1902              <xs:annotation>
1903                    <xs:documentation>Type of the minimum Duration that the Resource
1904    requires after receiving a request before the resource has is ready to respond to a new
1905    request.</xs:documentation>
1906              </xs:annotation>
1907              <xs:complexContent mixed="false">
1908                    <xs:extension base="emix:BaseConstraintType">
1909                          <xs:sequence>
1910                                <xs:element ref="emix:duration" minOccurs="1"
1911    maxOccurs="1"/>
1912                          </xs:sequence>
1913                    </xs:extension>
1914              </xs:complexContent>
1915        </xs:complexType>
1916        <xs:complexType name="MinimumNotificationDurationType" mixed="false">
1917              <xs:annotation>
1918                    <xs:documentation>Type of the minimum Duration that the Resource
1919    requires for Notification before initiating a response to a request.</xs:documentation>
1920              </xs:annotation>
1921              <xs:complexContent mixed="false">
1922                    <xs:extension base="emix:BaseConstraintType">
1923                          <xs:sequence>
1924                                <xs:element ref="emix:duration" minOccurs="1"
1925    maxOccurs="1"/>
```

```
                                        </xs:sequence>
                                </xs:extension>
                        </xs:complexContent>
                </xs:complexType>
                <xs:complexType name="MaximumNotificationDurationType" mixed="false">
                        <xs:annotation>
                                <xs:documentation>Type of the maximum Duration in advance that a
Resource is willing to accept a request for a response.</xs:documentation>
                        </xs:annotation>
                        <xs:complexContent mixed="false">
                                <xs:extension base="emix:BaseConstraintType">
                                        <xs:sequence>
                                                <xs:element ref="emix:duration" minOccurs="1"
maxOccurs="1"/>
                                        </xs:sequence>
                                </xs:extension>
                        </xs:complexContent>
                </xs:complexType>
                <xs:complexType name="ResponseTimeType" mixed="false">
                        <xs:annotation>
                                <xs:documentation>Type of the Duration required from receipt of a
request to initiation of a response by the resource</xs:documentation>
                        </xs:annotation>
                        <xs:complexContent mixed="false">
                                <xs:extension base="emix:BaseConstraintType">
                                        <xs:sequence>
                                                <xs:element ref="emix:duration" minOccurs="1"
maxOccurs="1"/>
                                        </xs:sequence>
                                </xs:extension>
                        </xs:complexContent>
                </xs:complexType>
                <xs:complexType name="MaximumInvocationsPerDurationType" mixed="false">
                        <xs:annotation>
                                <xs:documentation>Type of the Maximum number of invocations of service
during a given duration</xs:documentation>
                        </xs:annotation>
                        <xs:complexContent mixed="false">
                                <xs:annotation>
                                        <xs:documentation>The resource will only accept a given number
of requests for performance during a given interval.</xs:documentation>
                                </xs:annotation>
                                <xs:extension base="emix:BaseConstraintType">
                                        <xs:sequence>
                                                <xs:element name="starts" type="xs:int" minOccurs="1"
maxOccurs="1"/>
                                                <xs:element ref="emix:duration" minOccurs="1"
maxOccurs="1"/>
                                        </xs:sequence>
                                </xs:extension>
                        </xs:complexContent>
                </xs:complexType>
                <xs:complexType name="MaximumConsecutiveDurationsType" mixed="false">
                        <xs:annotation>
                                <xs:documentation>Type of Maximim consecutive durations in which
service can be invoked, e.g., it will not accept requests on more than 3 consecutive
days.</xs:documentation>
                        </xs:annotation>
                        <xs:complexContent mixed="false">
                                <xs:extension base="emix:BaseConstraintType">
                                        <xs:sequence>
                                                <xs:element name="durations" type="xs:int"/>
                                                <xs:element ref="emix:duration" minOccurs="1"
maxOccurs="1"/>
                                        </xs:sequence>
                                </xs:extension>
                        </xs:complexContent>
                </xs:complexType>
                <xs:complexType name="MinimumStartsPerDurationType" mixed="false">
                        <xs:annotation>
```

```
1996                        <xs:documentation>Type of the fewest Requests that the resource will
1997        accept during any duration. This constraint is typically used in market rather than in
1998        resource descriptions.</xs:documentation>
1999                </xs:annotation>
2000                <xs:complexContent mixed="false">
2001                        <xs:extension base="emix:BaseConstraintType">
2002                                <xs:sequence>
2003                                        <xs:element name="starts" type="xs:int" minOccurs="1"
2004        maxOccurs="1"/>
2005                                        <xs:element ref="emix:duration" minOccurs="1"
2006        maxOccurs="1"/>
2007                                </xs:sequence>
2008                        </xs:extension>
2009                </xs:complexContent>
2010        </xs:complexType>
2011        <xs:complexType name="MaximumRunDurationType" mixed="false">
2012                <xs:annotation>
2013                        <xs:documentation>Type of the Maximum duration for which a resource
2014        will accept a request</xs:documentation>
2015                </xs:annotation>
2016                <xs:complexContent mixed="false">
2017                        <xs:extension base="emix:BaseConstraintType">
2018                                <xs:sequence>
2019                                        <xs:element ref="emix:duration" minOccurs="1"
2020        maxOccurs="1"/>
2021                                </xs:sequence>
2022                        </xs:extension>
2023                </xs:complexContent>
2024        </xs:complexType>
2025        <xs:complexType name="MinimumRunDurationType" mixed="false">
2026                <xs:annotation>
2027                        <xs:documentation>Type of the Minimum duration for which a resource
2028        will accept a request</xs:documentation>
2029                </xs:annotation>
2030                <xs:complexContent mixed="false">
2031                        <xs:extension base="emix:BaseConstraintType">
2032                                <xs:sequence>
2033                                        <xs:element ref="emix:duration" minOccurs="1"
2034        maxOccurs="1"/>
2035                                </xs:sequence>
2036                        </xs:extension>
2037                </xs:complexContent>
2038        </xs:complexType>
2039        <!-- Business Schedules -->
2040        <xs:complexType name="AvailabilityScheduleType" mixed="false">
2041                <xs:annotation>
2042                        <xs:documentation>Type of the schedule of time for which a resource
2043        will accept requests. The schedule may include multiple availability windows. The
2044        scheduled duration must be entirely within an availability window.</xs:documentation>
2045                </xs:annotation>
2046                <xs:complexContent mixed="false">
2047                        <xs:extension base="emix:BaseConstraintType">
2048                                <xs:sequence>
2049                                        <xs:element ref="emix:businessSchedule"
2050        maxOccurs="unbounded"/>
2051                                </xs:sequence>
2052                        </xs:extension>
2053                </xs:complexContent>
2054        </xs:complexType>
2055        <xs:complexType name="NotificationScheduleType" mixed="false">
2056                <xs:annotation>
2057                        <xs:documentation>Type of the schedule of time during which a resource
2058        will accept requests. The schedule may include multiple notofication windows. The
2059        request must occur within a notification window.</xs:documentation>
2060                </xs:annotation>
2061                <xs:complexContent mixed="false">
2062                        <xs:extension base="emix:BaseConstraintType">
2063                                <xs:sequence>
2064                                        <xs:element ref="emix:businessSchedule"
2065        maxOccurs="unbounded"/>
2066                                </xs:sequence>
```

```
2067                     </xs:extension>
2068                 </xs:complexContent>
2069     </xs:complexType>
2070     <xs:complexType name="UnavailabilityScheduleType" mixed="false">
2071             <xs:annotation>
2072                     <xs:documentation>Type of the schedule of time for which a resource
2073     will not accept requests. The schedule may include multiple unavailability windows. The
2074     scheduled duration must not occur within or overlap an unavailability
2075     window.</xs:documentation>
2076
2077             </xs:annotation>
2078             <xs:complexContent mixed="false">
2079                     <xs:extension base="emix:BaseConstraintType">
2080                             <xs:sequence>
2081                                     <xs:element ref="emix:businessSchedule"
2082     maxOccurs="unbounded"/>
2083                             </xs:sequence>
2084                     </xs:extension>
2085             </xs:complexContent>
2086     </xs:complexType>
2087     <!-- 8.2 EMIX Requirements-->
2088     <xs:element name="marketRequirements" type="emix:MarketRequirementsType">
2089             <xs:annotation>
2090                     <xs:documentation>Market Requirements are the market portion of
2091     Constraints, i.e., they are used to state the offeror's expectations about a tender.It
2092     is possible for a given underlying resource to be offered to the market with different
2093     Requirements and therefor different values.</xs:documentation>
2094             </xs:annotation>
2095     </xs:element>
2096     <xs:complexType name="MarketRequirementsType">
2097             <xs:annotation>
2098                     <xs:documentation>Market Requirements are the market portion of
2099     Constraints, i.e., they are used to state the offeror's expectations about a tender.It
2100     is possible for a given underlying resource to be offered to the market with different
2101     Requirements and therefor different values.</xs:documentation>
2102             </xs:annotation>
2103             <xs:sequence>
2104                     <xs:element name="requirements" type="emix:ArrayOfRequirements"/>
2105             </xs:sequence>
2106     </xs:complexType>
2107     <xs:complexType name="BaseRequirementType" abstract="true">
2108             <xs:annotation>
2109                     <xs:documentation>Abstract base for all
2110     Requirements</xs:documentation>
2111             </xs:annotation>
2112     </xs:complexType>
2113     <xs:element name="baseRequirement" type="emix:BaseRequirementType">
2114             <xs:annotation>
2115                     <xs:documentation>Abstract base for all
2116     Requirements</xs:documentation>
2117             </xs:annotation>
2118     </xs:element>
2119     <xs:complexType name="ArrayOfRequirements">
2120             <xs:annotation>
2121                     <xs:documentation>Abstract base for a collection of
2122     requirements</xs:documentation>
2123             </xs:annotation>
2124             <xs:sequence>
2125                     <xs:element ref="emix:baseRequirement" minOccurs="0"
2126     maxOccurs="unbounded"/>
2127             </xs:sequence>
2128     </xs:complexType>
2129     <xs:element name="minimumEconomicRequirement"
2130     type="emix:MinimumEconomicRequirementType" substitutionGroup="emix:baseRequirement">
2131             <xs:annotation>
2132                     <xs:documentation> Minimum net remuneration this resource requires
2133     from a total response</xs:documentation>
2134             </xs:annotation>
2135     </xs:element>
2136     <xs:element name="requiredStartupCost" type="emix:RequiredStartupCostType"
2137     substitutionGroup="emix:baseRequirement">
```

```
2138            <xs:annotation>
2139                    <xs:documentation>Minimum remunuration required from start-up of this
2140    service. </xs:documentation>
2141            </xs:annotation>
2142    </xs:element>
2143    <xs:element name="minimumResourceCost" type="emix:MinimumResourceCostType"
2144    substitutionGroup="emix:baseRequirement">
2145            <xs:annotation>
2146                    <xs:documentation>Resource requires this amount per period, i.e., a
2147    minimum requirement for $100 / hour at whatever rate</xs:documentation>
2148            </xs:annotation>
2149    </xs:element>
2150    <xs:complexType name="MinimumEconomicRequirementType" mixed="false">
2151            <xs:annotation>
2152                    <xs:documentation>Minimum net remuneration this resource requires from
2153    a total response</xs:documentation>
2154            </xs:annotation>
2155            <xs:complexContent mixed="false">
2156                    <xs:extension base="emix:BaseRequirementType">
2157                            <xs:sequence>
2158                                    <xs:element ref="emix:price"/>
2159                            </xs:sequence>
2160                    </xs:extension>
2161            </xs:complexContent>
2162    </xs:complexType>
2163    <xs:complexType name="RequiredStartupCostType" mixed="false">
2164            <xs:annotation>
2165                    <xs:documentation> Minimum remunuration required from start-up of this
2166    service. </xs:documentation>
2167            </xs:annotation>
2168            <xs:complexContent mixed="false">
2169                    <xs:extension base="emix:BaseRequirementType">
2170                            <xs:sequence>
2171                                    <xs:element ref="emix:price" minOccurs="1"
2172    maxOccurs="1"/>
2173                            </xs:sequence>
2174                    </xs:extension>
2175            </xs:complexContent>
2176    </xs:complexType>
2177    <!-- 4.3.2 Minimum Resource Cost  -->
2178    <xs:complexType name="MinimumResourceCostType" mixed="false">
2179            <xs:annotation>
2180                    <xs:documentation>Resource requires this amount per period, i.e., a
2181    minimum requirement for $100 / hour at whatever rate</xs:documentation>
2182            </xs:annotation>
2183            <xs:complexContent mixed="false">
2184                    <xs:extension base="emix:BaseRequirementType">
2185                            <xs:sequence>
2186                                    <xs:element ref="emix:price"/>
2187                                    <xs:element ref="emix:duration"/>
2188                            </xs:sequence>
2189                    </xs:extension>
2190            </xs:complexContent>
2191    </xs:complexType>
2192  </xs:schema>
```

F.1.3 EMIX Warrants

```
2194    <?xml version="1.0" encoding="UTF-8"?>
2195    <!-- emix.xsd
2196    Schema Set for OASIS EMIX 1.0 WD23 (20110411)
2197    Set includes:
2198        EMIX, EMIX-Requirements, EMIX-Warrants (emix)
2199        Power, Power-Contracts, Power-Quality (power)
2200        Resource (resource)
2201
2202    This set built on the WS-Calendar v1.0 PRD02 Schemas.
2203    -->
```

```
<xs:schema xmlns:emix="http://docs.oasis-open.org/ns/emix"
xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://docs.oasis-open.org/ns/emix" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <xs:include schemaLocation="emix.xsd"/>
    <!-- 8.8 EMIX Warrants-->
    <xs:element name="warrants" type="emix:ArrayOfWarrants">
            <xs:annotation>
                    <xs:documentation>Warrants are "a written assurances that some product
or service will be provided or will meet certain specifications" . Sellers use EMIX
Warrants to provide information about the source of the energy or about its
environmental characteristics. Buyers can use warrants to indicate what they wish to
purchase. EMIX does not define specific warrants, although it does define base types for
extension by those who wish to develop the various types of warrants.</xs:documentation>
            </xs:annotation>
    </xs:element>
    <xs:element name="baseWarrant" type="emix:BaseWarrantType">
            <xs:annotation>
                    <xs:documentation>Abstract extension object for Emix
Warrants</xs:documentation>
            </xs:annotation>
    </xs:element>
    <xs:complexType name="ArrayOfWarrants">
            <xs:annotation>
                    <xs:documentation>Collection of Emix Warrants</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                    <xs:element ref="emix:baseWarrant" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
    </xs:complexType>
    <xs:complexType name="BaseWarrantType" abstract="true">
            <xs:annotation>
                    <xs:documentation>Type of Abstract extension object for Emix
Warrants</xs:documentation>
            </xs:annotation>
    </xs:complexType>
    <xs:element name="supportForPrice" type="emix:SupportForPriceType"
substitutionGroup="emix:baseWarrant"/>
    <xs:element name="qualityWarrant" type="emix:QualityWarrantType"
substitutionGroup="emix:baseWarrant"/>
    <xs:element name="environmentalWarrant" type="emix:EnvironmentalWarrantType"
substitutionGroup="emix:baseWarrant"/>
    <xs:element name="sourceWarrant" type="emix:SourceWarrantType"
substitutionGroup="emix:baseWarrant"/>
    <xs:element name="contentWarrant" type="emix:ContentWarrantType"
substitutionGroup="emix:baseWarrant"/>
    <xs:element name="controllabilityWarrant" type="emix:ControllabilityWarrantType"
substitutionGroup="emix:baseWarrant"/>
    <!-- 8.8.1 Core EMIX Warrants-->
    <xs:complexType name="SupportForPriceType" abstract="true" mixed="false">
            <xs:annotation>
                    <xs:documentation>Price Support products may be needed to justify the
price. An example would be a transport product that support the difference between a
product price at a point of delivery and a product price at a point of
receipt.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                    <xs:extension base="emix:BaseWarrantType"/>
            </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="QualityWarrantType" abstract="true" mixed="false">
            <xs:annotation>
                    <xs:documentation>A Quality Warrant asserts or requires that the power
be of a certain quality or better.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                    <xs:extension base="emix:BaseWarrantType">
                            <xs:sequence>
```

```
2275                                    <xs:element ref="emix:product" minOccurs="1"
2276   maxOccurs="unbounded"/>
2277                            </xs:sequence>
2278                        </xs:extension>
2279                </xs:complexContent>
2280       </xs:complexType>
2281       <xs:complexType name="EnvironmentalWarrantType" abstract="true" mixed="false">
2282            <xs:annotation>
2283                <xs:documentation>An Environmental Warrant asserts what environmental
2284   cost was created by the product.</xs:documentation>
2285            </xs:annotation>
2286            <xs:complexContent mixed="false">
2287                    <xs:extension base="emix:BaseWarrantType">
2288                            <xs:sequence>
2289                                    <xs:element ref="emix:product" minOccurs="1"
2290   maxOccurs="unbounded"/>
2291                            </xs:sequence>
2292                    </xs:extension>
2293                </xs:complexContent>
2294       </xs:complexType>
2295       <xs:complexType name="SourceWarrantType" abstract="true" mixed="false">
2296            <xs:annotation>
2297                <xs:documentation>A source warrant consists of assertions about
2298   through what proces energy originated. </xs:documentation>
2299            </xs:annotation>
2300            <xs:complexContent mixed="false">
2301                    <xs:extension base="emix:BaseWarrantType">
2302                            <xs:sequence>
2303                                    <xs:element ref="emix:product" minOccurs="1"
2304   maxOccurs="unbounded"/>
2305                            </xs:sequence>
2306                    </xs:extension>
2307                </xs:complexContent>
2308       </xs:complexType>
2309       <xs:complexType name="ContentWarrantType" abstract="true" mixed="false">
2310            <xs:annotation>
2311                <xs:documentation>A content warrant consists of assertions about where
2312   energy originated. </xs:documentation>
2313            </xs:annotation>
2314            <xs:complexContent mixed="false">
2315                    <xs:extension base="emix:BaseWarrantType">
2316                            <xs:sequence>
2317                                    <xs:element ref="emix:product" minOccurs="1"
2318   maxOccurs="unbounded"/>
2319                            </xs:sequence>
2320                    </xs:extension>
2321                </xs:complexContent>
2322       </xs:complexType>
2323       <xs:complexType name="ControllabilityWarrantType" abstract="true" mixed="false">
2324            <xs:annotation>
2325                <xs:documentation>A Controllability Warrant makes certifies that the
2326   resource is controolable byt the market buyer. </xs:documentation>
2327            </xs:annotation>
2328            <xs:complexContent mixed="false">
2329                    <xs:extension base="emix:BaseWarrantType">
2330                            <xs:sequence>
2331                                    <xs:element ref="emix:product" minOccurs="1"
2332   maxOccurs="unbounded"/>
2333                            </xs:sequence>
2334                    </xs:extension>
2335                </xs:complexContent>
2336       </xs:complexType>
2337   </xs:schema>
```

F.2 Power Schemas

The Power Schema is in 3 parts

2340 F.2.1 Power.xsd

```
2341    <?xml version="1.0" encoding="UTF-8"?>
2342    <!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)
2343
2344    Set includes:
2345        EMIX, EMIX-Requirements, EMIX-Warrants (emix)
2346        Power, Power-Contracts, Power-Quality (power)
2347        Resource (resource)
2348
2349    This set built on the WS-Calendar v1.0 PRD02 Schemas.
2350    -->
2351    <xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
2352    xmlns:emix="http://docs.oasis-open.org/ns/emix"
2353    xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
2354    xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
2355    xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xs="http://www.w3.org/2001/XMLSchema"
2356    targetNamespace="http://docs.oasis-open.org/ns/emix/power"
2357    elementFormDefault="unqualified" attributeFormDefault="unqualified">
2358        <xs:include schemaLocation="power-products.xsd"/>
2359        <xs:include schemaLocation="power-quality.xsd"/>
2360        <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
2361        <!-- 1.0 Core EMIX Power objects-->
2362        <!-- 1.1 Power Product-->
2363        <!-- 1.2 Reserves and Power Options-->
2364        <!-- 2.0  Contract Power Products -->
2365        <!-- 4.0 Resource Semantics -->
2366        <!-- 6.0 Power Quality -->
2367        <!-- 9.1.2 Unit Energy Price -->
2368        <xs:element name="unitEnergyPrice" type="power:UnitEnergyPriceType"/>
2369        <xs:complexType name="UnitEnergyPriceType">
2370            <xs:annotation>
2371                <xs:documentation>Price per Unit of Energy, i.e., Power times
2372    Duration</xs:documentation>
2373            </xs:annotation>
2374            <xs:sequence>
2375                <xs:element ref="emix:priceBase"/>
2376                <xs:element ref="power:energyItem"/>
2377            </xs:sequence>
2378    </xs:complexType>
2379    <xs:element name="energyQuantity" type="power:EnergyQuantityType"/>
2380    <xs:complexType name="EnergyQuantityType">
2381            <xs:annotation>
2382                <xs:documentation>Level of Energy</xs:documentation>
2383            </xs:annotation>
2384            <xs:sequence>
2385                <xs:element ref="emix:quantity"/>
2386                <xs:element ref="power:energyItem"/>
2387            </xs:sequence>
2388    </xs:complexType>
2389    <!-- 9.1.3 Power Delivery Rate -->
2390    <xs:element name="powerQuantity" type="power:PowerQuantityType"/>
2391    <xs:complexType name="PowerQuantityType">
2392            <xs:annotation>
2393                <xs:documentation>Quantity of Power</xs:documentation>
2394            </xs:annotation>
2395            <xs:sequence>
2396                <xs:element ref="emix:quantity"/>
2397                <xs:element ref="power:powerItem"/>
2398            </xs:sequence>
2399    </xs:complexType>
2400    <!-- 9.1.5 Reactive Power -->
2401    <xs:element name="varQuantity" type="power:VarQuantityType"/>
2402    <xs:complexType name="VarQuantityType">
2403            <xs:sequence>
2404                <xs:element ref="emix:quantity"/>
2405                <xs:element ref="power:powerReactive"/>
2406            </xs:sequence>
2407    </xs:complexType>
2408    <!-- 9.8 Interface Types -->
2409    <!-- 9.8.1 EndDevices -->
```

```
2410        <!-- updated name of this section to reflect the more generic EndDevice rather than
2411    Meters specifically -->
2412        <xs:element name="endDeviceAsset" type="power:EndDeviceAssetType"
2413    substitutionGroup="emix:emixInterface">
2414            <xs:annotation>
2415                    <xs:documentation>One type of EndDeviceAsset is a MeterAsset which can
2416    perform metering, load management, connect/disconnect, accounting functions, etc. Some
2417    EndDeviceAssets may be connected to a MeterAsset.</xs:documentation>
2418            </xs:annotation>
2419        </xs:element>
2420        <xs:complexType name="EndDeviceAssetType">
2421            <xs:annotation>
2422                    <xs:documentation>The EndDeviceAssets are the physical device or
2423    devices which could be meters or other types of devices that may be of
2424    interest</xs:documentation>
2425            </xs:annotation>
2426            <xs:complexContent>
2427                    <xs:extension base="emix:EmixInterfaceType">
2428                            <xs:sequence>
2429                                    <xs:element ref="power:mRID"/>
2430                            </xs:sequence>
2431                    </xs:extension>
2432            </xs:complexContent>
2433        </xs:complexType>
2434        <!-- 9.8.1.1 Meters -->
2435        <xs:element name="meterAsset" type="power:MeterAssetType"
2436    substitutionGroup="emix:emixInterface"/>
2437        <xs:complexType name="MeterAssetType">
2438            <xs:annotation>
2439                    <xs:documentation>The MeterAsset is the physical device or devices
2440    that performs the role of the meter</xs:documentation>
2441            </xs:annotation>
2442            <xs:complexContent>
2443                    <xs:extension base="emix:EmixInterfaceType">
2444                            <xs:sequence>
2445                                    <xs:element ref="power:mRID"/>
2446                            </xs:sequence>
2447                    </xs:extension>
2448            </xs:complexContent>
2449        </xs:complexType>
2450        <!-- 9.8.2 Nodes -->
2451        <xs:element name="pnode" type="power:PnodeType"
2452    substitutionGroup="emix:emixInterface"/>
2453        <xs:complexType name="PnodeType" mixed="false">
2454            <xs:annotation>
2455                    <xs:documentation>A pricing node is directly associated with a
2456    connectivity node.  It is a pricing location for which market participants submit their
2457    bids, offers, buy/sell CRRs, and settle.</xs:documentation>
2458            </xs:annotation>
2459            <xs:complexContent mixed="false">
2460                    <xs:extension base="emix:EmixInterfaceType">
2461                            <xs:sequence>
2462                                    <xs:element ref="power:node"/>
2463                            </xs:sequence>
2464                    </xs:extension>
2465            </xs:complexContent>
2466        </xs:complexType>
2467        <xs:element name="aggregatedPnode" type="power:AggregatedPnodeType"
2468    substitutionGroup="emix:emixInterface"/>
2469        <xs:complexType name="AggregatedPnodeType" mixed="false">
2470            <xs:annotation>
2471                    <xs:documentation>An aggregated pricing node is a specialized type of
2472    pricing node used to model items such as System Zone, Default Price Zone, Custom Price
2473    Zone, Control Area, Aggregated Generation, Aggregated Particpating Load, Aggregated Non-
2474    Participating Load, Trading Hub, DCA Zone</xs:documentation>
2475            </xs:annotation>
2476            <xs:complexContent mixed="false">
2477                    <xs:extension base="emix:EmixInterfaceType">
2478                            <xs:sequence>
2479                                    <xs:element ref="power:node"/>
2480                            </xs:sequence>
```

```
                    </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <xs:element name="serviceLocation" type="power:ServiceLocationType"
substitutionGroup="emix:emixInterface"/>
        <xs:complexType name="ServiceLocationType" mixed="false">
            <xs:annotation>
                <xs:documentation>A customer ServiceLocation has one or more
ServiceDeliveryPoint(s), which in turn relate to Meters. The location may be a point or
a polygon, depending on the specific circumstances. For distribution, the
ServiceLocation is typically the location of the utility customer's premise.
</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:extension base="emix:EmixInterfaceType">
                    <xs:sequence>
                        <xs:element ref="emix:geographicArea"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <xs:element name="serviceDeliveryPoint" type="power:ServiceDeliveryPointType"
substitutionGroup="emix:emixInterface"/>
        <xs:complexType name="ServiceDeliveryPointType" mixed="false">
            <xs:annotation>
                <xs:documentation>Logical point on the network where the ownership of
the service changes hands. It is one of potentially many service points within a
ServiceLocation, delivering service in accordance with a CustomerAgreement. Used at the
place where a meter may be installed.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:extension base="emix:EmixInterfaceType">
                    <xs:sequence>
                        <xs:element ref="power:node" maxOccurs="1"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <!-- 9.8.3 Transport Interface -->
        <xs:element name="transportInterface" type="power:TransportInterfaceType"
substitutionGroup="emix:emixInterface"/>
        <xs:complexType name="TransportInterfaceType" mixed="false">
            <xs:annotation>
                <xs:documentation>The Transport Interface delineates the edges at
either end of a transport segment.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:extension base="emix:EmixInterfaceType">
                    <xs:sequence>
                        <xs:element name="pointOfReceipt"
type="power:NodeType"/>
                        <xs:element name="pointOfDelivery"
type="power:NodeType"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <!-- 9.8.9 Base Elements for Interfaces -->
        <xs:element name="node" type="power:NodeType"/>
        <xs:simpleType name="NodeType">
            <xs:annotation>
                <xs:documentation>The Node is a place where something changes (often
ownership) or connects on the grid. Many nodes are associated with meters, but not all
are.</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:string"/>
        </xs:simpleType>
        <!-- 9.8.9.1 Base Elements for Interfaces -->
        <!--    The identifier for a EndDevice (meter or other), is mRID from IEC61968-->
        <xs:element name="mRID" type="power:mRIDType"/>
        <xs:simpleType name="mRIDType">
```

```
2552              <xs:annotation>
2553                  <xs:documentation>The mRID identifies the physical device that may be
2554  a CustomerMeter or other types of EndDevices.</xs:documentation>
2555              </xs:annotation>
2556              <xs:restriction base="xs:string"/>
2557      </xs:simpleType>
2558      <!-- 9.9 Enumerations -->
2559      <!--  9.9.1 Voltage   -->
2560      <xs:element name="voltage" type="power:VoltageType"
2561  substitutionGroup="emix:itemBase"/>
2562      <xs:complexType name="VoltageType" mixed="false">
2563          <xs:annotation>
2564              <xs:documentation>Voltage</xs:documentation>
2565          </xs:annotation>
2566          <xs:complexContent mixed="false">
2567              <xs:extension base="emix:ItemBaseType">
2568                  <xs:sequence>
2569                      <xs:element name="itemDescription" type="xs:string"
2570  fixed="Voltage"/>
2571                      <xs:element name="itemUnits" type="xs:string"
2572  fixed="V"/>
2573                      <xs:element ref="emix:scale"/>
2574                  </xs:sequence>
2575              </xs:extension>
2576          </xs:complexContent>
2577      </xs:complexType>
2578      <!-- 9.9.2 Energy Units -->
2579      <xs:element name="energyApparent" type="power:EnergyApparentType"
2580  substitutionGroup="power:energyItem"/>
2581      <xs:complexType name="EnergyApparentType" mixed="false">
2582          <xs:annotation>
2583              <xs:documentation>Apparent Energy, measured in volt-ampere hours
2584  (VAh)</xs:documentation>
2585          </xs:annotation>
2586          <xs:complexContent mixed="false">
2587              <xs:restriction base="power:EnergyItemType">
2588                  <xs:sequence>
2589                      <xs:element name="itemDescription" type="xs:string"
2590  fixed="ApparentEnergy"/>
2591                      <xs:element name="itemUnits" type="xs:string"
2592  fixed="VAh"/>
2593                      <xs:element name="scale" type="emix:SiScaleType"/>
2594                  </xs:sequence>
2595              </xs:restriction>
2596          </xs:complexContent>
2597      </xs:complexType>
2598      <xs:element name="energyReactive" type="power:EnergyReactiveType"
2599  substitutionGroup="power:energyItem"/>
2600      <xs:complexType name="EnergyReactiveType" mixed="false">
2601          <xs:annotation>
2602              <xs:documentation>Reactive Energy, volt-amperes reactive hours
2603  (VARh)</xs:documentation>
2604          </xs:annotation>
2605          <xs:complexContent mixed="false">
2606              <xs:restriction base="power:EnergyItemType">
2607                  <xs:sequence>
2608                      <xs:element name="itemDescription" type="xs:string"
2609  fixed="ReactiveEnergy"/>
2610                      <xs:element name="itemUnits" type="xs:string"
2611  fixed="VARh"/>
2612                      <xs:element name="scale" type="emix:SiScaleType"/>
2613                  </xs:sequence>
2614              </xs:restriction>
2615          </xs:complexContent>
2616      </xs:complexType>
2617      <xs:element name="energyReal" type="power:EnergyRealType"
2618  substitutionGroup="power:energyItem"/>
2619      <xs:complexType name="EnergyRealType" mixed="false">
2620          <xs:annotation>
2621              <xs:documentation>Real Energy, Watt Hours (Wh)</xs:documentation>
2622          </xs:annotation>
```

```
                    <xs:complexContent mixed="false">
                        <xs:restriction base="power:EnergyItemType">
                            <xs:sequence>
                                <xs:element name="itemDescription" type="xs:string"
fixed="RealEnergy"/>
                                <xs:element name="itemUnits" type="xs:string"
fixed="Wh"/>
                                <xs:element name="scale" type="emix:SiScaleType"/>
                            </xs:sequence>
                        </xs:restriction>
                    </xs:complexContent>
        </xs:complexType>
        <!-- ================================================== -->
        <!-- 9.9.5 Base Energy Item Type -->
        <!-- ================================================== -->
        <xs:element name="energyItem" type="power:EnergyItemType"
substitutionGroup="emix:itemBase"/>
        <xs:complexType name="EnergyItemType" abstract="true" mixed="false">
            <xs:annotation>
                <xs:documentation>Base for the measurement of
Energy</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:extension base="emix:ItemBaseType">
                    <xs:sequence>
                        <xs:element name="itemDescription" type="xs:string"/>
                        <xs:element name="itemUnits" type="xs:string"/>
                        <xs:element name="scale" type="emix:SiScaleType"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <!-- ================================================== -->
        <!-- 9.9.4 Power Units -->
        <!-- ================================================== -->
        <!-- ================================================== -->
        <xs:element name="powerApparent" type="power:PowerApparentType"
substitutionGroup="power:powerItem"/>
        <xs:complexType name="PowerApparentType" mixed="false">
            <xs:annotation>
                <xs:documentation>Apparent Power measured in volt-amperes
(VA)</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:restriction base="power:PowerItemType">
                    <xs:sequence>
                        <xs:element name="itemDescription" type="xs:string"
fixed="ApparentPower"/>
                        <xs:element name="itemUnits" type="xs:string"
fixed="VA"/>
                        <xs:element name="scale" type="emix:SiScaleType"/>
                        <xs:element ref="power:powerAttributes"/>
                    </xs:sequence>
                </xs:restriction>
            </xs:complexContent>
        </xs:complexType>
        <!-- ================================================== -->
        <xs:element name="powerReactive" type="power:PowerReactiveType"
substitutionGroup="power:powerItem"/>
        <xs:complexType name="PowerReactiveType" mixed="false">
            <xs:annotation>
                <xs:documentation>Reactive power, measured in volt-amperes reactive
(VAR)</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:restriction base="power:PowerItemType">
                    <xs:sequence>
                        <xs:element name="itemDescription" type="xs:string"
fixed="ReactivePower"/>
                        <xs:element name="itemUnits" type="xs:string"
fixed="VAR"/>
```

```
2694                          <xs:element name="scale" type="emix:SiScaleType"/>
2695                          <xs:element ref="power:powerAttributes"/>
2696                      </xs:sequence>
2697                  </xs:restriction>
2698              </xs:complexContent>
2699      </xs:complexType>
2700      <!-- ================================================== -->
2701      <xs:element name="powerReal" type="power:PowerRealType"
2702  substitutionGroup="power:powerItem"/>
2703      <xs:complexType name="PowerRealType" mixed="false">
2704          <xs:annotation>
2705              <xs:documentation>Real power measured in Watts (W) or Joules/second
2706  (J/s)</xs:documentation>
2707          </xs:annotation>
2708          <xs:complexContent mixed="false">
2709              <xs:restriction base="power:PowerItemType">
2710                  <xs:sequence>
2711                      <xs:element name="itemDescription" type="xs:string"
2712  fixed="RealPower"/>
2713                      <xs:element name="itemUnits">
2714                          <xs:simpleType>
2715                              <xs:restriction base="xs:token">
2716                                  <xs:enumeration value="W"/>
2717                                  <xs:enumeration value="J/s"/>
2718                              </xs:restriction>
2719                          </xs:simpleType>
2720                      </xs:element>
2721                      <xs:element name="scale" type="emix:SiScaleType"/>
2722                      <xs:element ref="power:powerAttributes"/>
2723                  </xs:sequence>
2724              </xs:restriction>
2725          </xs:complexContent>
2726      </xs:complexType>
2727      <!-- ================================================== -->
2728      <!-- 9.9.5 Base Power Item Type -->
2729      <!-- ================================================== -->
2730      <xs:element name="powerItem" type="power:PowerItemType"
2731  substitutionGroup="emix:itemBase"/>
2732      <xs:complexType name="PowerItemType" abstract="true" mixed="false">
2733          <xs:annotation>
2734              <xs:documentation>Base for the measurement of Power</xs:documentation>
2735          </xs:annotation>
2736          <xs:complexContent mixed="false">
2737              <xs:extension base="emix:ItemBaseType">
2738                  <xs:sequence>
2739                      <xs:element name="itemDescription" type="xs:string"/>
2740                      <xs:element name="itemUnits" type="xs:string"/>
2741                      <xs:element name="scale" type="emix:SiScaleType"/>
2742                      <xs:element ref="power:powerAttributes"/>
2743                  </xs:sequence>
2744              </xs:extension>
2745          </xs:complexContent>
2746      </xs:complexType>
2747      <!-- ================================================== -->
2748      <xs:element name="powerAttributes" type="power:PowerAttributesType"/>
2749      <xs:complexType name="PowerAttributesType">
2750          <xs:sequence>
2751              <xs:element name="hertz" type="xs:decimal"/>
2752              <xs:element name="voltage" type="xs:decimal"/>
2753              <xs:element name="ac" type="xs:boolean"/>
2754          </xs:sequence>
2755      </xs:complexType>
2756      <!-- 9.9.5 Enumeration for Reserves and other Power Options -->
2757      <xs:element name="powerOptionType" type="power:PowerOptionTypeType"/>
2758      <xs:simpleType name="PowerOptionTypeType">
2759          <xs:union memberTypes="power:PowerOptionTypeEnumeratedType
2760  emix:EmixExtensionType"/>
2761      </xs:simpleType>
2762      <xs:simpleType name="PowerOptionTypeEnumeratedType">
2763          <xs:annotation>
2764              <xs:documentation>Power Reserve Options</xs:documentation>
```

```
2765                    </xs:annotation>
2766                    <xs:restriction base="xs:string">
2767                            <xs:enumeration value="SpinningReserve"/>
2768                            <xs:enumeration value="NonSpinningReserve"/>
2769                            <xs:enumeration value="OperatingReserve"/>
2770                            <xs:enumeration value="DemandResponse"/>
2771                    </xs:restriction>
2772            </xs:simpleType>
2773    </xs:schema>
```

F.2.2 Power Quality

Demonstrates extensibility of base Warrant classes, as well.

```
2776        <?xml version="1.0" encoding="UTF-8"?>
2777        <!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)
2778
2779        Set includes:
2780            EMIX, EMIX-Requirements, EMIX-Warrants (emix)
2781            Power, Power-Contracts, Power-Quality (power)
2782            Resource (resource)
2783
2784        This set built on the WS-Calendar v1.0 PRD02 Schemas.
2785        -->
2786        <xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
2787        xmlns:emix="http://docs.oasis-open.org/ns/emix"
2788        xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
2789        xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
2790        xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xs="http://www.w3.org/2001/XMLSchema"
2791        targetNamespace="http://docs.oasis-open.org/ns/emix/power"
2792        elementFormDefault="qualified" attributeFormDefault="unqualified">
2793            <xs:include schemaLocation="power.xsd"/>
2794            <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
2795            <!-- 6.0 Quality Warrants -->
2796            <xs:element name="powerQualityWarrant" type="power:PowerQualityWarrantType"
2797        substitutionGroup="emix:baseWarrant"/>
2798            <xs:complexType name="PowerQualityWarrantType" mixed="false">
2799                    <xs:annotation>
2800                            <xs:documentation>A Power Quality Warrant asserts or requires that the
2801        power be of a certain quality or better.</xs:documentation>
2802                    </xs:annotation>
2803                    <xs:complexContent mixed="false">
2804                            <xs:extension base="emix:BaseWarrantType">
2805                                    <xs:sequence>
2806                                            <xs:element name="measurementProtocol"
2807        type="power:MeasurementProtocolType"/>
2808                                            <xs:element name="constraints"
2809        type="power:ArrayOfPowerQualities"/>
2810                                    </xs:sequence>
2811                            </xs:extension>
2812                    </xs:complexContent>
2813            </xs:complexType>
2814            <!-- 6.1 Power Quality -->
2815            <xs:element name="powerQuality" type="power:PowerQualityType">
2816                    <xs:annotation>
2817                            <xs:documentation>Power Quality warrant</xs:documentation>
2818                    </xs:annotation>
2819            </xs:element>
2820            <xs:complexType name="PowerQualityType">
2821                    <xs:annotation>
2822                            <xs:documentation>Power Quality consists of a number of measures,
2823        based on contract, negotiation, and local regulation. Extend Power Qulity to incorporate
2824        new elements by creating additional elements based on
2825        PowerQualityBaseType</xs:documentation>
2826                    </xs:annotation>
2827                    <xs:sequence>
2828                            <xs:element name="measurementProtocol"
2829        type="power:MeasurementProtocolType"/>
2830                            <xs:element name="constraints" type="power:ArrayOfPowerQualities"/>
2831                    </xs:sequence>
2832            </xs:complexType>
```

```
2833            <xs:element name="basePowerQualityMeasurement"
2834    type="power:BasePowerQualityMeasurementType" abstract="true">
2835                <xs:annotation>
2836                    <xs:documentation>Abstract extension object for Power
2837    Qualities</xs:documentation>
2838                </xs:annotation>
2839        </xs:element>
2840        <xs:complexType name="ArrayOfPowerQualities">
2841                <xs:annotation>
2842                    <xs:documentation>Collection of Power Qualities</xs:documentation>
2843                </xs:annotation>
2844                <xs:sequence>
2845                    <xs:element ref="power:basePowerQualityMeasurement" minOccurs="0"
2846    maxOccurs="unbounded"/>
2847                </xs:sequence>
2848        </xs:complexType>
2849        <xs:complexType name="BasePowerQualityMeasurementType" abstract="true">
2850                <xs:annotation>
2851                    <xs:documentation>An identification of the standard or other protocol
2852    used to measure power quality. Sets definition for all other power attributes. Type of
2853    Abstract extension object  for Power Qualities</xs:documentation>
2854                </xs:annotation>
2855        </xs:complexType>
2856        <!-- 6.1 Defined Power Qualities -->
2857        <xs:element name="powerFrequency" type="power:PowerFrequencyType"
2858    substitutionGroup="power:basePowerQualityMeasurement"/>
2859        <xs:element name="supplyVoltageVariations" type="power:SupplyVoltageVariationsType"
2860    substitutionGroup="power:basePowerQualityMeasurement"/>
2861        <xs:element name="rapidVoltageChanges" type="power:RapidVoltageChangesType"
2862    substitutionGroup="power:basePowerQualityMeasurement"/>
2863        <xs:element name="flicker" type="power:FlickerType"
2864    substitutionGroup="power:basePowerQualityMeasurement"/>
2865        <xs:element name="supplyVoltageDips" type="power:SupplyVoltageDipsType"
2866    substitutionGroup="power:basePowerQualityMeasurement"/>
2867        <xs:element name="shortInterruptions" type="power:ShortInterruptionsType"
2868    substitutionGroup="power:basePowerQualityMeasurement"/>
2869        <xs:element name="longInterruptions" type="power:LongInterruptionsType"
2870    substitutionGroup="power:basePowerQualityMeasurement"/>
2871        <xs:element name="temporaryOvervoltage" type="power:TemporaryOvervoltageType"
2872    substitutionGroup="power:basePowerQualityMeasurement"/>
2873        <xs:element name="supplyVoltageImbalance" type="power:SupplyVoltageImbalanceType"
2874    substitutionGroup="power:basePowerQualityMeasurement"/>
2875        <xs:element name="harmonicVoltage" type="power:HarmonicVoltageType"
2876    substitutionGroup="power:basePowerQualityMeasurement"/>
2877        <xs:element name="mainsVoltage" type="power:MainsVoltageType"
2878    substitutionGroup="power:basePowerQualityMeasurement"/>
2879        <!-- 6.2 Defines Power Quality Measuress -->
2880        <xs:complexType name="PowerFrequencyType" mixed="false">
2881                <xs:annotation>
2882                    <xs:documentation>measured Power frequency, e.g. 50.4, 59.9, ,
2883    measured as per referenced measurement protocol. 0 for DC
2884                                    </xs:documentation>
2885                </xs:annotation>
2886                <xs:complexContent mixed="false">
2887                    <xs:extension base="power:BasePowerQualityMeasurementType">
2888                        <xs:sequence>
2889                            <xs:element name="frequency" type="xs:float"/>
2890                        </xs:sequence>
2891                    </xs:extension>
2892                </xs:complexContent>
2893        </xs:complexType>
2894        <xs:complexType name="SupplyVoltageVariationsType" mixed="false">
2895                <xs:annotation>
2896                    <xs:documentation>count of Supply Voltage Variations during the
2897    period, measured as per referenced measurement protocol
2898                </xs:documentation>
2899                </xs:annotation>
2900                <xs:complexContent mixed="false">
2901                    <xs:extension base="power:BasePowerQualityMeasurementType">
2902                        <xs:sequence>
2903                            <xs:element name="count" type="xs:int"/>
```

```
2904                                              </xs:sequence>
2905                                      </xs:extension>
2906                              </xs:complexContent>
2907              </xs:complexType>
2908              <xs:complexType name="RapidVoltageChangesType" mixed="false">
2909                      <xs:annotation>
2910                              <xs:documentation>count of Rapid Voltage Changes during the period,
2911      measured as per referenced measurement protocol
2912                      </xs:documentation>
2913                      </xs:annotation>
2914                      <xs:complexContent mixed="false">
2915                              <xs:extension base="power:BasePowerQualityMeasurementType">
2916                                      <xs:sequence>
2917                                              <xs:element name="count" type="xs:int"/>
2918                                      </xs:sequence>
2919                              </xs:extension>
2920                      </xs:complexContent>
2921              </xs:complexType>
2922              <xs:complexType name="FlickerType" mixed="false">
2923                      <xs:annotation>
2924                              <xs:documentation>count of Flicker during the period, measured as per
2925      referenced measurement protocol
2926                      </xs:documentation>
2927                      </xs:annotation>
2928                      <xs:complexContent mixed="false">
2929                              <xs:extension base="power:BasePowerQualityMeasurementType">
2930                                      <xs:sequence>
2931                                              <xs:element name="count" type="xs:int"/>
2932                                      </xs:sequence>
2933                              </xs:extension>
2934                      </xs:complexContent>
2935              </xs:complexType>
2936              <xs:complexType name="SupplyVoltageDipsType" mixed="false">
2937                      <xs:annotation>
2938                              <xs:documentation>count of Supply Voltage Dips during the period,
2939      measured as per referenced measurement protocol
2940                      </xs:documentation>
2941                      </xs:annotation>
2942                      <xs:complexContent mixed="false">
2943                              <xs:extension base="power:BasePowerQualityMeasurementType">
2944                                      <xs:sequence>
2945                                              <xs:element name="count" type="xs:int"/>
2946                                      </xs:sequence>
2947                              </xs:extension>
2948                      </xs:complexContent>
2949              </xs:complexType>
2950              <xs:complexType name="ShortInterruptionsType" mixed="false">
2951                      <xs:annotation>
2952                              <xs:documentation>count of Short Interruptions during the period,
2953      measured as per referenced measurement protocol
2954                      </xs:documentation>
2955                      </xs:annotation>
2956                      <xs:complexContent mixed="false">
2957                              <xs:extension base="power:BasePowerQualityMeasurementType">
2958                                      <xs:sequence>
2959                                              <xs:element name="count" type="xs:int"/>
2960                                      </xs:sequence>
2961                              </xs:extension>
2962                      </xs:complexContent>
2963              </xs:complexType>
2964              <xs:complexType name="LongInterruptionsType" mixed="false">
2965                      <xs:annotation>
2966                              <xs:documentation>count of Long Interuptions during the period,
2967      measured as per referenced measurement protocol
2968                      </xs:documentation>
2969                      </xs:annotation>
2970                      <xs:complexContent mixed="false">
2971                              <xs:extension base="power:BasePowerQualityMeasurementType">
2972                                      <xs:sequence>
2973                                              <xs:element name="count" type="xs:int"/>
2974                                      </xs:sequence>
```

```
                                </xs:extension>
                        </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="TemporaryOvervoltageType" mixed="false">
                <xs:annotation>
                        <xs:documentation>count of Temporary Overvoltage Events during the
period, measured as per referenced measurement protocol
                </xs:documentation>
                </xs:annotation>
                <xs:complexContent mixed="false">
                        <xs:extension base="power:BasePowerQualityMeasurementType">
                                <xs:sequence>
                                        <xs:element name="count" type="xs:int"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="SupplyVoltageImbalanceType" mixed="false">
                <xs:annotation>
                        <xs:documentation>count of Supply Voltage Imbalance events during the
period,  measured as per referenced measurement protocol. Not meaningful for DC.
                </xs:documentation>
                </xs:annotation>
                <xs:complexContent mixed="false">
                        <xs:extension base="power:BasePowerQualityMeasurementType">
                                <xs:sequence>
                                        <xs:element name="count" type="xs:int"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="HarmonicVoltageType" mixed="false">
                <xs:annotation>
                        <xs:documentation>Harmonic Voltage during the period, measured as per
referenced measurement protocol. For DC, distortion is with respect to a signal of 0 Hz,
   The period is usually much shorter than other power quality measures.
                </xs:documentation>
                </xs:annotation>
                <xs:complexContent mixed="false">
                        <xs:extension base="power:BasePowerQualityMeasurementType">
                                <xs:sequence>
                                        <xs:element name="voltage" type="xs:float"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="MainsVoltageType" mixed="false">
                <xs:annotation>
                        <xs:documentation>Mains [Signaling] Voltage. Nominal value, e.g, 110,
130, 220, 208. See referenced measurement protocol for definition.
                </xs:documentation>
                </xs:annotation>
                <xs:complexContent mixed="false">
                        <xs:extension base="power:BasePowerQualityMeasurementType">
                                <xs:sequence>
                                        <xs:element name="voltage" type="xs:float"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <xs:simpleType name="MeasurementProtocolType">
                <xs:union memberTypes="power:MeasurementProtocolEnumeratedType
emix:EmixExtensionType"/>
        </xs:simpleType>
        <xs:simpleType name="MeasurementProtocolEnumeratedType">
                <xs:annotation>
                        <xs:documentation>An identification of the standard or other protocol
used to measure power quality. Sets definition for all other power
attributes</xs:documentation>
                </xs:annotation>
                <xs:restriction base="xs:string">
```

```
3046                    <xs:enumeration value="EN 50160"/>
3047                    <xs:enumeration value="IEEE 1549-2009"/>
3048            </xs:restriction>
3049        </xs:simpleType>
3050    </xs:schema>
```

F.2.3 Power Products.xsd

```
3052    <?xml version="1.0" encoding="UTF-8"?>
3053    <!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)
3054
3055    Set includes:
3056        EMIX, EMIX-Requirements, EMIX-Warrants (emix)
3057        Power, Power-Contracts, Power-Quality (power)
3058        Resource (resource)
3059
3060    This set built on the WS-Calendar v1.0 PRD02 Schemas.
3061    -->
3062    <xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
3063    xmlns:emix="http://docs.oasis-open.org/ns/emix"
3064    xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
3065    xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
3066    xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xs="http://www.w3.org/2001/XMLSchema"
3067    targetNamespace="http://docs.oasis-open.org/ns/emix/power"
3068    elementFormDefault="qualified" attributeFormDefault="unqualified">
3069        <xs:include schemaLocation="power.xsd"/>
3070        <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
3071        <!-- 2.0 Power Products -->
3072        <xs:element name="powerProductDescription" type="power:PowerProductDescriptionType"
3073    substitutionGroup="emix:productDescription"/>
3074        <xs:complexType name="PowerProductDescriptionType" abstract="true">
3075            <xs:annotation>
3076                    <xs:documentation>Type of Product Description for simple transactions.
3077    Also used as template for other Power Product Description Types. A product is advertised
3078    (or bought) with a constant power, which dictates the rate of delivery. After a
3079    specified duration, energy has been delivered, at a price per energy, price per unit
3080    energy</xs:documentation>
3081            </xs:annotation>
3082            <xs:complexContent>
3083                    <xs:extension base="emix:ProductDescriptionType">
3084                            <xs:sequence>
3085                                    <xs:element ref="power:productType"/>
3086                                    <xs:element ref="emix:emixInterface"/>
3087                                    <xs:element ref="power:unitEnergyPrice"/>
3088                                    <xs:element ref="power:powerItem"/>
3089                                    <xs:element name="charges"
3090    type="power:ArrayOfCharges"/>
3091                            </xs:sequence>
3092                    </xs:extension>
3093            </xs:complexContent>
3094        </xs:complexType>
3095        <!-- 2.1 Full Requirements Power -->
3096        <xs:element name="fullRequirementsPower" type="power:FullRequirementsPowerType"
3097    substitutionGroup="power:powerProductDescription"/>
3098        <xs:complexType name="FullRequirementsPowerType">
3099            <xs:annotation>
3100                    <xs:documentation>Type of Product Description for Supplier to provide
3101    for full requirements of buyer. Simple prices, will supply all used. Demand Charges
3102    Optional. Often used in retail residential rates.</xs:documentation>
3103            </xs:annotation>
3104            <xs:complexContent>
3105                    <xs:extension base="power:PowerProductDescriptionType">
3106                            <xs:sequence>
3107                                    <xs:element ref="emix:priceBase"/>
3108                                    <xs:element name="maximumPower"
3109    type="emix:QuantityType"/>
3110                                    <xs:element name="minimumPower"
3111    type="emix:QuantityType"/>
3112                            </xs:sequence>
3113                    </xs:extension>
```

```
                    </xs:complexContent>
            </xs:complexType>
            <!-- 2.2 Block Power Full Requirements -->
            <xs:element name="blockPowerFullRequirements"
type="power:BlockPowerFullRequirementsType"
substitutionGroup="power:powerProductDescription"/>
            <xs:complexType name="BlockPowerFullRequirementsType">
                    <xs:annotation>
                            <xs:documentation>Type of Product Description for Supplier to provide
for full requirements of buyer in "blocks". Price is constant within a block, but
changes as each block is used during a period. Demand Charges MAY be included. Often
used in retail residential rates.</xs:documentation>
                    </xs:annotation>
                    <xs:complexContent>
                            <xs:extension base="power:PowerProductDescriptionType">
                                    <xs:sequence minOccurs="1" maxOccurs="unbounded">
                                            <xs:element name="powerPriceTiers"
type="power:ArrayOfBlockPowerPrices" minOccurs="1" maxOccurs="1"/>
                                            <xs:element name="maximumPower"
type="emix:QuantityType" minOccurs="0" maxOccurs="1"/>
                                            <xs:element name="minimumPower"
type="emix:QuantityType" minOccurs="0" maxOccurs="1"/>
                                    </xs:sequence>
                            </xs:extension>
                    </xs:complexContent>
            </xs:complexType>
            <!-- 2.3 Transport Service -->
            <xs:element name="transportProduct" type="power:TransportProductType"
substitutionGroup="power:powerProductDescription"/>
            <xs:complexType name="TransportProductType">
                    <xs:annotation>
                            <xs:documentation>Type of Product Description for charges and revenue
related to Transport Services for a Power Product, i.e., the movement of Power through
Transmission and Distribution. The Interface used matches a segment of the transport
infrastructure, usually idetifed by an injection node and a delivery
node.</xs:documentation>
                    </xs:annotation>
                    <xs:complexContent>
                            <xs:extension base="power:PowerProductDescriptionType">
                                    <xs:sequence>
                                            <xs:element name="transportCharges"
type="power:ArrayOfTransportCharges"/>
                                    </xs:sequence>
                            </xs:extension>
                    </xs:complexContent>
            </xs:complexType>
            <!-- 2.1 TEMIX Power -->
            <xs:element name="temixPower" type="power:TemixPowerType"
substitutionGroup="emix:productDescription"/>
            <xs:complexType name="TemixPowerType">
                    <xs:annotation>
                            <xs:documentation>Type of contract Product Description for Supplier to
a specific sized block of power to buyer. Simple prices, will supply fixed block.
Derived directly from emix:ProductDescriptionType rather than
power:PowerProductDescriptionType because optionality stripped out.</xs:documentation>
                    </xs:annotation>
                    <xs:complexContent>
                            <xs:extension base="emix:ProductDescriptionType">
                                    <xs:sequence>
                                            <xs:element ref="emix:emixInterface"/>
                                            <xs:element ref="emix:price" minOccurs="1"
maxOccurs="1"/>
                                            <xs:element ref="power:powerItem" minOccurs="1"
maxOccurs="1"/>
                                            <xs:element ref="power:energyItem" minOccurs="1"
maxOccurs="1"/>
                                            <xs:element ref="emix:quantity" minOccurs="1"
maxOccurs="1"/>
                                    </xs:sequence>
                            </xs:extension>
                    </xs:complexContent>
```

```
3185            </xs:complexType>
3186            <!--
3187    ================================================================================
3188    ====== -->
3189        <!-- Charge Defintions -->
3190        <!--
3191    ================================================================================
3192    ====== -->
3193        <!-- 2.5 Charge Abstractions -->
3194        <xs:element name="baseCharge" type="power:BaseChargeType" abstract="true">
3195            <xs:annotation>
3196                <xs:documentation>Abstract extension object for Emix Power Product
3197    Charges</xs:documentation>
3198            </xs:annotation>
3199        </xs:element>
3200        <xs:complexType name="ArrayOfCharges">
3201            <xs:annotation>
3202                <xs:documentation>Collection of Emix Power Product
3203    Charges</xs:documentation>
3204            </xs:annotation>
3205            <xs:sequence>
3206                <xs:element ref="power:baseCharge" minOccurs="0"
3207    maxOccurs="unbounded"/>
3208            </xs:sequence>
3209        </xs:complexType>
3210        <xs:complexType name="BaseChargeType" abstract="true">
3211            <xs:annotation>
3212                <xs:documentation>Type of Abstract extension object for  Emix Power
3213    Product Charges</xs:documentation>
3214            </xs:annotation>
3215        </xs:complexType>
3216        <!--
3217    ================================================================================
3218    ====== -->
3219        <!-- 2.6 General Charges -->
3220        <!-- 2.6.1 Blocks for use in Block Power -->
3221        <xs:element name="blockPowerPrice" type="power:BlockPowerPriceType"
3222    substitutionGroup="power:baseCharge"/>
3223        <xs:complexType name="BlockPowerPriceType" mixed="false">
3224            <xs:complexContent mixed="false">
3225                <xs:extension base="power:BaseChargeType">
3226                    <xs:sequence>
3227                        <xs:element ref="emix:priceBase"/>
3228                        <xs:element name="maximumEnergyQuantity"
3229    type="emix:QuantityType"/>
3230                    </xs:sequence>
3231                </xs:extension>
3232            </xs:complexContent>
3233        </xs:complexType>
3234        <xs:complexType name="ArrayOfBlockPowerPrices">
3235            <xs:annotation>
3236                <xs:documentation>Collection of Emix Block Power
3237    Prices</xs:documentation>
3238            </xs:annotation>
3239            <xs:sequence>
3240                <xs:element ref="power:blockPowerPrice" minOccurs="0"
3241    maxOccurs="unbounded"/>
3242            </xs:sequence>
3243        </xs:complexType>
3244        <!--
3245    ================================================================================
3246    ====== -->
3247        <!-- 2.6.2 Demand Charges -->
3248        <xs:element name="demandCharge" type="power:DemandChargeType"
3249    substitutionGroup="power:baseCharge"/>
3250        <xs:complexType name="DemandChargeType" mixed="false">
3251            <xs:complexContent mixed="false">
3252                <xs:extension base="power:BaseChargeType">
3253                    <xs:sequence>
3254                        <xs:element name="demandChargeUnits"
3255    type="power:PowerItemType"/>
```

```
3256                                        <xs:element name="demandChargeFloor"
3257   type="emix:QuantityType"/>
3258                                        <xs:element name="demandChargeRate"
3259   type="emix:PriceBaseType"/>
3260                                        <xs:element name="measurementInterval"
3261   type="emix:DurationType"/>
3262                                        <xs:element name="collectionInterval"
3263   type="emix:DurationType"/>
3264                                        <xs:element name="collectionPeriod"
3265   type="emix:DurationType"/>
3266                                        <xs:element name="chargeDuration"
3267   type="emix:DurationType"/>
3268                                </xs:sequence>
3269                        </xs:extension>
3270                </xs:complexContent>
3271        </xs:complexType>
3272        <!--
3273   ================================================================================
3274   ====== -->
3275        <!-- Transport Charges and Losses Types -->
3276        <!--
3277   ================================================================================
3278   ====== -->
3279        <!-- 2.7 Transport Abstract Types -->
3280        <xs:element name="baseTransportCharge" type="power:BaseTransportChargeType"
3281   abstract="true" substitutionGroup="power:baseCharge">
3282                <xs:annotation>
3283                        <xs:documentation>Abstract extension object for Emix Power Product
3284   Charges</xs:documentation>
3285                </xs:annotation>
3286        </xs:element>
3287        <xs:complexType name="ArrayOfTransportCharges">
3288                <xs:annotation>
3289                        <xs:documentation>Collection of Emix Power Transport Product
3290   Charges</xs:documentation>
3291                </xs:annotation>
3292                <xs:sequence>
3293                        <xs:element ref="power:baseTransportCharge" minOccurs="0"
3294   maxOccurs="unbounded"/>
3295                </xs:sequence>
3296        </xs:complexType>
3297        <xs:complexType name="BaseTransportChargeType" abstract="true" mixed="false">
3298                <xs:annotation>
3299                        <xs:documentation>Type of Abstract extension object for  Emix
3300   Transport Charges</xs:documentation>
3301                </xs:annotation>
3302                <xs:complexContent mixed="false">
3303                        <xs:extension base="power:BaseChargeType"/>
3304                </xs:complexContent>
3305        </xs:complexType>
3306        <!--
3307   ================================================================================
3308   ====== -->
3309        <!-- 2.8 Congestion and Loss Charges -->
3310        <!-- 2.8.1 Congestion Revenue Rights Charge-->
3311        <xs:element name="congestionRevenueRights" type="power:CongestionRevenueRightsType"
3312   substitutionGroup="power:baseTransportCharge"/>
3313        <xs:complexType name="CongestionRevenueRightsType" mixed="false">
3314                <xs:annotation>
3315                        <xs:documentation>Financial Hedge for Congestion, a forward contract
3316   for congestion revenues to potentially offset congestion charges. Also known as
3317   Financial Transmission Rights or Congestion Revenue Rights</xs:documentation>
3318                </xs:annotation>
3319                <xs:complexContent mixed="false">
3320                        <xs:extension base="power:BaseTransportChargeType">
3321                                <xs:sequence>
3322                                        <xs:element ref="power:transportInterface"/>
3323                                        <xs:element ref="power:transportCongestionFee"/>
3324                                </xs:sequence>
3325                        </xs:extension>
3326                </xs:complexContent>
```

```
3327            </xs:complexType>
3328            <!-- 2.8.2 Congestion Charge-->
3329            <xs:element name="congestionCharge" type="power:CongestionChargeType"
3330    substitutionGroup="power:baseTransportCharge"/>
3331            <xs:complexType name="CongestionChargeType" mixed="false">
3332                    <xs:annotation>
3333                            <xs:documentation>Congestion Charge is the cost of purchasing the
3334    right to transfer power over a given segment of the grid.</xs:documentation>
3335                    </xs:annotation>
3336                    <xs:complexContent mixed="false">
3337                            <xs:extension base="power:BaseTransportChargeType">
3338                                    <xs:sequence>
3339                                            <xs:element ref="power:transportInterface"/>
3340                                            <xs:element ref="power:transportCongestionFee"/>
3341                                    </xs:sequence>
3342                            </xs:extension>
3343                    </xs:complexContent>
3344            </xs:complexType>
3345            <!-- 2.8.3 Marginal Loss Charge -->
3346            <xs:element name="marginalLossCharge" type="power:MarginalLossChargeType"
3347    substitutionGroup="power:baseTransportCharge"/>
3348            <xs:complexType name="MarginalLossChargeType" mixed="false">
3349                    <xs:complexContent mixed="false">
3350                            <xs:extension base="power:BaseTransportChargeType">
3351                                    <xs:sequence>
3352                                            <xs:element ref="power:marginalLossFee"/>
3353                                    </xs:sequence>
3354                            </xs:extension>
3355                    </xs:complexContent>
3356            </xs:complexType>
3357            <!-- 2.8.4 Marginal Loss -->
3358            <xs:element name="marginalLoss" type="power:MarginalLossType"
3359    substitutionGroup="power:baseTransportCharge"/>
3360            <xs:complexType name="MarginalLossType" mixed="false">
3361                    <xs:complexContent mixed="false">
3362                            <xs:extension base="power:BaseTransportChargeType">
3363                                    <xs:sequence>
3364                                            <xs:element ref="power:lossFactor"/>
3365                                    </xs:sequence>
3366                            </xs:extension>
3367                    </xs:complexContent>
3368            </xs:complexType>
3369            <!-- 2.8.5 Conversion Loss -->
3370            <xs:element name="conversionLoss" type="power:ConversionLossType"
3371    substitutionGroup="power:baseTransportCharge"/>
3372            <xs:complexType name="ConversionLossType" mixed="false">
3373                    <xs:complexContent mixed="false">
3374                            <xs:extension base="power:BaseTransportChargeType">
3375                                    <xs:sequence>
3376                                            <xs:element ref="power:pnode"/>
3377                                            <xs:element ref="power:lossFactor"/>
3378                                    </xs:sequence>
3379                            </xs:extension>
3380                    </xs:complexContent>
3381            </xs:complexType>
3382            <xs:element name="transportAccessFee" type="power:TransportAccessFeeType"
3383    substitutionGroup="power:baseTransportCharge"/>
3384            <xs:complexType name="TransportAccessFeeType" mixed="false">
3385                    <xs:annotation>
3386                            <xs:documentation>Transport Access Fee is a Fixed Charge (not
3387    dependent on congestion or quantity) to access a transport system.</xs:documentation>
3388                    </xs:annotation>
3389                    <xs:complexContent mixed="false">
3390                            <xs:extension base="power:BaseTransportChargeType">
3391                                    <xs:sequence>
3392                                            <xs:element ref="power:transportInterface"/>
3393                                            <xs:element ref="emix:price"/>
3394                                    </xs:sequence>
3395                            </xs:extension>
3396                    </xs:complexContent>
3397            </xs:complexType>
```

```
3398          <!--
3399 ================================================================================
3400 ====== -->
3401       <!-- 2.9 Elemental Charge and Loss Types -->
3402       <!--
3403 ================================================================================
3404 ====== -->
3405       <!-- 2.9.3 Loss Fee  -->
3406       <xs:element name="marginalLossFee" type="power:MarginalLossFeeType"/>
3407       <xs:simpleType name="MarginalLossFeeType">
3408             <xs:annotation>
3409                   <xs:documentation>Marginal Loss Fee</xs:documentation>
3410             </xs:annotation>
3411             <xs:restriction base="xs:decimal"/>
3412       </xs:simpleType>
3413       <!-- 2.9.4 Transport Congestion Fee -->
3414       <xs:element name="transportCongestionFee" type="power:TransportCongestionFeeType"/>
3415       <xs:simpleType name="TransportCongestionFeeType">
3416             <xs:annotation>
3417                   <xs:documentation>Financial Transmission Rights (FTR) regarding
3418 transmission capacity.</xs:documentation>
3419             </xs:annotation>
3420             <xs:restriction base="xs:decimal"/>
3421       </xs:simpleType>
3422       <!-- 2.9.5 Loss Factor -->
3423       <xs:element name="lossFactor" type="power:LossFactorType"/>
3424       <xs:simpleType name="LossFactorType">
3425             <xs:annotation>
3426                   <xs:documentation>Reduction in amount delivered as product travels.
3427 (lossFactor * purchase amount) = delivered amount</xs:documentation>
3428             </xs:annotation>
3429             <xs:restriction base="xs:float">
3430                   <xs:maxInclusive value="1"/>
3431             </xs:restriction>
3432       </xs:simpleType>
3433       <!-- 2.9.6 Enumeration & Simple Types for Products -->
3434       <xs:element name="productType" type="power:ProductTypeType"/>
3435       <xs:simpleType name="ProductTypeType">
3436             <xs:union memberTypes="power:ProductTypeEnumeratedType emix:EmixExtensionType
3437 power:PowerOptionTypeType"/>
3438       </xs:simpleType>
3439       <xs:simpleType name="ProductTypeEnumeratedType">
3440             <xs:restriction base="xs:string">
3441                   <xs:enumeration value="Energy"/>
3442                   <xs:enumeration value="Transport"/>
3443                   <xs:enumeration value="EnergyOption"/>
3444                   <xs:enumeration value="TransportOption"/>
3445                   <xs:enumeration value="FullRequirementsPower"/>
3446                   <xs:enumeration value="FullRequirementsPowerWithDemandCharge"/>
3447                   <xs:enumeration value="FullRequirementsPowerWithMaximumAndMinimum"/>
3448                   <xs:enumeration value="HourlyDayAhead"/>
3449                   <xs:enumeration value="Ex-AnteRealTimePrice"/>
3450                   <xs:enumeration value="TimeOfUsePricing"/>
3451                   <xs:enumeration value="Transport"/>
3452                   <xs:enumeration value="CongestionRevenueRights"/>
3453             </xs:restriction>
3454       </xs:simpleType>
3455 </xs:schema>
```

F.3 Resource.xsd

```
3457 <?xml version="1.0" encoding="UTF-8"?>
3458 <!-- edited with XMLSpy v2011 rel. 2 (x64) (http://www.altova.com) by Toby Considine
3459 (TC9, Inc) -->
3460 <!-- resource.xsd - Resource Descriptions for OASIS EMIX 1.0 WD23 (20110411)
3461
3462 Set includes:
3463     EMIX, EMIX-Requirements, EMIX-Warrants (emix)
3464     Power, Power-Contracts, Power-Quality (power)
3465     Resource (resource)
```

```
3466              This set built on the WS-Calendar v1.0 PRD02 Schemas.
3467              -->
3468      <xs:schema xmlns:resource="http://docs.oasis-open.org/ns/emix/power/resource"
3469      xmlns:power="http://docs.oasis-open.org/ns/emix/power" xmlns:emix="http://docs.oasis-
3470      open.org/ns/emix" xmlns:xcal="urn:ietf:params:xml:ns:icalendar-2.0"
3471      xmlns:clm5ISO42173A="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
3472      xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xs="http://www.w3.org/2001/XMLSchema"
3473      targetNamespace="http://docs.oasis-open.org/ns/emix/power/resource"
3474      elementFormDefault="qualified" attributeFormDefault="unqualified">
3475         <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
3476         <xs:import namespace="http://docs.oasis-open.org/ns/emix/power"
3477      schemaLocation="power.xsd"/>
3478         <!-- 3.0 Resource are described in terms of their capabilities Capabilities to aid in
3479      the matching of need and supplier -->
3480         <xs:element name="loadReduction" type="resource:LoadReductionType"
3481      substitutionGroup="emix:productDescription"/>
3482         <xs:element name="generation" type="resource:GenerationType"
3483      substitutionGroup="emix:productDescription"/>
3484         <xs:element name="activeReserve" type="resource:ActiveReserveType"
3485      substitutionGroup="emix:productDescription"/>
3486         <xs:element name="regulationService" type="resource:RegulationServiceType"
3487      substitutionGroup="emix:productDescription"/>
3488         <xs:element name="productVoltageRegulation"
3489      type="resource:ProductVoltageRegulationType"
3490      substitutionGroup="emix:productDescription"/>
3491         <!-- 3.1 Load resource -->
3492         <xs:complexType name="LoadReductionType">
3493              <xs:annotation>
3494                      <xs:documentation>A Load Reduction Resource ramps down, stays down,
3495      and then ramps up. For stagingRamps, endRamp is less than beginRamp. For recoveryRamps,
3496      endRamp is greater than beginRamp.</xs:documentation>
3497              </xs:annotation>
3498              <xs:complexContent>
3499                      <xs:extension base="resource:PowerResponseType"/>
3500              </xs:complexContent>
3501         </xs:complexType>
3502         <!-- 3.2 Generation Resource -->
3503         <xs:complexType name="GenerationType">
3504              <xs:annotation>
3505                      <xs:documentation>A Generation Resource ramps up, stays up, and then
3506      ramps down. For stagingRamps, endRamp is greater than beginRamp. For recoveryRamps,
3507      endRamp is less than beginRamp.</xs:documentation>
3508              </xs:annotation>
3509              <xs:complexContent>
3510                      <xs:extension base="resource:PowerResponseType">
3511                              <xs:sequence>
3512                                      <xs:element name="Type"
3513      type="resource:ResourceTypeType" minOccurs="0" maxOccurs="1"/>
3514                              </xs:sequence>
3515                      </xs:extension>
3516              </xs:complexContent>
3517         </xs:complexType>
3518         <!-- 3.5 Active Reserve -->
3519         <xs:complexType name="ActiveReserveType">
3520              <xs:annotation>
3521                      <xs:documentation>Active Reserve</xs:documentation>
3522              </xs:annotation>
3523              <xs:complexContent>
3524                      <xs:extension base="resource:ResourceDescriptionType">
3525                              <xs:sequence>
3526                                      <xs:element ref="power:powerOptionType"/>
3527                                      <xs:element ref="resource:targetRegulation"/>
3528                                      <xs:element ref="resource:dispatchTime"/>
3529                                      <xs:element ref="emix:autonomous" minOccurs="0">
3530                                              <xs:annotation>
3531                                                      <xs:documentation>Resource provides
3532      autonomous management of its local circuits. If true, service notes local conditions and
3533      dispatches itself. If false, it waits for dispatch request from VTN.</xs:documentation>
3534                                              </xs:annotation>
3535                                      </xs:element>
```

```xml
                                        <xs:element ref="resource:maximumDeliveryRate"/>
                                        <xs:element ref="resource:minimumDeliveryRate"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <!-- 3.6 Regulation Service Product -->
        <xs:complexType name="RegulationServiceType">
                <xs:annotation>
                        <xs:documentation>Regulation Service</xs:documentation>
                </xs:annotation>
                <xs:complexContent>
                        <xs:extension base="resource:ResourceDescriptionType">
                                <xs:sequence>
                                        <xs:element ref="resource:productTypeRegulation"/>
                                        <xs:element ref="resource:targetRegulation"/>
                                        <xs:element ref="resource:dispatchUp"/>
                                        <xs:element ref="resource:dispatchDown"/>
                                        <xs:element ref="emix:autonomous"/>
                                        <!-- frequency response faster than freq regulation -->
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <!-- 3.6 Voltage Regulation -->
        <xs:complexType name="ProductVoltageRegulationType">
                <xs:annotation>
                        <xs:documentation>Voltage Regulation</xs:documentation>
                        <xs:appinfo>At the end of the scheduled interval, VAR resources should
return to their original state</xs:appinfo>
                </xs:annotation>
                <xs:complexContent>
                        <xs:extension base="resource:ResourceDescriptionType">
                                <xs:sequence>
                                        <!-- *** <xs:element name="voltVar"
type="resource:VoltVarType" maxOccurs="unbounded"/> -->
                                        <xs:element ref="resource:rampTime">
                                                <xs:annotation>
                                                        <xs:documentation>Requested ramp time to
move from the current setpoint to the new setpoint</xs:documentation>
                                                </xs:annotation>
                                        </xs:element>
                                        <xs:element ref="resource:window">
                                                <xs:annotation>
                                                        <xs:documentation>Time window within
which to randomly execute the command. If the time window is zero, the command will be
executed immediately, (if not included, then default time window for this function will
be used)</xs:documentation>
                                                </xs:annotation>
                                        </xs:element>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
        <!-- 3.9 Resource Description -->
        <xs:complexType name="ResourceDescriptionType">
                <xs:annotation>
                        <xs:documentation>Resource Description based on the EMIX Product
Description.</xs:documentation>
                </xs:annotation>
                <xs:complexContent>
                        <xs:extension base="emix:ProductDescriptionType">
                                <xs:sequence>
                                        <xs:element ref="resource:mrid"/>
                                        <xs:element ref="emix:emixInterface"/>
                                        <xs:element ref="emix:constraints" minOccurs="0"
maxOccurs="1"/>
                                </xs:sequence>
                        </xs:extension>
                </xs:complexContent>
        </xs:complexType>
```

```
3608        <!-- 3.9.1 Resource Types -->
3609        <xs:element name="resourceType" type="resource:ResourceTypeType"/>
3610        <xs:simpleType name="ResourceTypeType">
3611                <xs:union memberTypes="resource:ResourceTypeEnumeratedType
3612    emix:EmixExtensionType"/>
3613        </xs:simpleType>
3614        <xs:simpleType name="ResourceTypeEnumeratedType">
3615                <xs:annotation>
3616                        <xs:documentation>Resource types share common responsiveness and
3617    predictability characteristics, sometimes covarying across resources in the same class.
3618    (Example: Solar in the same region failing at the same time)</xs:documentation>
3619                </xs:annotation>
3620                <xs:restriction base="xs:token">
3621                        <xs:enumeration value="DispatchableHydro"/>
3622                        <xs:enumeration value="NonDispatchableHydro"/>
3623                        <xs:enumeration value="WindGeneration"/>
3624                        <xs:enumeration value="SolarGeneration"/>
3625                        <xs:enumeration value="TollingContract"/>
3626                        <xs:enumeration value="AggregateResource"/>
3627                        <xs:enumeration value="DispatchableStorage"/>
3628                </xs:restriction>
3629        </xs:simpleType>
3630        <!-- 3.9.2 Regulation Products -->
3631        <xs:element name="productTypeRegulation" type="resource:ProductTypeRegulationType"/>
3632        <xs:simpleType name="ProductTypeRegulationType">
3633                <xs:annotation>
3634                        <xs:documentation>enumerates the Voltage Regulation
3635    Products</xs:documentation>
3636                </xs:annotation>
3637                <xs:restriction base="xs:string">
3638                        <xs:enumeration value="RegulationUp"/>
3639                        <xs:enumeration value="RegulationDn"/>
3640                        <xs:enumeration value="RegulationUp-Dn"/>
3641                </xs:restriction>
3642        </xs:simpleType>
3643        <!-- 4.0 Resource Semantics -->
3644        <!-- 4.1 Resource Capability -->
3645        <xs:element name="powerResponse" type="resource:PowerResponseType"/>
3646        <xs:complexType name="PowerResponseType" abstract="true">
3647                <xs:annotation>
3648                        <xs:documentation>Generic model describing the power response
3649    capabilities of a resource</xs:documentation>
3650                </xs:annotation>
3651                <xs:complexContent>
3652                        <xs:extension base="resource:ResourceDescriptionType">
3653                                <xs:sequence>
3654                                        <xs:element ref="resource:stagingRamp" minOccurs="0"
3655    maxOccurs="1"/>
3656                                        <xs:element ref="resource:maximumResponse"
3657    minOccurs="0" maxOccurs="1"/>
3658                                        <xs:element ref="resource:minimumResponse"
3659    maxOccurs="1"/>
3660                                        <xs:element ref="resource:recoveryRamp" minOccurs="0"
3661    maxOccurs="1"/>
3662                                        <xs:element ref="resource:offerCurve" minOccurs="0"
3663    maxOccurs="1"/>
3664                                </xs:sequence>
3665                        </xs:extension>
3666                </xs:complexContent>
3667        </xs:complexType>
3668        <!-- 4.1 Ramp Rates -->
3669        <!-- 4.1.3 Power Ramp Rate -->
3670        <xs:element name="stagingRamp" type="resource:ArrayOfRampSegments"/>
3671        <xs:element name="recoveryRamp" type="resource:ArrayOfRampSegments"/>
3672        <xs:element name="powerRamp" type="resource:ArrayOfRampSegments"/>
3673        <xs:complexType name="PowerRampType">
3674                <xs:annotation>
3675                        <xs:documentation>A Power Ramp is an Array of of Ramp Segments that
3676    describing a Resource's ability to change level. A Power Ramp is either monotonically
3677    increasing or monotonically decreasing.</xs:documentation>
3678                </xs:annotation>
```

```
3679                 <xs:sequence>
3680                         <xs:element ref="resource:rampSegments"/>
3681                 </xs:sequence>
3682         </xs:complexType>
3683         <xs:element name="rampSegments" type="resource:ArrayOfRampSegments"/>
3684         <xs:complexType name="ArrayOfRampSegments">
3685                 <xs:annotation>
3686                         <xs:documentation>Collection of Power Ramp Segments</xs:documentation>
3687                 </xs:annotation>
3688                 <xs:sequence>
3689                         <xs:element ref="resource:powerRampSegment" minOccurs="0"
3690 maxOccurs="unbounded"/>
3691                 </xs:sequence>
3692         </xs:complexType>
3693         <xs:element name="powerRampSegment" type="resource:PowerRampSegmentType"/>
3694         <xs:complexType name="PowerRampSegmentType">
3695                 <xs:annotation>
3696                         <xs:documentation>A Power Ramp Segment describes a change up or down
3697 in units/duration. A ramp rate holds for the duration between beginRamp to
3698 endRamp</xs:documentation>
3699                 </xs:annotation>
3700                 <xs:sequence>
3701                         <xs:element name="rate" type="power:PowerQuantityType"/>
3702                         <xs:element ref="emix:duration"/>
3703                         <xs:element ref="resource:beginRamp"/>
3704                         <xs:element ref="resource:endRamp"/>
3705                         <xs:element ref="emix:integralOnly"/>
3706                 </xs:sequence>
3707         </xs:complexType>
3708         <xs:element name="beginRamp" type="xs:int"/>
3709         <xs:element name="endRamp" type="xs:int"/>
3710         <!-- 4.1.4 Power Ramp Rate -->
3711         <xs:element name="percentRampRate" type="resource:PercentRampRateType"/>
3712         <xs:complexType name="PercentRampRateType">
3713                 <xs:annotation>
3714                         <xs:documentation>Change up or down in percent of total
3715 response.</xs:documentation>
3716                 </xs:annotation>
3717                 <xs:sequence>
3718                         <xs:element ref="resource:rate"/>
3719                         <xs:element ref="emix:duration"/>
3720                 </xs:sequence>
3721         </xs:complexType>
3722         <!-- 4.2 Constraints and Requirements unique to Power Resources-->
3723         <xs:element name="minimumLoad" type="resource:MinimumLoadType"
3724 substitutionGroup="emix:baseConstraint">
3725                 <xs:annotation>
3726                         <xs:documentation>Constraint on Minimum Load that a Resource can
3727 maintain</xs:documentation>
3728                 </xs:annotation>
3729         </xs:element>
3730         <xs:element name="maximumPower" type="resource:MaximumPowerType"
3731 substitutionGroup="emix:baseConstraint">
3732                 <xs:annotation>
3733                         <xs:documentation>Constraint on Maximum Power available from a
3734 resource</xs:documentation>
3735                 </xs:annotation>
3736         </xs:element>
3737         <xs:element name="maximumEnergy" type="resource:MaximumEnergyType"
3738 substitutionGroup="emix:baseConstraint">
3739                 <xs:annotation>
3740                         <xs:documentation>Constraint on Maximum Energy available from a
3741 resource</xs:documentation>
3742                 </xs:annotation>
3743         </xs:element>
3744         <xs:element name="minimumLoadReduction" type="resource:MinimumLoadReductionType"
3745 substitutionGroup="emix:baseConstraint">
3746                 <xs:annotation>
3747                         <xs:documentation>Constraint on Minimum Load Reduction resource can
3748 make</xs:documentation>
3749                 </xs:annotation>
```

```
3750                </xs:element>
3751            <xs:complexType name="MinimumLoadType" mixed="false">
3752                    <xs:annotation>
3753                            <xs:documentation>ype of Constraint on Minimum Load that a Resource
3754    can maintain</xs:documentation>
3755                    </xs:annotation>
3756                    <xs:complexContent mixed="false">
3757                            <xs:extension base="emix:BaseConstraintType">
3758                                    <xs:sequence>
3759                                            <xs:element ref="power:powerQuantity"/>
3760                                    </xs:sequence>
3761                            </xs:extension>
3762                    </xs:complexContent>
3763            </xs:complexType>
3764            <xs:complexType name="MaximumPowerType" mixed="false">
3765                    <xs:annotation>
3766                            <xs:documentation>Type of Constraint on Maximum Power available from a
3767    resource</xs:documentation>
3768                    </xs:annotation>
3769                    <xs:complexContent mixed="false">
3770                            <xs:extension base="emix:BaseConstraintType">
3771                                    <xs:sequence>
3772                                            <xs:element ref="power:powerQuantity" minOccurs="1"
3773    maxOccurs="1"/>
3774                                    </xs:sequence>
3775                            </xs:extension>
3776                    </xs:complexContent>
3777            </xs:complexType>
3778            <xs:complexType name="MaximumEnergyType" mixed="false">
3779                    <xs:annotation>
3780                            <xs:documentation> Type of Constraint on Maximum Energy available from
3781    a resource</xs:documentation>
3782                    </xs:annotation>
3783                    <xs:complexContent mixed="false">
3784                            <xs:extension base="emix:BaseConstraintType">
3785                                    <xs:sequence>
3786                                            <xs:element ref="power:energyQuantity"/>
3787                                    </xs:sequence>
3788                            </xs:extension>
3789                    </xs:complexContent>
3790            </xs:complexType>
3791            <!-- 4.2.5 Minimum Load Reduction -->
3792            <xs:complexType name="MinimumLoadReductionType" mixed="false">
3793                    <xs:annotation>
3794                            <xs:documentation>Minimum units for a load reduction (e.g., MW rating
3795    of a discrete pump)</xs:documentation>
3796                    </xs:annotation>
3797                    <xs:complexContent mixed="false">
3798                            <xs:extension base="emix:BaseConstraintType">
3799                                    <xs:sequence>
3800                                            <xs:element ref="power:powerQuantity" minOccurs="1"
3801    maxOccurs="1"/>
3802                                    </xs:sequence>
3803                            </xs:extension>
3804                    </xs:complexContent>
3805            </xs:complexType>
3806            <!-- 4.3.1 Offer Segment elements  -->
3807            <xs:element name="offerCurve" type="resource:OfferCurveType"
3808    substitutionGroup="emix:baseRequirement"/>
3809            <xs:complexType name="OfferCurveType" mixed="false">
3810                    <xs:annotation>
3811                            <xs:documentation>Type of a collection of Offer Segments used to
3812    compute cost requirements across a range of power.</xs:documentation>
3813                    </xs:annotation>
3814                    <xs:complexContent mixed="false">
3815                            <xs:extension base="emix:BaseRequirementType">
3816                                    <xs:sequence>
3817                                            <xs:element name="offerSegment"
3818    type="resource:OfferSegmentType" maxOccurs="unbounded"/>
3819                                    </xs:sequence>
3820                            </xs:extension>
```

```xml
                    </xs:complexContent>
            </xs:complexType>
            <xs:element name="offerSegment" type="resource:OfferSegmentType"/>
            <xs:complexType name="OfferSegmentType">
                    <xs:annotation>
                            <xs:documentation> Type of Marginal offer for Power within a range.
Marginal costs must be computed within the context of a range of segments as conformed
by the Offer Type</xs:documentation>
                    </xs:annotation>
                    <xs:sequence>
                            <xs:element ref="emix:price"/>
                            <xs:element ref="emix:quantity"/>
                            <xs:element ref="power:powerItem"/>
                            <xs:element ref="emix:integralOnly"/>
                    </xs:sequence>
            </xs:complexType>
            <!-- 4.3.9 Resource ID -->
            <xs:element name="mrid" type="resource:MridType"/>
            <xs:simpleType name="MridType">
                    <xs:annotation>
                            <xs:documentation>multi-part resource  id from the ISO TC57
CIM.</xs:documentation>
                    </xs:annotation>
                    <xs:restriction base="xs:string"/>
            </xs:simpleType>
            <!-- 4.4  Volt-Var Elements -->
            <!-- 4.4.1 VMin -->
            <!-- These are the 4 parts of an inverter.. -->
            <xs:element name="vMin" type="resource:VMinType"/>
            <xs:complexType name="VMinType">
                    <xs:annotation>
                            <xs:documentation>The minimum voltage level of the Voltage Regulation
Service. In IEEE 1547, this represents a voltage level of 88% of nominal voltage for a
photovoltaic (PV) inverter.</xs:documentation>
                    </xs:annotation>
                    <xs:sequence>
                            <xs:element ref="power:voltage"/>
                    </xs:sequence>
            </xs:complexType>
            <!-- 4.4.2 VMax -->
            <xs:element name="vMax" type="resource:VMaxType"/>
            <xs:complexType name="VMaxType">
                    <xs:annotation>
                            <xs:documentation>VMax is the IEEE 1547 maximum voltage level of 110%
of nominal voltage where the PV inverter must disconnect.</xs:documentation>
                    </xs:annotation>
                    <xs:sequence>
                            <xs:element ref="power:voltage"/>
                    </xs:sequence>
            </xs:complexType>
            <!-- 4.4.3 QMax -->
            <xs:element name="qMax" type="resource:QMaxType"/>
            <xs:complexType name="QMaxType">
                    <xs:annotation>
                            <xs:documentation>Qmax is the inverter's var capability and may be
positive (capacitive) or negative (inductive).</xs:documentation>
                    </xs:annotation>
                    <xs:sequence>
                            <xs:element ref="power:varQuantity"/>
                    </xs:sequence>
            </xs:complexType>
            <!-- 4.4.4 volt-var -->
            <xs:element name="pMax" type="resource:PMaxType"/>
            <xs:complexType name="PMaxType">
                    <xs:annotation>
                            <xs:documentation>PMax is the inverter's watt capability and may be
positive or negative. </xs:documentation>
                    </xs:annotation>
                    <xs:sequence>
                            <xs:element ref="power:powerQuantity"/>
                    </xs:sequence>
```

```
3892          </xs:complexType>
3893          <!-- 4.9 Miscelenous Semantic elementsvolt-var -->
3894          <xs:element name="dispatchTime" type="emix:DurationType"/>
3895          <xs:element name="maximumDeliveryRate" type="emix:QuantityType"/>
3896          <xs:element name="minimumDeliveryRate" type="emix:QuantityType"/>
3897          <xs:element name="maximumResponse" type="emix:QuantityType"/>
3898          <xs:element name="minimumResponse" type="emix:QuantityType"/>
3899          <xs:element name="rate" type="emix:QuantityType"/>
3900          <xs:element name="targetRegulation" type="power:PowerAttributesType"/>
3901          <xs:element name="dispatchUp" type="emix:DurationType">
3902                  <xs:annotation>
3903                          <xs:documentation>Time in which resource can respond to a request to
3904      increase energy provided. If zero, no dispatchUp available. Can also be startup delay
3905      for non-spinning reserve.</xs:documentation>
3906                  </xs:annotation>
3907          </xs:element>
3908          <xs:element name="dispatchDown" type="emix:DurationType">
3909                  <xs:annotation>
3910                          <xs:documentation>Time in which resource can respond to a request to
3911      decrease energy provided. If zero, no dispatch Down available.</xs:documentation>
3912                  </xs:annotation>
3913          </xs:element>
3914          <xs:element name="rampTime" type="emix:DurationType">
3915                  <xs:annotation>
3916                          <xs:documentation>Requested ramp time to move from the current
3917      setpoint to the new setpoint</xs:documentation>
3918                  </xs:annotation>
3919          </xs:element>
3920          <xs:element name="window" type="emix:DurationType">
3921                  <xs:annotation>
3922                          <xs:documentation>Time window within which to randomly execute the
3923      command. If the time window is zero, the command will be executed immediately, (If not
3924      included, then default time window for this function will be used)</xs:documentation>
3925                  </xs:annotation>
3926          </xs:element>
3927      </xs:schema>
```

G. An Example

24 Hours of pricing on a full requirements contract.

```xml
<?xml version="1.0" encoding="utf-16"?>
<!--
Jira 274 Price Publication
    emix = EMIXType
    createdDateTime = 2-12-2011 14:00
    transactive State = Tender
    currency = USD

    terms:
    PriceType = absolutePrice

    Gluon: StartTime = 2-13-2001 00:00, Duration = 3600 seconds
    Intervals (0.71,0.21,-0.13, 0.15,0.70,0.86,0.90,1.01,1.12,1.14,1.15,2.74,
                    1.25,1.20,1.29,1.31,1.00,0.99,0.89,0.86,0.79,0.88,0.87,0.76)
-->
<emix:product xmlns:power="http://docs.oasis-open.org/ns/emix/power"
xmlns:emix="http://docs.oasis-open.org/ns/emix" xmlns:xcal="http://docs.oasis-
open.org/ns/ws-calendar/201103" xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance">
    <xcal:properties>
        <xcal:created>
            <xcal:utc-date-time>20110328</xcal:utc-date-time>
        </xcal:created>
    </xcal:properties>
    <xcal:components>
        <xcal:gluon>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>b375a906-64bc-4573-9971-
045b52e30a56@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                            <xcal:text>CHILD</xcal:text>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>cd6de037-1c39-481d-87cf-
8c587df56dfb@examples.oasis-open.org</xcal:uid>
                </xcal:related-to>
                <xcal:dtstart>
                    <xcal:parameters>
                        <xcal:tzid>
                            <xcal:text>America/New_York</xcal:text>
                        </xcal:tzid>
                    </xcal:parameters>
                    <xcal:date-time>20110330T00000000</xcal:date-time>
                </xcal:dtstart>
                <xcal:duration>
                </xcal:duration>
                <xcal:x-wsCalendar-attach>
                    <emix:productDescription
xs:type="power:PowerProductDescription">
                        <power:unitEnergyPrice>
```

```
                                            <emix:priceAbsolute>

<emix:priceEnumeration>0.111</emix:priceEnumeration>
                                            </emix:priceAbsolute>
                                            <power:wattHours>
                                                <emix:scale>#k</emix:scale>
                                            </power:wattHours>
                                        </power:unitEnergyPrice>
                                        <power:Watts>
                                            <emix:scale>#M</emix:scale>
                                            <power:powerAttributes>
                                                <power:hertz>60</power:hertz>
                                                <power:voltage>220</power:voltage>
                                                <power:ac>true</power:ac>
                                            </power:powerAttributes>
                                        </power:Watts>
                                        <power:serviceLocation>
                                            <power:node>xxNode.IDxx</power:node>
                                        </power:serviceLocation>
                                    </emix:productDescription>
                                </xcal:x-wsCalendar-attach>
                        </xcal:properties>
                    </xcal:gluon>
                    <xcal:interval>
                        <xcal:properties>
                            <xcal:uid>
                                <xcal:text>cd6de037-1c39-481d-87cf-
8c587df56dfb@examples.oasis-open.org</xcal:text>
                            </xcal:uid>
                            <xcal:x-wsCalendar-attach>
                                <emix:productDescription
xs:type="power:PowerProductDescription">
                                    <power:unitEnergyPrice>
                                        <emix:priceAbsolute>

<emix:priceEnumeration>0.71</emix:priceEnumeration>
                                        </emix:priceAbsolute>
                                    </power:unitEnergyPrice>
                                </emix:productDescription>
                            </xcal:x-wsCalendar-attach>
                        </xcal:properties>
                    </xcal:interval>
                    <xcal:interval>
                        <xcal:properties>
                            <xcal:uid>
                                <xcal:text>78839070-98bc-42c4-b216-
3665efdeaef4@examples.oasis-open.org</xcal:text>
                            </xcal:uid>
                            <xcal:related-to>
                                <xcal:parameters>
                                    <xcal:reltype>
                                    </xcal:reltype>
                                </xcal:parameters>
                                <xcal:uid>cd6de037-1c39-481d-87cf-
8c587df56dfb@examples.oasis-open.org</xcal:uid>
                            </xcal:related-to>
                            <xcal:x-wsCalendar-attach>
                                <emix:productDescription
xs:type="power:PowerProductDescription">
                                    <power:unitEnergyPrice>
                                        <emix:priceAbsolute>

<emix:priceEnumeration>0.21</emix:priceEnumeration>
```

```
4049                              </emix:priceAbsolute>
4050                          </power:unitEnergyPrice>
4051                      </emix:productDescription>
4052                  </xcal:x-wsCalendar-attach>
4053              </xcal:properties>
4054          </xcal:interval>
4055          <xcal:interval>
4056              <xcal:properties>
4057                  <xcal:uid>
4058                      <xcal:text>3fbb3ccb-a38e-43b2-968b-
4059  0117c57a1d24@examples.oasis-open.org</xcal:text>
4060                  </xcal:uid>
4061                  <xcal:related-to>
4062                      <xcal:parameters>
4063                          <xcal:reltype>
4064                          </xcal:reltype>
4065                      </xcal:parameters>
4066                      <xcal:uid>78839070-98bc-42c4-b216-
4067  3665efdeaef4@examples.oasis-open.org</xcal:uid>
4068                  </xcal:related-to>
4069                  <xcal:x-wsCalendar-attach>
4070                      <emix:productDescription
4071  xs:type="power:PowerProductDescription">
4072                          <power:unitEnergyPrice>
4073                              <emix:priceAbsolute>
4074                                  <emix:priceEnumeration>-
4075  0.13</emix:priceEnumeration>
4076                              </emix:priceAbsolute>
4077                          </power:unitEnergyPrice>
4078                      </emix:productDescription>
4079                  </xcal:x-wsCalendar-attach>
4080              </xcal:properties>
4081          </xcal:interval>
4082          <xcal:interval>
4083              <xcal:properties>
4084                  <xcal:uid>
4085                      <xcal:text>6a5d23b2-0aa9-4309-b2d3-
4086  b04f630add99@examples.oasis-open.org</xcal:text>
4087                  </xcal:uid>
4088                  <xcal:related-to>
4089                      <xcal:parameters>
4090                          <xcal:reltype>
4091                          </xcal:reltype>
4092                      </xcal:parameters>
4093                      <xcal:uid>3fbb3ccb-a38e-43b2-968b-
4094  0117c57a1d24@examples.oasis-open.org</xcal:uid>
4095                  </xcal:related-to>
4096                  <xcal:x-wsCalendar-attach>
4097                      <emix:productDescription
4098  xs:type="power:PowerProductDescription">
4099                          <power:unitEnergyPrice>
4100                              <emix:priceAbsolute>
4101
4102  <emix:priceEnumeration>0.15</emix:priceEnumeration>
4103                              </emix:priceAbsolute>
4104                          </power:unitEnergyPrice>
4105                      </emix:productDescription>
4106                  </xcal:x-wsCalendar-attach>
4107              </xcal:properties>
4108          </xcal:interval>
4109          <xcal:interval>
4110              <xcal:properties>
4111                  <xcal:uid>
```

```
                                    <xcal:text>5b0b2104-f2bd-4b7b-819c-
ab970b29f668@examples.oasis-open.org</xcal:text>
                                </xcal:uid>
                                <xcal:related-to>
                                    <xcal:parameters>
                                        <xcal:reltype>
                                        </xcal:reltype>
                                    </xcal:parameters>
                                    <xcal:uid>6a5d23b2-0aa9-4309-b2d3-
b04f630add99@examples.oasis-open.org</xcal:uid>
                                </xcal:related-to>
                                <xcal:x-wsCalendar-attach>
                                    <emix:productDescription
xs:type="power:PowerProductDescription">
                                        <power:unitEnergyPrice>
                                            <emix:priceAbsolute>

<emix:priceEnumeration>0.70</emix:priceEnumeration>
                                            </emix:priceAbsolute>
                                        </power:unitEnergyPrice>
                                    </emix:productDescription>
                                </xcal:x-wsCalendar-attach>
                            </xcal:properties>
                        </xcal:interval>
                        <xcal:interval>
                            <xcal:properties>
                                <xcal:uid>
                                    <xcal:text>0270f6af-56bc-4d9c-a7c2-
bebcfe3bc0ea@examples.oasis-open.org</xcal:text>
                                </xcal:uid>
                                <xcal:related-to>
                                    <xcal:parameters>
                                        <xcal:reltype>
                                        </xcal:reltype>
                                    </xcal:parameters>
                                    <xcal:uid>5b0b2104-f2bd-4b7b-819c-
ab970b29f668@examples.oasis-open.org</xcal:uid>
                                </xcal:related-to>
                                <xcal:x-wsCalendar-attach>
                                    <emix:productDescription
xs:type="power:PowerProductDescription">
                                        <power:unitEnergyPrice>
                                            <emix:priceAbsolute>

<emix:priceEnumeration>0.86</emix:priceEnumeration>
                                            </emix:priceAbsolute>
                                        </power:unitEnergyPrice>
                                    </emix:productDescription>
                                </xcal:x-wsCalendar-attach>
                            </xcal:properties>
                        </xcal:interval>
                        <xcal:interval>
                            <xcal:properties>
                                <xcal:uid>
                                    <xcal:text>3c36a01d-1229-4f7f-86dc-
31728bfaba6d@examples.oasis-open.org</xcal:text>
                                </xcal:uid>
                                <xcal:related-to>
                                    <xcal:parameters>
                                        <xcal:reltype>
                                        </xcal:reltype>
                                    </xcal:parameters>
```

```
                        <xcal:uid>0270f6af-56bc-4d9c-a7c2-
bebcfe3bc0ea@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>0.90</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>0d96802b-bf0f-41e6-8d55-
804598879be0@examples.oasis-open.org</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>3c36a01d-1229-4f7f-86dc-
31728bfaba6d@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>1.01</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>a843d8d0-28f8-4a31-a7ee-
25b14c701036@examples.oasis-open.org</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>0d96802b-bf0f-41e6-8d55-
804598879be0@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>
```

```
4236
4237        <emix:priceEnumeration>1.12</emix:priceEnumeration>
4238                            </emix:priceAbsolute>
4239                        </power:unitEnergyPrice>
4240                    </emix:productDescription>
4241                </xcal:x-wsCalendar-attach>
4242            </xcal:properties>
4243        </xcal:interval>
4244        <xcal:interval>
4245            <xcal:properties>
4246                <xcal:uid>
4247                    <xcal:text>2097d7d8-f554-469c-8c92-
4248 9eb8551c086d@examples.oasis-open.org</xcal:text>
4249                </xcal:uid>
4250                <xcal:related-to>
4251                    <xcal:parameters>
4252                        <xcal:reltype>
4253                        </xcal:reltype>
4254                    </xcal:parameters>
4255                    <xcal:uid>a843d8d0-28f8-4a31-a7ee-
4256 25b14c701036@examples.oasis-open.org</xcal:uid>
4257                </xcal:related-to>
4258                <xcal:x-wsCalendar-attach>
4259                    <emix:productDescription
4260 xs:type="power:PowerProductDescription">
4261                        <power:unitEnergyPrice>
4262                            <emix:priceAbsolute>
4263
4264 <emix:priceEnumeration>1.14</emix:priceEnumeration>
4265                            </emix:priceAbsolute>
4266                        </power:unitEnergyPrice>
4267                    </emix:productDescription>
4268                </xcal:x-wsCalendar-attach>
4269            </xcal:properties>
4270        </xcal:interval>
4271        <xcal:interval>
4272            <xcal:properties>
4273                <xcal:uid>
4274                    <xcal:text>f2f630f6-f673-47ae-8a89-
4275 3d2b23f379c6@examples.oasis-open.org</xcal:text>
4276                </xcal:uid>
4277                <xcal:related-to>
4278                    <xcal:parameters>
4279                        <xcal:reltype>
4280                        </xcal:reltype>
4281                    </xcal:parameters>
4282                    <xcal:uid>2097d7d8-f554-469c-8c92-
4283 9eb8551c086d@examples.oasis-open.org</xcal:uid>
4284                </xcal:related-to>
4285                <xcal:x-wsCalendar-attach>
4286                    <emix:productDescription
4287 xs:type="power:PowerProductDescription">
4288                        <power:unitEnergyPrice>
4289                            <emix:priceAbsolute>
4290
4291 <emix:priceEnumeration>1.15</emix:priceEnumeration>
4292                            </emix:priceAbsolute>
4293                        </power:unitEnergyPrice>
4294                    </emix:productDescription>
4295                </xcal:x-wsCalendar-attach>
4296            </xcal:properties>
4297        </xcal:interval>
4298        <xcal:interval>
```

```
4299                        <xcal:properties>
4300                            <xcal:uid>
4301                                <xcal:text>1d9ad168-433e-4504-97f5-
4302    05a8df1d1f20@examples.oasis-open.org</xcal:text>
4303                            </xcal:uid>
4304                            <xcal:related-to>
4305                                <xcal:parameters>
4306                                    <xcal:reltype>
4307                                    </xcal:reltype>
4308                                </xcal:parameters>
4309                                <xcal:uid>f2f630f6-f673-47ae-8a89-
4310    3d2b23f379c6@examples.oasis-open.org</xcal:uid>
4311                            </xcal:related-to>
4312                            <xcal:x-wsCalendar-attach>
4313                                <emix:productDescription
4314    xs:type="power:PowerProductDescription">
4315                                    <power:unitEnergyPrice>
4316                                        <emix:priceAbsolute>
4317
4318    <emix:priceEnumeration>2.74</emix:priceEnumeration>
4319                                        </emix:priceAbsolute>
4320                                    </power:unitEnergyPrice>
4321                                </emix:productDescription>
4322                            </xcal:x-wsCalendar-attach>
4323                        </xcal:properties>
4324                    </xcal:interval>
4325                    <xcal:interval>
4326                        <xcal:properties>
4327                            <xcal:uid>
4328                                <xcal:text>f65aa213-3176-47e3-88c3-
4329    0f7dbe0ec99e@examples.oasis-open.org</xcal:text>
4330                            </xcal:uid>
4331                            <xcal:related-to>
4332                                <xcal:parameters>
4333                                    <xcal:reltype>
4334                                    </xcal:reltype>
4335                                </xcal:parameters>
4336                                <xcal:uid>1d9ad168-433e-4504-97f5-
4337    05a8df1d1f20@examples.oasis-open.org</xcal:uid>
4338                            </xcal:related-to>
4339                            <xcal:x-wsCalendar-attach>
4340                                <emix:productDescription
4341    xs:type="power:PowerProductDescription">
4342                                    <power:unitEnergyPrice>
4343                                        <emix:priceAbsolute>
4344
4345    <emix:priceEnumeration>1.25</emix:priceEnumeration>
4346                                        </emix:priceAbsolute>
4347                                    </power:unitEnergyPrice>
4348                                </emix:productDescription>
4349                            </xcal:x-wsCalendar-attach>
4350                        </xcal:properties>
4351                    </xcal:interval>
4352                    <xcal:interval>
4353                        <xcal:properties>
4354                            <xcal:uid>
4355                                <xcal:text>920c3e26-cea2-4456-84a7-
4356    9f5f369cf491@examples.oasis-open.org</xcal:text>
4357                            </xcal:uid>
4358                            <xcal:related-to>
4359                                <xcal:parameters>
4360                                    <xcal:reltype>
4361                                    </xcal:reltype>
```

```
                        </xcal:parameters>
                        <xcal:uid>f65aa213-3176-47e3-88c3-
0f7dbe0ec99e@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>1.20</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>bd51713d-c84c-4a18-b641-
65c37ec2728d@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>920c3e26-cea2-4456-84a7-
9f5f369cf491@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>1.29</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>2482a65c-bd0c-4a33-b777-
cf89d637d1b8@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>bd51713d-c84c-4a18-b641-
65c37ec2728d@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>
```

```
4425
4426                <emix:priceEnumeration>1.31</emix:priceEnumeration>
4427                                    </emix:priceAbsolute>
4428                                </power:unitEnergyPrice>
4429                            </emix:productDescription>
4430                        </xcal:x-wsCalendar-attach>
4431                    </xcal:properties>
4432                </xcal:interval>
4433                <xcal:interval>
4434                    <xcal:properties>
4435                        <xcal:uid>
4436                            <xcal:text>01063041-4d93-46a6-9fd6-
4437    7b4f92a938e0@examples.oasis-open.org</xcal:text>
4438                        </xcal:uid>
4439                        <xcal:related-to>
4440                            <xcal:parameters>
4441                                <xcal:reltype>
4442                                </xcal:reltype>
4443                            </xcal:parameters>
4444                            <xcal:uid>2482a65c-bd0c-4a33-b777-
4445    cf89d637d1b8@examples.oasis-open.org</xcal:uid>
4446                        </xcal:related-to>
4447                        <xcal:x-wsCalendar-attach>
4448                            <emix:productDescription
4449    xs:type="power:PowerProductDescription">
4450                                <power:unitEnergyPrice>
4451                                    <emix:priceAbsolute>
4452
4453    <emix:priceEnumeration>1.00</emix:priceEnumeration>
4454                                    </emix:priceAbsolute>
4455                                </power:unitEnergyPrice>
4456                            </emix:productDescription>
4457                        </xcal:x-wsCalendar-attach>
4458                    </xcal:properties>
4459                </xcal:interval>
4460                <xcal:interval>
4461                    <xcal:properties>
4462                        <xcal:uid>
4463                            <xcal:text>2bb61d4e-a3ea-423a-98ef-
4464    395e25b35674@examples.oasis-open.org</xcal:text>
4465                        </xcal:uid>
4466                        <xcal:related-to>
4467                            <xcal:parameters>
4468                                <xcal:reltype>
4469                                </xcal:reltype>
4470                            </xcal:parameters>
4471                            <xcal:uid>01063041-4d93-46a6-9fd6-
4472    7b4f92a938e0@examples.oasis-open.org</xcal:uid>
4473                        </xcal:related-to>
4474                        <xcal:x-wsCalendar-attach>
4475                            <emix:productDescription
4476    xs:type="power:PowerProductDescription">
4477                                <power:unitEnergyPrice>
4478                                    <emix:priceAbsolute>
4479
4480    <emix:priceEnumeration>0.99</emix:priceEnumeration>
4481                                    </emix:priceAbsolute>
4482                                </power:unitEnergyPrice>
4483                            </emix:productDescription>
4484                        </xcal:x-wsCalendar-attach>
4485                    </xcal:properties>
4486                </xcal:interval>
4487                <xcal:interval>
```

```
                    <xcal:properties>
                        <xcal:uid>
                            <xcal:text>b50081d2-d7f8-4ab3-9881-
f730ccfa76b4@examples.oasis-open.org</xcal:text>
                        </xcal:uid>
                        <xcal:related-to>
                            <xcal:parameters>
                                <xcal:reltype>
                                </xcal:reltype>
                            </xcal:parameters>
                            <xcal:uid>2bb61d4e-a3ea-423a-98ef-
395e25b35674@examples.oasis-open.org</xcal:uid>
                        </xcal:related-to>
                        <xcal:x-wsCalendar-attach>
                            <emix:productDescription
xs:type="power:PowerProductDescription">
                                <power:unitEnergyPrice>
                                    <emix:priceAbsolute>

<emix:priceEnumeration>0.89</emix:priceEnumeration>
                                    </emix:priceAbsolute>
                                </power:unitEnergyPrice>
                            </emix:productDescription>
                        </xcal:x-wsCalendar-attach>
                    </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                    <xcal:properties>
                        <xcal:uid>
                            <xcal:text>49c38d29-e377-4c21-8089-
6fb8919fefd2@examples.oasis-open.org</xcal:text>
                        </xcal:uid>
                        <xcal:related-to>
                            <xcal:parameters>
                                <xcal:reltype>
                                </xcal:reltype>
                            </xcal:parameters>
                            <xcal:uid>b50081d2-d7f8-4ab3-9881-
f730ccfa76b4@examples.oasis-open.org</xcal:uid>
                        </xcal:related-to>
                        <xcal:x-wsCalendar-attach>
                            <emix:productDescription
xs:type="power:PowerProductDescription">
                                <power:unitEnergyPrice>
                                    <emix:priceAbsolute>

<emix:priceEnumeration>0.86</emix:priceEnumeration>
                                    </emix:priceAbsolute>
                                </power:unitEnergyPrice>
                            </emix:productDescription>
                        </xcal:x-wsCalendar-attach>
                    </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                    <xcal:properties>
                        <xcal:uid>
                            <xcal:text>22ed71e5-7ab2-4fae-ac7a-
f02f7496b391@examples.oasis-open.org</xcal:text>
                        </xcal:uid>
                        <xcal:related-to>
                            <xcal:parameters>
                                <xcal:reltype>
                                </xcal:reltype>
```

```
                        </xcal:parameters>
                        <xcal:uid>49c38d29-e377-4c21-8089-
6fb8919fefd2@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>0.79</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>60cc5194-9c26-4ee8-a75f-
02d476e1361d@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>22ed71e5-7ab2-4fae-ac7a-
f02f7496b391@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>

<emix:priceEnumeration>0.88</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>3d0b45e4-382b-4964-8779-
1b7b8c3d7133@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>60cc5194-9c26-4ee8-a75f-
02d476e1361d@examples.oasis-open.org</xcal:uid>
                    </xcal:related-to>
                    <xcal:x-wsCalendar-attach>
                        <emix:productDescription
xs:type="power:PowerProductDescription">
                            <power:unitEnergyPrice>
                                <emix:priceAbsolute>
```

```
4614
4615            <emix:priceEnumeration>0.87</emix:priceEnumeration>
4616                            </emix:priceAbsolute>
4617                        </power:unitEnergyPrice>
4618                    </emix:productDescription>
4619                </xcal:x-wsCalendar-attach>
4620            </xcal:properties>
4621        </xcal:interval>
4622        <xcal:interval>
4623            <xcal:properties>
4624                <xcal:uid>
4625                    <xcal:text>9e89696d-6511-427b-9086-
4626  b595ef25f23c@examples.oasis-open.org</xcal:text>
4627                </xcal:uid>
4628                <xcal:related-to>
4629                    <xcal:parameters>
4630                        <xcal:reltype>
4631                        </xcal:reltype>
4632                    </xcal:parameters>
4633                    <xcal:uid>3d0b45e4-382b-4964-8779-
4634  1b7b8c3d7133@examples.oasis-open.org</xcal:uid>
4635                </xcal:related-to>
4636                <xcal:x-wsCalendar-attach>
4637                    <emix:productDescription
4638  xs:type="power:PowerProductDescription">
4639                        <power:unitEnergyPrice>
4640                            <emix:priceAbsolute>
4641
4642  <emix:priceEnumeration>0.76</emix:priceEnumeration>
4643                            </emix:priceAbsolute>
4644                        </power:unitEnergyPrice>
4645                    </emix:productDescription>
4646                </xcal:x-wsCalendar-attach>
4647            </xcal:properties>
4648        </xcal:interval>
4649    </xcal:components>
4650    <emix:currency>USD</emix:currency>
4651    <emix:marketContext>market.context@examples.com</emix:marketContext>
4652    <emix:transactiveState>Tender</emix:transactiveState>
4653  </emix:product>
```

H. Revision History

Revision	Date	Editor	Changes Made
WD01	2009-12-08	Toby Considine	Initial Draft from templates and outline
WD02	2010-01-12	William Cox	Inserted information model details from TC discussions
WD03	2010-03-10	William Cox	Change to envelope and certificate metaphor. Changes in mandatory and optional definitions.
WD04	2010-03-24	William Cox	Updates based on TC comments and corrections. Additional open issues in TC agenda.
WD05	2010-05-18	Toby Considine	Aligned elements with current draft if WS-Calendar, cleaned up some language to align with the last two months of conversation. Extended envelop and intrinsic/extrinsic language
WD06	2010-05-21	Toby Considine	Began incorporating TeMIX language. Changed Certificates to Warrants. Fleshed out Energy Artifacts
WD07	2010-07-07	Toby Considine	Incorporated Aaron Snyder's extensive re-write into Power & Energy section
WD08	2010-08-10	Toby Considine	Extensive re-write for narrative quality, responded to first 52 comments, Updated to include WS-Calendar WD08 language, added tables of table, examples
WD09	2010-08-18	Toby Considine	Incorporated recent WS-Calendar changes to update Products. Added explanation of WS-Calendar. Cleaned up double entry of Partitions.
WD10	2010-08-30	Toby Considine	Reduced argumentation in intro, excluded WS-Calendar re-writes, pointed to WS-Calendar appendices. Merged AC and DC
WD11	2010-09-05	Toby Considine	Distinguished between Intrinsic elements and Generic Product, incorporated inheritance language into GP, Re-created T&D as a much smaller Transport Artifact, changed envelope language to face and contents.
WD12	2010-10-26	Toby Considine	Responded to many Jira comments. Re-created T&D as a much smaller Transport Artifact, changed envelope language to face and contents. Responded to many Jira comments. Descriptions now based on WD12

Revision	Date	Editor	Changes Made
			Schema.
WD13	2010-11-01	Toby Considine Ed Cazalet Dave Holmberg	Removed repetitive discussion of WS-Calendar objects. Reflect new use of WS-Calendar Sequence in Schema. Recast Options to describe reserves.
WD14	2010-11-09	Toby Considine Ed Cazalet	Changes to resources, block power, misc. tightening of document
WD15	2010-11-14	Toby Considine Ed Cazalet Sean Crimmins	EMIX Sequence changed to EMIX Base. General tightening. Addition of Load and Power Offers, including 3-part bids for each.
CSD01	2010-11-15	Toby Considine	Minor changes as per comments
WD16	2011-01-15	Toby Considine	46 Minor issues from PR01 Adopted new WD format Moved namespaces into section 1 Adjusted duplicate table names Fixed section numbering anomalies
WD17	2011-02-08	Toby Considine	Issue Resolution. See Release Notes from Jira
WD18	2011-03-07	Toby Considine	Numerous Jira Issues, (see release notes), Significant Schema work: Resources as dicussed, General EMIX constraints and requirements now in Core EMIX namespace, but isolated in requirements.xsd. Added schedule constraints as optional constraint
WD19	2011-03-17	Toby Considine	Tightened language, some egregious errors and references not found removed
WD20	2011-03022	Toby Considine	Simplified Tables, Added NAESB appendix, updated schemas in appendix
WD21	2011-0323	Toby Considine	Quick Pass for show-stoppers, Purged last 16 uses of EMIXTerms for EMIX Base,
WD22	2011-0329	Toby Considine	Minor edits and comments from Jira. Made explicit relations between Base, Product Description, Items, Interfaces, and all derived extensions
WD23	2011-0411	Toby Considine	Extensive review and re-write to consolidate changes as logged in Jira

4656