~~OASIS~~

OASIS

Energy Market Information Exchange (EMIX) Version 1.0

Committee Specification Draft ~~01 /~~ 02 / Public Review Draft ~~01~~02

~~15 November 2010~~

28 April 2011

Specification URIs:

This ~~Version~~version:

~~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.doc (Authoritative)~~

http://docs.oasis-open.org/emix/emix/v1.0/csprd02/emix-v1.0-csprd02.doc (Authoritative)

http://docs.oasis-open.org/emix/emix/v1.0/csprd02/emix-v1.0-csprd02.html

http://docs.oasis-open.org/emix/emix/v1.0/csprd02/emix-v1.0-csprd02.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

~~Previous Version:~~

~~N/A~~

Latest ~~Version~~version:

~~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.doc~~

~~http://docs.oasis-open.org/emix/emix/v1.0/emix-v1.0.pdf~~

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):

~~Ed Cazalet,~~

~~William T. Cox~~

William Cox, Individual

Edward Cazalet, Individual

Editor(s):

~~Toby Considine~~

Style Definition: Normal

Style Definition: TOC 7

Style Definition: TOC 4

Style Definition: TOC 5

Style Definition: TOC 6

Style Definition: Caption: Left, Don't keep with next

Formatted: Bottom: 0.56", Footer distance from edge: 0.7"

Formatted: Subtitle, Space After: 0 pt, Border: Top: (No border)

Formatted: Font: Bold, Font color: Custom Color(RGB(59,0,111))

Formatted: Font: Bold, Font color: Custom Color(RGB(59,0,111))

Formatted: Font color: Custom Color(RGB(59,0,111))

Formatted: Font color: Custom Color(RGB(0,0,238))

Formatted: Font color: Auto

Formatted: Font: Bold, Font color: Custom Color(RGB(59,0,111))

Formatted: Font: Bold, Font color: Custom Color(RGB(59,0,111))

Formatted: Font color: Custom Color(RGB(59,0,111))

Formatted: Default Paragraph Font

Formatted: Font: 8 pt

Toby Considine, University of North Carolina at Chapel Hill

Related work:

~~This specification replaces or supersedes:~~

- ~~N/A~~

This specification is related to:

- ~~OASIS Specification WS-Calendar V1.0, in process~~
- ~~OASIS Specification Energy Interoperation V1.0, in process~~
- OASIS Specification WS-Calendar V1.0, in process
- OASIS Specification Energy Interoperation V1.0, in process
- XML schema(s): emix/v1.0/csprd02/xsd/

Declared XML ~~Namespace~~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/contract~~
~~http://docs.oasis-open.org/ns/emix/power/quality~~
http://docs.oasis-open.org/ns/emix/power/resource
~~http://docs.oasis-open.org/ns/emix/power/transport~~

Abstract:

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

Formatted: Related Work, Indent: First line: 0.5"

Formatted: Bulleted + Level: 1 + Aligned at: 1" + Indent at: 1.5"

Formatted: Font: 8 pt

- 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. ~~The TC will coordinate with others to ensure that commonly used market and communication models are supported.~~

Status:

This document was last revised or approved by the OASIS Energy Market Information Exchange ~~Technical Committee~~(EMIX) TC on the above date. The level of approval is also listed above. Check the "Latest ~~Version" or "Latest Approved~~ 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"~~"Send A Comment" button on the Technical Committee's web page at ~~http://www.oasis-open.org/committees/emix/~~http://www.oasis-open.org/committees/emix/.

Formatted: Font color: Black

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.~~http://www.oasis-open.org/committees/emix/ipr.php).

Formatted: Hyperlink

Citation ~~Format~~format:

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

[EMIX-v1.0 ~~OASIS Committee Specification Draft 01, ~~**]**

Energy Market Information Exchange (EMIX) Version 1.0~~, November 2010. http://docs.oasis-open.org/emix/emix/v1.0/csd01/emix-v1.0-csd01.doc~~. 28 April, 2011. OASIS Committee Specification Draft 02 / Public Review Draft 02. http://docs.oasis-open.org/emix/emix/v1.0/csprd02/emix-v1.0-csprd02.doc

Formatted: Font: Not Italic

Formatted: Abstract

Notices

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~~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,~~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.

Field Code Changed

Formatted: Font: 8 pt

Table of Contents

Formatted: Font: 8 pt

Formatted: Font: 8 pt

Formatted: Font: 8 pt

Formatted: Font: Bold, Font color: Auto

Formatted: Font: 8 pt

Tables, Figures & Examples

Formatted: Notices, Space After: 0 pt, No page break before, Border: Top: (No border)

Index to Figures

Index to ~~Figures~~Tables

Formatted: Subtitle

~~Index to Tables~~

Formatted: Font: 8 pt

Formatted: Font: 8 pt

Formatted: Plain Text

Formatted: Font: 8 pt

1 Introduction

This document defines ~~a set of messages~~an information model, to ~~communicate~~exchange Price and Product ~~definition~~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 Energy Market Information Exchange~~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 ~~information model~~ **[XML Schema]** for Price and Product definition ~~using the Unified Modeling Language~~ ~~**[UML]**~~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).

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

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Font: 10 pt, Not Bold, Font color: Black

Formatted: Font: 10 pt, Font color: Black

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

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

1.3 Normative References

RFC2119 S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.

~~**RFC5545** B. Desruisseaux *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*, http://www.ietf.org/rfc/rfc5545.txt, IETF RFC 5545, September 2009.~~

~~**Calendar Product Schema** C. Joy, C. Daboo, M Douglas, *Schema for representing Products for calendaring and scheduling services*, http://tools.ietf.org/html/draft-cal-Product-schema-00, (Internet-Draft), April 2010.~~

~~**CEFACT** Currency codes, e.g. USD or GBP. Add full reference citation to CEFACT or UBL profile of CEFACT~~

~~**Stoft** S. Stoft, *Power System Economics: Designing Markets for Electricity*. Piscataway, NJ: IEEE Press, 2002.~~

CEFACT *United Nations Centre for Trade Facilitation and Electronic Business*, Currency codes, ISO 4217 3A - Code List Schema Module http://www.unece.org/uncefact/codelist/standard/ISO_ISO3AlphaCurrencyCode_20100407.xsd

GML L van den Brink, C Portele, P. Vretanos *Geography Markup Language (GML) simple features profile*, OpenGIS® Implementation Standard, GML 3.2 Profile, Version 2.0, October 2010, http://schemas.opengis.net/gml/3.2.1/gml.xsd

SOA-RM M MacKenzie, K Laskey, F McCabe, P Brown, R Metz, *OASIS Reference Model for Service Oriented Architecture 1.0*, October 2006 http://docs.oasis-open.org/soa-rm/v1.0/

UML *Unified Modeling Language (UML), Version 2.2*, Object Management Group, February, 2009. ~~http://www.omg.org/technology/documents/formal/uml.htm~~ http://www.omg.org/spec/UML/2.2/

~~**WS-Calendar** OASIS WS-Calendar Technical Committee, specification in progress~~

~~**xCal** C. Daboo, M Douglas, S Lees *xCal: The XML format for iCalendar*, http://tools.ietf.org/html/draft-daboo-et-al-icalendar-in-xml-05, Internet-Draft, April 2010.~~

~~**XLINK** *XML Linking Language (XLink) Version 1.1. S DeRose, E Maler, D Orchard, N Walsh, http://www.w3.org/TR/xlink11/ May 2010.*~~

~~**XPOINTER** S DeRose, E Maler, R Daniel Jr. *XPointer xpointer Scheme*, http://www.w3.org/TR/xptr-xpointer/ December 2002.~~

~~**XML Schema**~~ **URI** T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifier (URI): Generic Syntax*, http://www.ietf.org/rfc/rfc3986.txt, January 2005

WS-Calendar T. Considine, M. Douglas, *OASIS WS-Calendar Public Review Draft 02*, April 2011, specification in progress, http://docs.oasis-open.org/ws-calendar/ws-calendar-spec/v1.0/csprd02/ws-calendar-spec-v1.0-csprd02.pdf

XML Schema H. Thompson, D Beech, M Maloney, N Mendelsohn, *XML Schema Part 1: Structures Second Edition*, http://www.w3.org/TR/xmlschema-1/ October 2004

PV Biron, A Malhotra, *XML Schema Part 2: Datatypes Second Edition*, http://www.w3.org/TR/xmlschema-2/ October 2004.

1.4 Non-Normative References

EISA Energy Independence and Security Act (EISA~~), online. Link retrieved 06/23/2010: http://www.nist.gov/smartgrid/upload/EISA-Energy-bill-110-140-TITLE-XIII.pdf~~ 2007) http://www.gpo.gov/fdsys/pkg/PLAW-110publ140/content-detail.html

~~**FIX** **The FIX protocol (need formal reference)**~~

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 10 pt, Not Bold, Font color: Auto

Formatted: Font: 10 pt, Not Bold, Font color: Auto

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

FIX Financial Information eXchange (FIX) Protocol, http://www.fixprotocol.org/specifications/FIX.5.0SP2

IEC TC57 ~~The home of the~~ IEC TC 57 ~~is~~Power and Load Management, http://tc57.iec.ch/index-tc57.html ~~(link retrieved 06/23/2010)~~

~~**ISO20022** **International Standards Organization, ISO 20022 (need full reference)**~~

NIEM NIEM Technical Architecture Committee (NTAC), *National Information Exchange Model Naming and Design Rules v1.3*, October 2008, http://www.niem.gov/pdf/NIEM-NDR-1-3.pdf

TeMIX Transactional Energy Market Information Exchange [TeMIX] an approved White Paper of the EMIX TC. Ed Cazalet et al. http://www.oasis-open.org/committees/download.php/37954/TeMIX-20100523.pdf

NAESB 03 *Requirements Specification for Common Electricity Product and Pricing Definition*, North American Energy Standards Board [NAESB], March, 2010 (Public Review Draft). http://naesb.org/pdf4/weq_2010_ap6a_retail_2010_ap9a_rec.doc

NIST Roadmap NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0, online. ~~Link retrieved 06/23/1010: http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf~~http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf

PAP03 Details of PAP03 may be found at http://collaborate.nist.gov/twiki-sggrid/bin/view/SmartGrid/PAP03PriceProduct (link retrieved 06/23/2010)

~~**White Paper on WS-Calendar** Link to final paper here.~~

RFC5545 B. Desruisseaux *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*, http://www.ietf.org/rfc/rfc5545.txt, IETF RFC 5545, September 2009.

White Paper on WS-Calendar http://docs.oasis-open.org/ws-calendar/ws-calendar/v1.0/CD01/WS-Calendar-Conceptual-Overview-CD01.pdf

1.5 Namespace

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

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

Formatted: Hyperlink, Font: 10 pt, Not Bold, Font color: Black

Formatted: Hyperlink, Font: 10 pt, Not Bold, Font color: Black

Formatted: Hyperlink, Font: 10 pt, Not Bold, Font color: Black

Formatted: Font: 10 pt, Not Bold, Font color: Auto

Formatted: Font: 8 pt

~~1.5~~1.6 Naming Conventions

This specification generally follows ~~some~~the follows the National Information Exchange Model **[NIEM]** naming ~~conventions~~and design rules for artifacts ~~defined by~~defining the specification, as follows:

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

```
<element name="componentTypecomponentService"
type="energyinterop:ComponentTypeemix:ComponentServiceType"/>
```

For the names of types within XSD files, the names follow the ~~lower~~Upper CamelCase convention with all names starting with ~~a lower~~an upper case letter ~~prefixed by "type-".~~postfixed with "Type". For example,

```
<complexType name="type-componentServiceComponentServiceType">
```

~~For the names of intents, the names follow the lower camelCase convention, with all names starting with a lower case letter, EXCEPT for cases where the intent represents an established acronym, in which case the entire name is in upper case.~~

~~An example of an intent that is an acronym is the "SOAP" intent.~~

~~1.6~~1.7 Editing Conventions

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

~~All elements in the tables not marked as "optional" are mandatory.~~

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

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

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.

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

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. ~~Lack of standardized of messages~~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 ~~political~~not strictly economic processes. These tariffs convey the price and product information to ~~making~~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, ~~contracts~~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 ~~OASIS Energy Market Information Exchange Technical Committee (~~EMIX TC~~)~~ has prepared a white paper which ~~paper~~ 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.

~~Energy~~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.

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 10 pt, Font color: Auto

Formatted: Font: 8 pt

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

Throughout this work, we refer to the intrinsic and extrinsic properties of an energy product. An intrinsic property is one *"belonging to a thing by its very nature."*[1] An extrinsic property is one *"not forming an essential part of a thing or arising or originating from the outside."*[2] In EMIX, the term intrinsic properties refers to those that can be measured and / or verified at the point of delivery, i.e., electric power and price. The term extrinsic properties refers to those that can only be known with prior knowledge, such as the carbon cost, the energy source, or the sulfate load from generation.

EMIX ~~messages~~artifacts communicate both intrinsic and extrinsic properties; extrinsic properties must be able to clear in the market just as ~~does~~do intrinsic ~~energy~~properties.

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

> *The purpose of using a capability is to realize one or more real world effects. At its core, an interaction is "an act" as opposed to "an object" and the result of an interaction is an effect (or a set/series of effects). This effect may be the return of information or the change in the state of entities (known or unknown) that are involved in the interaction.*
>
> *We are careful to distinguish between public actions and private actions; private actions are inherently unknowable by other parties. On the other hand, public actions result in changes to the state that is shared between at least those involved in the current execution context and possibly shared by others. Real world effects are, then, couched in terms of changes to this shared state*

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

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

---

[1] ~~http://wordnet.princeton.edu/~~

[2] ~~Ibid~~

Formatted: Font: 8 pt

2.3 Information Structure

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

On the face of the envelope, EMIX lists the intrinsic qualities of the energy product. In the simplest model, the intrinsic qualities are limited to the price and the information a meter can provide. In a market of homogenous energy sources and commodity energy, only the intrinsic qualities are actionable. In postal handling, information on the face is meant for high-speed automated processing. The simplest devices, including the proverbial smart toaster[3]~~, may understand only the intrinsic qualities.~~, may understand only the intrinsic qualities. The phrase "prices to devices" is used in energy policy discussions to describe a market model in which energy use decisions are distributed to each device that uses energy. Under this model, decisions about whether to use energy now or delay energy use until later are best made where the value is received for that energy use, the end device. The smart toaster is shorthand for the smallest, least capable device that can receive such a message. The Committee anticipates that the information on the face of the envelope will be sufficient for many if not most energy decisions.

The envelope contents are the supporting documents that explain and ~~justify~~support the price for the intrinsic qualities~~.~~ on the face of the envelope. These extrinsic qualities are separable from the intrinsic transaction and perhaps can be traded in secondary markets. The contents can include Warrants about the source and the environmental attributes which provide information about the energy, but they are not the energy. The extrinsic qualities enable traceability and auditing, increasing public trust in energy markets and on energy differentiation. The simplest gateways and devices may ignore the warrants, that is, they can forward or process messages without opening the envelope.

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

2.4 EMIX Time and Schedules

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

WS-Calendar defines a mechanism to apply a schedule to a ~~sequence~~Sequence of time ~~intervals.~~Intervals. WS-Calendar further defines how to use a process analogous to inheritance to apply a single information artifact to each ~~interval~~Interval in the ~~sequence~~Sequence, allowing elements of that

[3] ~~The phrase "prices to devices" is used in energy policy discussions to describe a market model in which energy use decisions are distributed to each device that uses energy. Under this model, decisions about whether to use energy now or delay energy use until later are best made where the value is received for that energy use, the end device. The smart toaster is shorthand for the smallest, least capable device that can receive such a message.~~

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Footnote Text

Formatted: English (U.S.)

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

artifact to be over-ridden within any given ~~interval.~~Interval. WS-Calendar also defines a schedule entry point, defining how specific performance can be contracted and scheduled.

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

2.5 Tenders and Transactions for Power Products and Resource Capabilities

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

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

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

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

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

2.6 Transport

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Product ~~Transport incurs specific~~transport costs ~~that~~ vary over time. Delivery costs come in two general forms. Congestion charges apply to each unit of ~~Product~~product that passes through a particular point in the distribution system. Congestion charges increase the cost of the Product delivered in a particular Interval. Loss reduces the ~~Product~~product delivered ~~below the amount contracted for~~ as it passes from the purchase point to the delivery point. Loss may reduce the amount of ~~Product~~product received or a loss ~~change~~charge may be applied to purchase replacement energy for the energy loss.

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 Products~~Power Transport Product* Descriptions.

2.7 Verification

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

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

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

Formatted: Font: 8 pt

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

Formatted: Font: 8 pt

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

All product descriptions include the EMIX Interface. The EMIX Interface is where something transfers ownership for the market. In Power, it can be a node or meter, and aggregation of nodes or meters, a pair of nodes, or a geographic area. 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 types.

~~2.8~~3.2 Extensibility

EMIX supports a modular model in which extensions to EMIX can easily be propagated into standards ~~the~~that communicate EMIX. There are multiple EMIX envelopes to participate in different roles; each includes a set of EMIX ~~Terms~~Base Types that describe what is tendered or transacted~~. EMIX Terms are described by applying an EMIX Product Description to a WS-Calendar Sequence~~.

New efforts could specify ~~additional~~novel Product Descriptions~~.~~ by extending the EMIX Product Description Type. These new ~~product~~Product Descriptions would ~~generate~~define new conformant EMIX ~~Terms~~ Products merely by ~~applying the new Product Descriptions to~~application the ~~WS-Calendar Sequence.~~EMIX Base Type. Such conforming Products could then be transported on any EMIX Envelope. Any Specification that communicates EMIX ~~Terms~~Products can ~~then communicate~~exchange market information about these new Product Descriptions. A new committee can extend EMIX into new products without re-considering any aspects of the EMIX specification itself.

A similar logic applies to the warrants, which ~~are not specified in v1.0.~~outside the scope of this specification. If the warrant information varies over time, the warrant information can be applied to a WS-Calendar ~~sequence~~Sequence just as if it were a Product Description.

Extensibility mechanisms supported in EMIX are discussed in Appendix B.

3.3 Power and Resource Schemas

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

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

~~3~~4 Overview of the Information Elements

EMIX describes the ~~Terms~~market communications (EMIX ~~Terms~~Base type) of tenders and transactions for products whose ~~markets are volatile.~~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~~Interval of a ~~sequence~~Sequence of ~~intervals~~Intervals may vary for energy transactions made in a ~~sequence~~Sequence of ~~intervals~~Intervals.

EMIX ~~Terms~~Base derived types are ~~defined~~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 ~~time interval~~Interval or a ~~sequence~~Sequence of ~~time intervals.~~Intervals. An EMIX Product Description for a constant rate of delivery power product over a single ~~interval of time~~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 ~~messaged~~exchanged in advance, the ~~message~~information exchanged to deliver the product is simply "start (reference ~~Uri~~**[URI]** to product) at 3:00 AM for 0.75 hours."

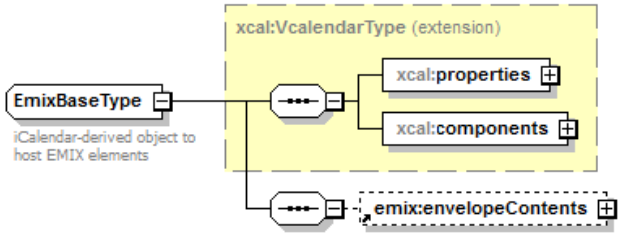


Figure 4-1: EMIX Base Type

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

For example, a resource representing a responsive load may ~~require~~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 ~~these~~ invariant characteristics as part of a product description or resource, while offering the variable run-time to the market.

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

EMIX ~~Terms also~~Base types minimize the size of ~~EMIX based messages~~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.

~~3.1~~4.1 The Intrinsic Elements: EMIX Products

The following table ~~(Table 3-1)~~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

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

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.

Central to each is the Base Product Description Type. The Base Product Description Type is the abstract class from which all Product Descriptions are derived. A Product is a description of the product or service applied to a delivery schedule. Product Descriptions as concrete classes, make up most of the sections 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

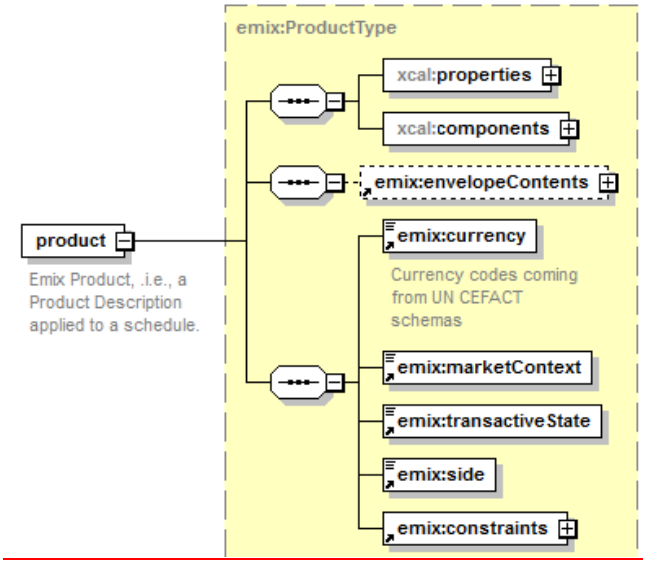


Figure 4-2: EMIX Product Type

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

Table 4-1: ~~Intrinsic~~ Elements ~~the "Face~~ of the ~~Envelope"~~ EMIX Product

~~Intrinsic~~Product Element		~~Specification~~Definition	~~Note~~
~~Uid~~UID		Identifier of this artifact. In many (if not most) markets the UID is required to be globally unique.	
~~Created date time~~		~~Datetime this artifact was produced~~	
Transactive State	~~Enumerated string~~	Used to aid parsing and conformance, e.g., to distinguish between ~~tender and transactive~~different purposes for EMIX communications See *Table 4-5: Transactive States Enumeration*	

Deleted Cells
Formatted Table
Formatted Table
Deleted Cells
Formatted: Font: 8 pt

IntrinsicProduct Element		SpecificationDefinition	Note
TermsEMIX Product	EMIX Terms artifact as defined in later sections of this specification	EMIX Terms describe the product/ commodity, the location and delivery intervals. EMIX Terms are constructed by the application of a Product Description to the gluons and intervals of a WS-Calendar Sequence. In the simplest case of direct specification of an interval, with no gluon, this leaves only the product description, the performance time, and the durationEMIX Products are created by applying a Product Description to a Schedule using the Base Product abstract class.	
Price		Float. (Optional)	Is the sum of the extended price of intervals only if the intervals are purchased as a single tender or transaction.
Package Discount		Float (Optional)	There may be market reasons for the price to be different than the Extended Price
Market Context	Xs:anyUri. An identification of the market in which the product is offered.	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, andor an identification of the microgrid in which the productProduct is priced.	
Party Type		Xs:anyUri. An identifier for one of the parties to a tender or transaction.Identifies whether information originator is Buyer or Seller	
CurrencyCounterParty		Xs:anyUri. An identifier for one of the parties to a tender or transaction.A code that indicates the currency used, as specified in **[CEFACT]**	
SideConstraints		The role (buyer or seller) of the Party. The Counterparty takes the other role.A collection of business and performance rules that constrain the option agreement. See Constraints at Section 16	
Envelope ContentsCurrency		A code that indicates the currency used, as specified in **[CEFACT]**As defined in section 4.3 *Inside the Envelope – the Extrinsic Items*	Examples include USD, CAD, GBP, EUR, CNY. Could be a nominative or shadow price referenced to e.g. microgrids

Deleted Cells
Formatted Table
Formatted Table
Deleted Cells
Deleted Cells

3.2 ExtrinsicIntrinsic Elements

Extrinsic elements are those that are not inherent to the nature of the product. Customers or regulations may value them, and they may affect the price received on the market for a product. Extrinsic elements are contained within the envelope.

Formatted: Font: 8 pt

4.1.2 ~~Table 3-2 lists defines contents of the envelope, i.e., the extrinsic elements in the~~ EMIX ~~information model. These items are in the general from of an EMIX Product Description, and can be elaborated using EMIX Terms if there is a time element to its information.~~ Option

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

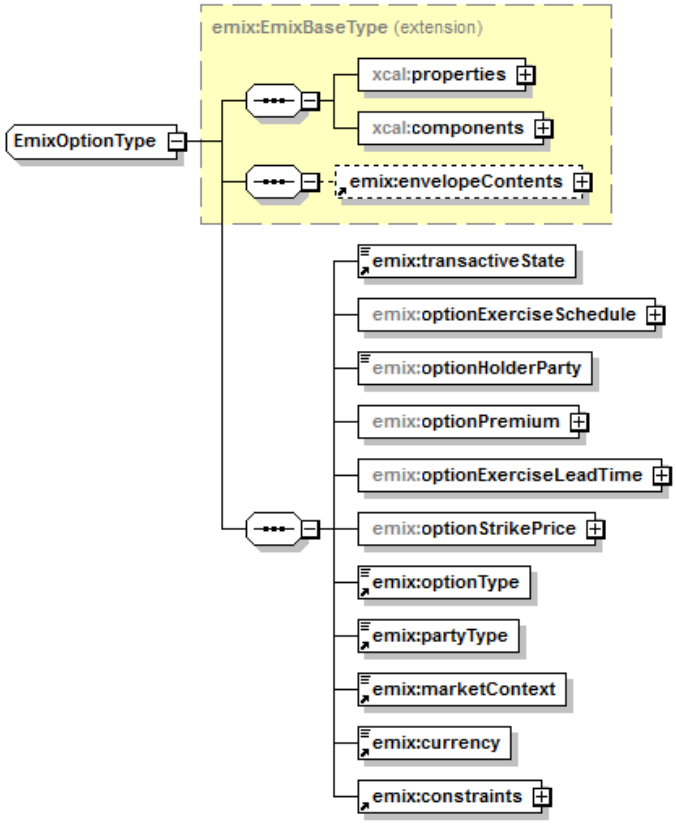


Figure 4-3: EMIX Option Type

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

~~2: Extrinsic~~ Table 4-2: Option Elements ~~"Contents~~ another "Face of the Envelope"

~~Extrinsic Element~~	~~Specification~~	~~Note~~
~~**Envelope**~~	~~Optional. Container for extrinsic information as defined in the next section.~~	~~The envelope contains supporting information that goes beyond that natively in the transaction or tender.~~

Formatted: Heading 3,H3, Outline numbered + Level: 3 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.5"

Formatted: Font: 8 pt

~~Extrinsic Element~~	~~Specification~~	~~Note~~
~~Warrant List~~	~~The container for array of warrants. Optional.~~	~~An array of the warrants included in the envelope. See section 4 for warrants.~~
~~Support of Price~~	~~Container holding information supporting price information~~	~~May include EMIX Terms, if several are combined to produce the intrinsic price.~~
~~Program~~	~~A possibly structured name for a program in which the price and product are tendered or transacted.~~	~~This may be analogous to a contract identifier. The variety of DR "programs" inspired this proposed element.~~

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

Formatted: Font: 8 pt

~~Extrinsic Element~~	~~Specification~~	~~Note~~
Constraints	A collection of business and performance rules that constrain the option agreement. See Constraints at Section 16	

4.1.3 Intrinsic Elements of the TeMIX

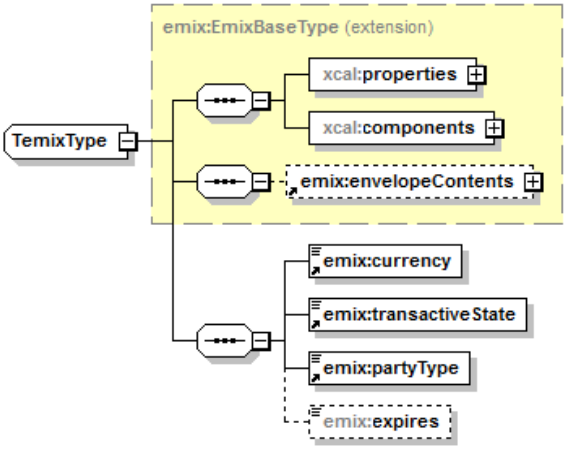


Figure 4-4: The TEMIX Product

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

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*

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

Formatted: Font: 8 pt

4.1.4 Intrinsic Elements of Delivery

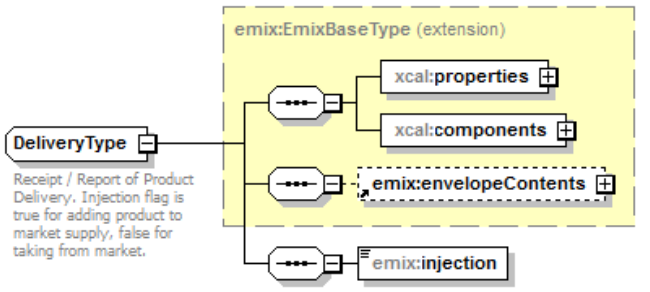


Figure 4-5: Delivery

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 ~~other~~ 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.

Formatted: Font: 8 pt

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.

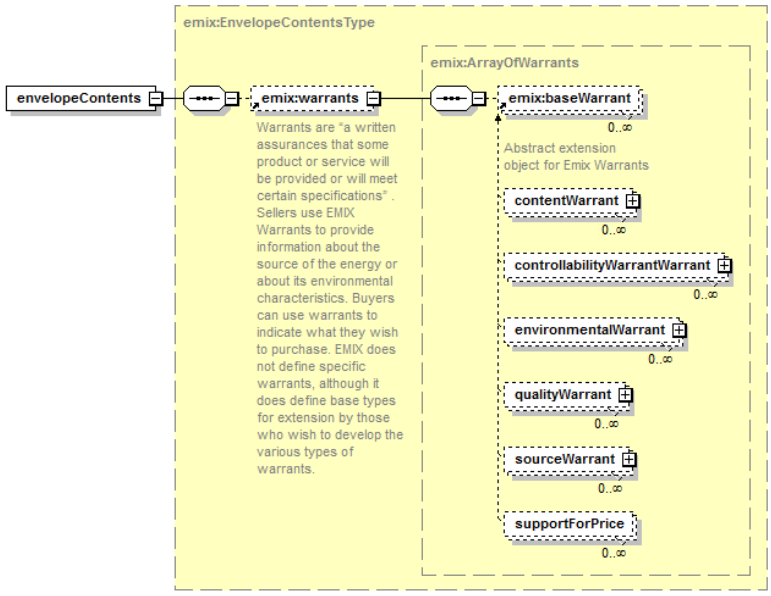


Figure 4-6: Envelope Contents

Formatted: Font: 8 pt

The definition of a warrant is “a written assurance that some product or service will be provided or will meet certain specifications”[4] . 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. It seems a fundamental market rule that a middleman cannot sell more wind power than he has bought. Such rules are beyond the scope of EMIX, but EMIX-based information exchanges are designed to support such market rules.

EMIX Warrants are assertions about the EMIX TermsProduct.

4.4 Summary of the EMIX Base Derivations

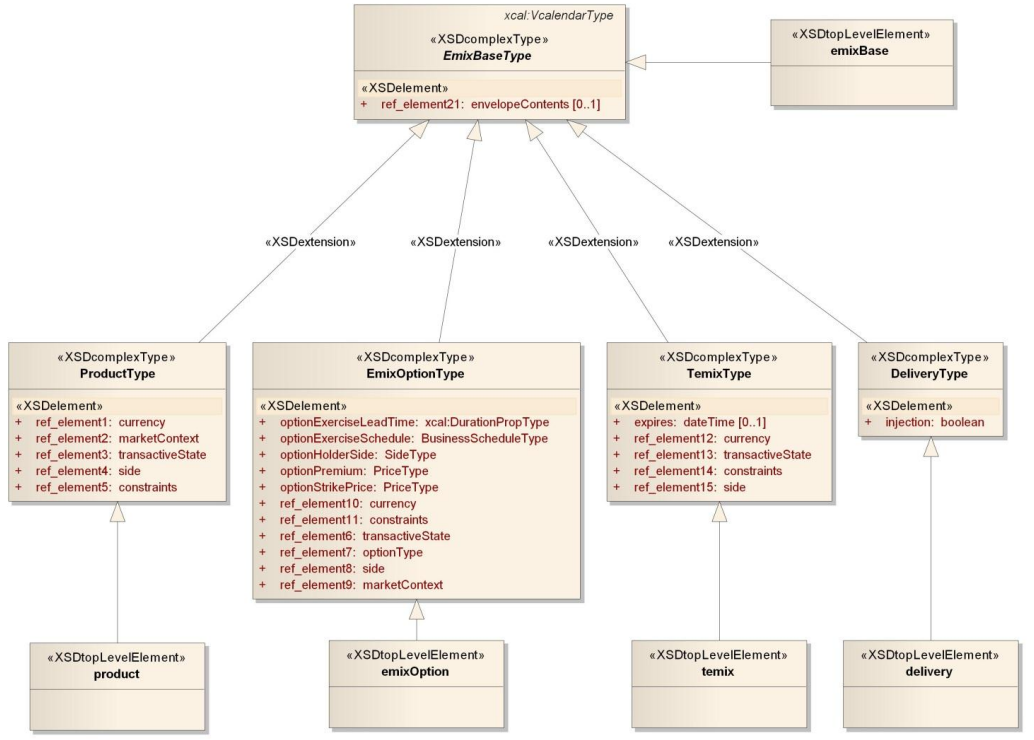


Figure 4-7: UML of EMIX Base and its Extensions

[4] Ibid

Formatted: Font: 8 pt

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 0 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

Formatted: Font: 8 pt

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

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

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

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

~~*Table 3-3: Examples of Warrant Information*~~

Warrants are discussed in Section 15.

Formatted: Font: 8 pt

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.

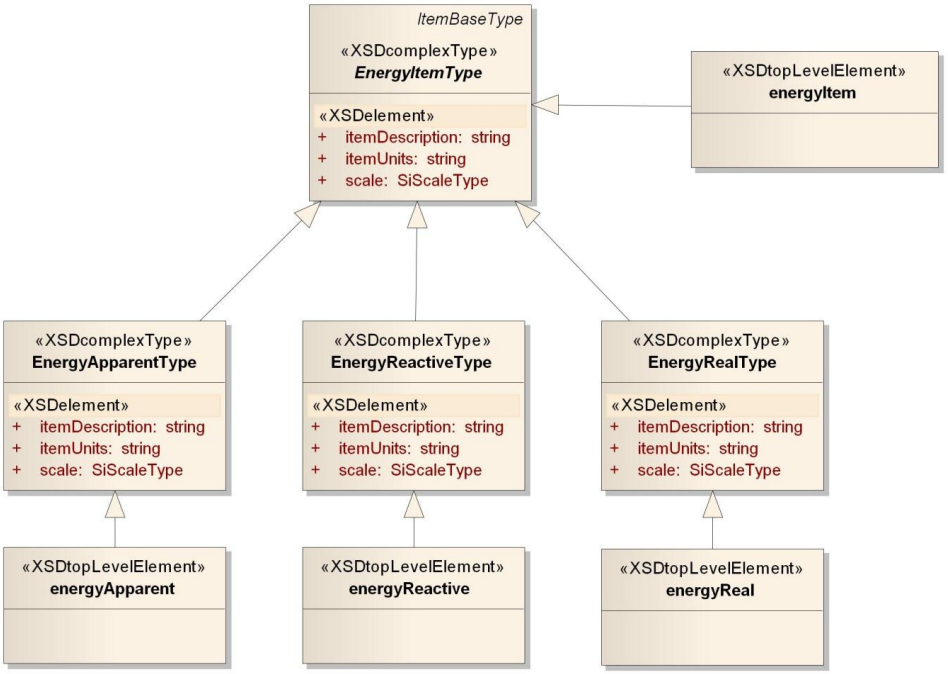
The Item Base is used not only to quantify the Item, but potential attributes of the Item as well.

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.

Formatted: Font: 8 pt



542

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

Formatted: Font: 8 pt

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

~~Warrant~~ Gluon Element		~~Specification~~Definition	~~Note~~
~~**Quality Warrant**~~	~~A Product-specific assertion of Quality. For Electric Power products, these are based upon **[IEEE 1159]** based metrics.~~**Product Description**	~~For a tender, this can be a promise of or requirement for quality. For verification, this can be actual measurements. If during an indication of interest, might be a desired minimum standard.~~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.	
~~**Environmental Warrant**~~**Gluon Duration**	~~An enumeration of the environmental burden caused by the production of the energy product in the quantity and units indicated~~	~~The initial EMIX standard included a non-normative artifact contributed by the Energy Information Standards Alliance (EIS Alliance). It is anticipated that markets will create environmental warrants relevant to their unique needs.~~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.	

Deleted Cells
Formatted Table
Deleted Cells
Formatted: Font: Bold
Deleted Cells

Formatted: Font: 8 pt

~~Warrant~~ Gluon Element		~~Specification~~Definition	~~Note~~
~~Content Warrant~~Gluon Unit Price		~~A warrant about the means of production for the energy. These may be used to warrant the content of storage, as the nature of the original input to storage is not altered when drawn from storage.~~A Price set in a Gluon can be inherited by a Sequence of Intervals, subject to the inheritance rules. Not present in all Gluons.	~~The proportion of the product defined that is from non-fossil fuel sources, including but not limited to "hydroelectric", "solar", and "wind".~~
~~Sequence~~~~Source Warrant~~	~~Individual source warrants~~	~~In aggregate may be the same as Content Warrant~~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	
~~Controllability Warrant~~Designated Interval	~~An authority warrants that a resource can be controlled to the standards used by that authority~~	~~Usually a prerequisite for participation in direct control contracts.~~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.	

Deleted Cells
Formatted Table
Formatted Table
Deleted Cells
Formatted: Font: Bold
Deleted Cells
Deleted Cells

~~3.3 EMIX Options~~

~~The EMIX Option is a variation on the EMIX envelope described above in section "The Intrinsic Elements". An option gives the buyer the right, but not the obligation, to buy or sell a product at a set price during given time windows. The EMIX option also specifies specific response times. The "face of the envelope" displays additional information to support these requirements.~~

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~~: 4: Option Elements – another "Face of the Envelope"~~

~~Intrinsic Element~~	~~Specification~~	~~Note~~
~~Uid~~	~~Identifier of this artifact~~	~~The format of this ID varies by the communication is is intended for. For wider markets, the UID should be globally unique.~~
~~Created date time~~	~~Datetime this artifact was produced~~	

Formatted: Font: 8 pt

Intrinsic Element	Specification	Note
Transactive State	Enumerated string	Used to aid parsing and conformance testing, e.g., to distinguish between tenders, transactions, and history.
Terms	EMIX Terms artifact as defined in later sections of this specification	EMIX Terms describe the product/ commodity, the location and delivery intervals. EMIX Terms are constructed by the application of a Product Description to the gluons and intervals of a WS-Calendar Sequence. In the simplest case of direct specification of an interval, with no gluon, this leaves only the product description, the performance time, and the duration
Option Exercise Schedule	Vcalendar collection (from **[ICalendar]**)	An option may 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.
Option Holder Party	Xs:anyUri	The party which enjoys the benefit of choosing whether or not to exercise the terms specified in the option. The Promisee.
Option Premium	EMIX Price	The price paid to the Promisor for the rights involved
Option Strike Price	EMIX Price	The price at which an option holder (Promisee) has the right to require the option writer (Promisor) to deliver.
Option Exercise Delivery Time	duration	An EMIX Option specifies required lead time before the response as well as the ability to deliver.
Extended Price	EMIX Price. The sum of all intervals in the Product above. (Optional)	Is the sum of the extended price of intervals only if the intervals are purchased as a single tender or transaction.
Package Discount	EMIX Price. (Optional)	There may be market reasons for the price to be different than the Extended Price
Market Context	Xs:anyUri. An identification of the market in which the product is offered, or the counterparty if part of a bilateral non-market transaction. (Optional)	This may include standard financial exchanges, markets managed by or for aggregators and distributors, and an identification of the microgrid in which the product is priced.
Currency	A code that indicates the currency used, as specified in **[CEFACT]**	Examples include USD, CAD, GBP, EUR, CNY. Could be a nominative or shadow price referenced to e.g. microgrids
Envelope	Container for extrinsic information as defined in the next section. (Optional).	The envelope contains supporting information that goes beyond that natively in the transaction or tender.

Formatted: Font: 8 pt

4 Generic EMIX Terms

The generic EMIX Terms are defined by a set of EMIX Elements as described in Table 4-1. The Generic Terms become specific when a Product Description is applied to the Generic Terms. Specific Product Descriptions contain additional EMIX Elements as described Section 5 through 10.

This section also indicates how information from the product description, along with price and quantity, is applied to the gluon and interval. Schedule information can be applied to each as described in **[WS-Calendar]**

Table 4-1: EMIX Product Elements

Product Element	Specification	Note
Product Description	Emix.ProductDescription object	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. Inheritance is as described in **[WS-Calendar].** The ProductDescription is an extension of the Artifact that is a part of each Interval and Gluon.
Gluon Duration	WS-Calendar duration Optional	Sets default duration for Intervals in the Sequence. Not known in all interactions and nor present in all Gluons.
Gluon Quantity	Float Optional	Sets default Quantity for all Intervals in the Sequence. Not known in all interactions and nor present in all Gluons.
Gluon Unit Price	EMIX Price, Optional	Sets for all Intervals in the Sequence not otherwise priced. Not known in all interactions and nor present in all Gluons.
Sequence	WS-Calendar:Sequence (collection of Intervals) Mandatory	A sequential set of Intervals including expression of Price, Quantity, or Both. May also include elements of the Product Description
Starting DateTime	Optional	Only required when scheduling a sequence. Applies to the associated interval – starting times of other intervals are computed from the sequence based on the sequence starting time, the temporal reltions between intervals, and the duration of each.
Associated Interval	From WS-Calendar	Link from the EMIX Gluon into the sequence of Intervals.

Formatted: Font: 8 pt

4.1 EMIX Intervals

The Gluons point to a set of intervals with defined temporal relationships. An example of intervals with temporal relationships is a set of consecutive intervals. A collection of such intervals is known as a Sequence.

Table 4-1: EMIX Base Product Elements- the Interval

Product Element	Specification	Note
Product	Emix.ProductDescription object	Elements of the Product Description that can be inherited without change from the Gluon need not be expressed in the Interval. The ProductDescription is an extension of the Artifact that is a part of each Interval and Gluon.
Duration	WS-Calendar duration Optional	Can be inherited from the Gluon Set
Quantity	Float Optional	Can be inherited from the Gluon Set
Unit Price	Float, Optional	Can be inherited from the Gluon Set
Starting DateTime	Optional	Usually be inherited from the Gluon Set. Only one Interval per sequence gets a Starting DateTime
Temporal Relation	From WS-Calendar	Link from one interval to other intervals in the sequence.

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.

Formatted: Font: 8 pt

~~4.2~~7.3 EMIX Product Model

The illustration below provides a model ~~for how this can work~~demonstrating a sequence of three intervals, and the successive application of Gluons to bring them to market.

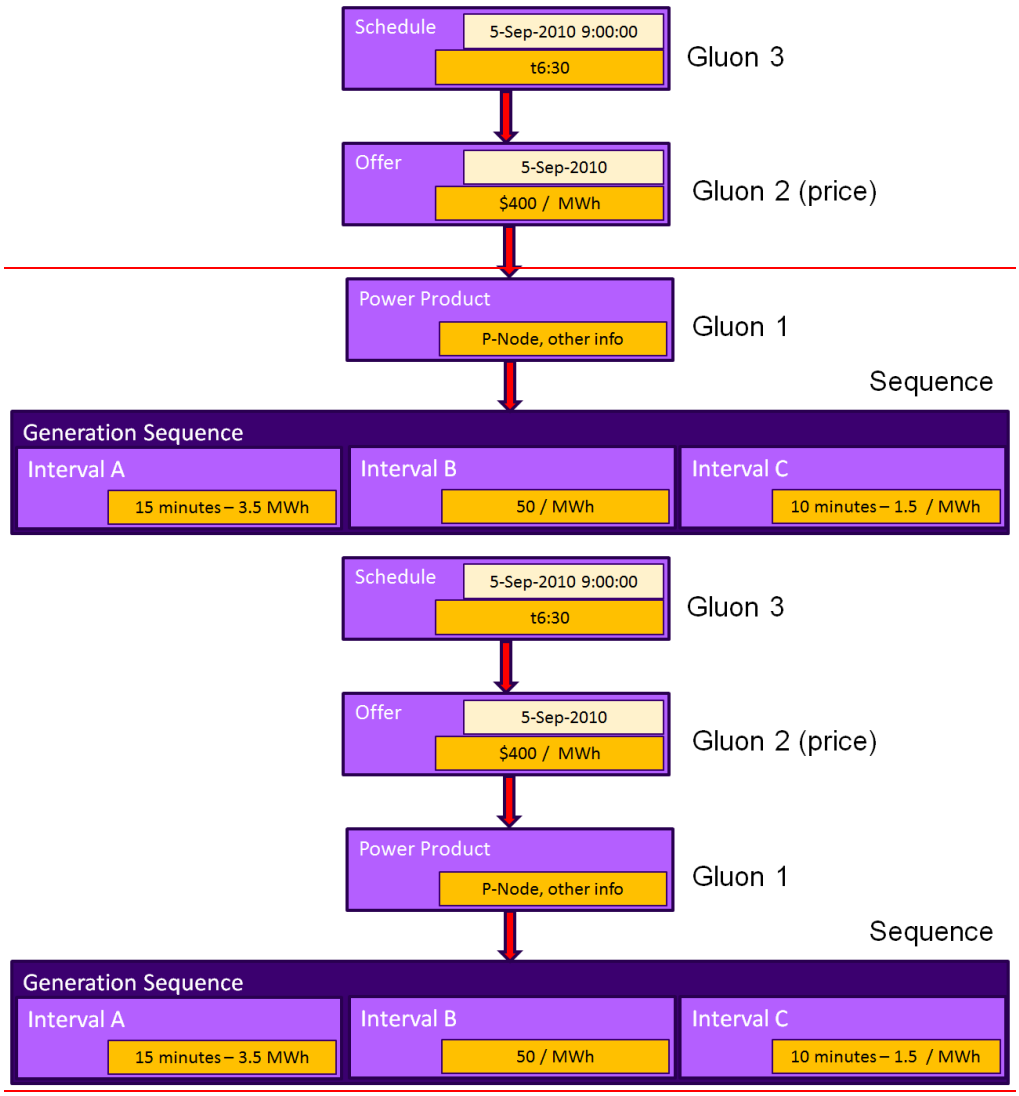


Figure 1-1: EMIX Model

1. *Power source defines product to market (Sequence and Gluon 1).*
2. *Product is offered to market on a particular day ([1] and Gluon 2) (Date but not time, required price specified)*
3. *Transaction specifies start time (9:00) and duration (6:30) (Gluon 3), inherited by Sequence through Gluons 2 and 1. Interval B (linked to Gluon 1) is the ~~interval~~Interval that starts at 9:00.*

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

58 EMIX ~~Electrical Energy and~~ Power Product Descriptions

Electrical Energy ~~(measured in MWh, for example) does work. Electrical Power is the rate of delivery of Energy (measured in MW, for example). Often the terms energy~~ and must be described precisely as it comes to market. Different products can provide total power ~~are used in conversation interchangeably without confusion, for EMIX, precision of language for energy and~~, real power ~~is crucial.~~, 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 ~~C~~E.

~~5.1 Taxonomy of EMIX Power Product Descriptions~~

~~EMIX~~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 ~~Descriptions are broken down into the following~~. The sections ahead discuss three classes ~~discussed below:~~ of Product Description:

1) Power Product Descriptions
2) Resource Offer Descriptions
3) Transport Product Descriptions

~~All~~ 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.

~~5.1.1~~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 ~~for transactions are discussed in the rest of this section.~~in Section 9, *Power Product Descriptions*.

~~5.1.2~~8.2 Resource Offer Descriptions

Resources ~~are~~include generators that can produce ~~energy~~power and other services, storage devices that can consume, store and then produce ~~Power Product~~power, and load ~~curtailment contracts~~ that produce a ~~Power Product from~~ 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 ~~7~~0, *Energy Resources*.

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Heading 2,H2, Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Heading 2,H2, Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: Italic

Formatted: Font: 8 pt

~~5.1.3~~8.3 Transport Product Descriptions

Product ~~Transport~~ transport incurs specific costs that vary over time. ~~Delivery~~ Transport costs ~~come in two general forms. Congestion~~include congestion charges that apply to each unit of Product that passes through a particular point in the distribution system~~. Congestion charges increase the cost of the Product delivered in a particular Interval. Loss~~, and loss, .which reduces the Product delivered ~~below the amount contracted for as it passes from the purchase point to the delivery point. Loss may reduce the amount of Product received or a loss charge may be applied to purchase replacement energy for the energy loss.~~

. 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 Products Descriptions.*~~

~~6~~ ~~Power Product~~, *Power Transport* *Product* Descriptions ~~*Descriptions*~~.

Formatted: Heading 2,H2, Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: Italic

Formatted: Font: Italic

Formatted: Normal

Formatted: Font: 8 pt

6.19 Transactive Power Product DescriptionDescriptions

The Transactive Power Product Description is based on a simple product description: Power, Price, The information model in this section is described in POWER-PRODUCTS.XSD

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

Attributes, and Service Location. As defined in EMIX, a Power Interval has two potential forms, a ramped power interval and for a constant power interval. A constant power interval uses the power quantity specified locally or one inherited from the Gluon. A ramped power interval cannot inherit the power quantity because it contains two power quantities internally: the starting rate and the final rate. Both interval types are reflected in the table below:

Table 9-1: Power Interval DescriptionSemantic Elements common to Multiple Power Products

Name		Definition
Constant Power Quantity	**EMIX.Quantity Interface**	Defines Constant Power Intervals. Does not coexist with Starting and Final Po… number of metering points (as defined below), an aggregate point, or a geogra… ownership
AttributesStarting Power Quantity		EMIX.QuantityEssential characteristics (Voltage, Hz, AC or DC) of delivered electricity.
Final Power QuantityVoltage		EMIX.QuantityOne of three elements hereafter referred to as the Power Attributes.
Power UnitsHertz		Power UnitsOne of three elements hereafter referred to as the Power Attributes. Always 0 for DC
Service LocationAC	Service Location	Should normally be only in the Gluon and omitted from the intervals. If the Proc… multiple locations, one per interval, then it MAY appear in the interval instead C… the Power Attributes.
Power AttributesUnits		Power AttributesEnumeration of Power Units, e.g., total power (VA), real power (W), and reactive power (VAR)
UnitPriceEnergy Units		EMIX.PriceEnumeration of Energy Units, e.g., including real energy (Wh), reactive energy, (VARh), and apparent energy (VAh)
PriceVoltage Units		EMIX.PriceEnumeration of Voltage Units, e.g., MV
DurationVAR Units		From WS-CalendarEnumeration of volt amperes reactive (var) units, e.g., Kvar
PerformanceMeter Asset		From WS-CalendarIdentifier for an actual or virtual meter
Node		Grid Location identifier

Formatted: Heading 1,Heading 1 Char1 Char,Heading 1 Char Char Char, Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Font: 10 pt, Font color: Black

Deleted Cells

Formatted Table

Deleted Cells

Formatted: Font: 10 pt, Font color: Auto

Formatted: Font: Bold

Formatted: Font: Bold, Font color: Black

Formatted: Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers

Deleted Cells

Formatted: Font color: Black, Highlight

Formatted: Font color: Black, Highlight

Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers

Formatted: Font color: Auto, Not Highlight

Formatted: Font color: Auto, Not Highlight

Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers

Deleted Cells

Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers

Formatted: Font color: Auto, Not Highlight

Formatted: Font color: Auto, Not Highlight

Formatted: Font color: Auto, Not Highlight

Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers

Formatted Table

Formatted: Font color: Auto, Not Highlight

Deleted Cells

Formatted: Font color: Auto, Not Highlight

Formatted: Font: 8 pt

Name	Definition
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, a of a reference to a market context.

Deleted Cells

Formatted Table

~~The Gluon shares the same information elements with the exception that ramps are not defined for Gluons.~~

Formatted: Font: 8 pt

9.1 Base Power Contract

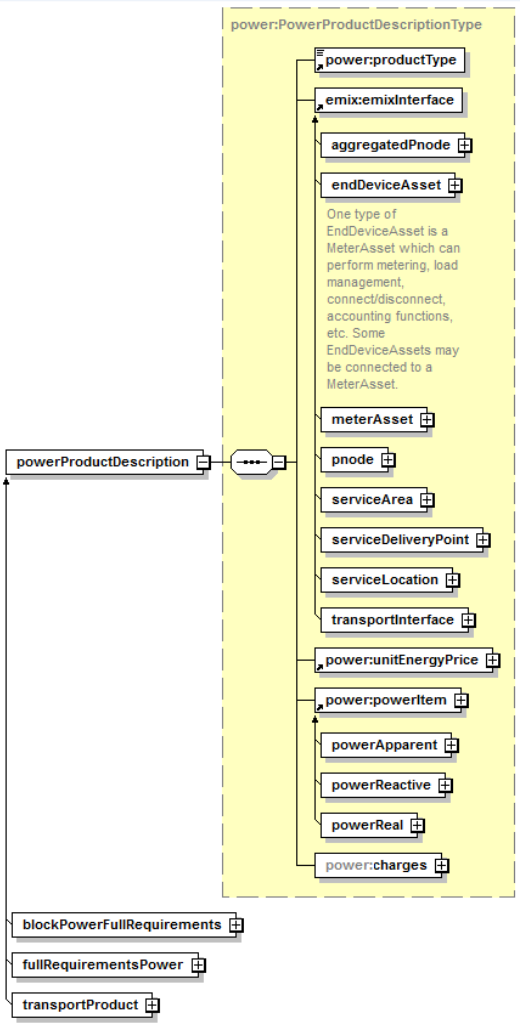


Figure 9-1: Base Power Product

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

Table 9-2: Base Power ~~Gluon~~Product Description

Name		Definition
Power ~~Quantity~~Product Type	~~EMIX.Quantity~~	~~Defines Constant Power Intervals. Does not coexist with Starting and Final P~~ Product. Used to determine conformance requirements.

Deleted Cells
Formatted Table
Deleted Cells
Formatted: Font: 8 pt

Name		Definition	
~~Power Units~~		~~Power Units, enum~~	~~As defined below~~
~~Service Location~~EMIX Interface	~~Service Location~~	~~If response is for a single location, should be in gluon to apply to the entire s~~ EMIX Interface is any of a number of market exchange points including a po which a product exchanges ownership.	
~~Power Attributes~~		~~Power Attributes~~	~~As define below~~
Unit Energy Price	~~EMIX.Price~~	Price per Unit ~~Quantity. Includes currency~~of Energy.	
~~Price~~Power Item		~~EMIX.~~Can indicate Real, Apparent, or Reactive Power~~Price~~	~~Ext~~ curr
~~Duration~~Charges		~~From WS-Calendar~~Charges affect the power product in addition to the cost of the product delivered. Charges are defined below.	~~May~~
~~Performance~~		~~From WS-Calendar~~	~~Indicates performance requirements such as fixed~~

~~No element in the gluon need appear in the interval unless the interval information supercedes the gluon information.~~

~~The constant power product is sufficient for all Transactive Energy uses. Many tenders that are offered or solicited as Resources are normally executed, i.e., contracted for performance, as a constant power product. (Ancillary Products are an exception – see section 8.) As the Power Quantity varies over intervals in the sequence, it describes a load curve. As the Price varies over intervals in the sequence, it describes a price curve.~~

~~6.2 Requirements Power Product Descriptions~~

~~The Requirements Power Product Descriptions below can successfully describe contracted power in use today including~~

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

9.2 Full Requirements Power

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

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

Table 9-3: Full Requirements Power ~~Products~~Product Description

Name	~~Definition~~Note
Price	EMIX Price is a choice one of a Price, a Price Multiplier, or a Price Relative.
~~Full Requirements Power~~Energy Units	~~Traditional power contract to provide all power used. Often used in retail residential rates. Demand Charges Optional~~Denominates the units that the Price applies to.
Attributes	Essential characteristics (Voltage, Hz, AC or DC) of delivered electricity.

Deleted Cells
Formatted Table
Deleted Cells
Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers
Formatted: Adjust space between Latin and Asian text, Adjust space between Asian text and numbers
Formatted: Font color: Auto, Not Highlight
Formatted: Font color: Auto, Not Highlight
Formatted: Font color: Black, Highlight
Formatted: Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers
Deleted Cells
Formatted: Font color: Black, Highlight
Formatted: Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers
Formatted: Font color: Black, Highlight
Formatted: Font color: Black, Highlight
Formatted: Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers
Formatted Table
Formatted: Font color: Black, Highlight
Formatted: Font: Bold
Formatted: Font color: Black, Highlight
Deleted Cells
Formatted: Don't adjust space between Latin and Asian text, Don't adjust space between Asian text and numbers
Formatted: Font color: Black, Highlight
Formatted: Font color: Black, Highlight
Formatted: Highlight
Formatted: Font: 8 pt

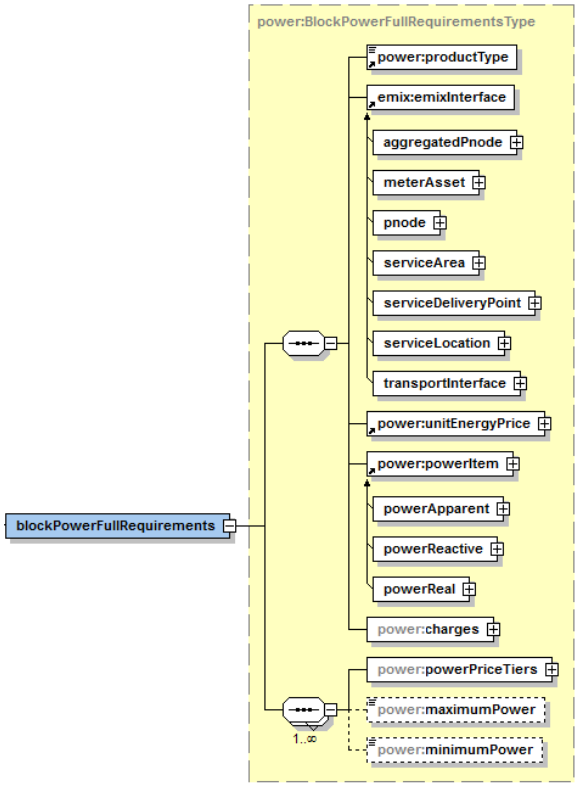
Name	DefinitionNote
Full RequirementsMaximum Power with Demand Charges	Often used in mid-sized and small commercial. Same as Full Requirements Power but with demand charges for "excess" use.Denominates the most power available for transacting during the period.
Requirements with Maximum and Minimum Power	Customer must draw energy at least the minimum rate (power) and no more than the maximum rate during any measurement interval.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.

Formatted: Font color: Black, Highlight

Formatted: Highlight

690 ## 9.3 Block Power Full Requirements

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



694

695 *Figure 9-2 Block Power Full Requirements*

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

698 *Table 9-4: Block Power Full Requirements*

Formatted: Font: 8 pt

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

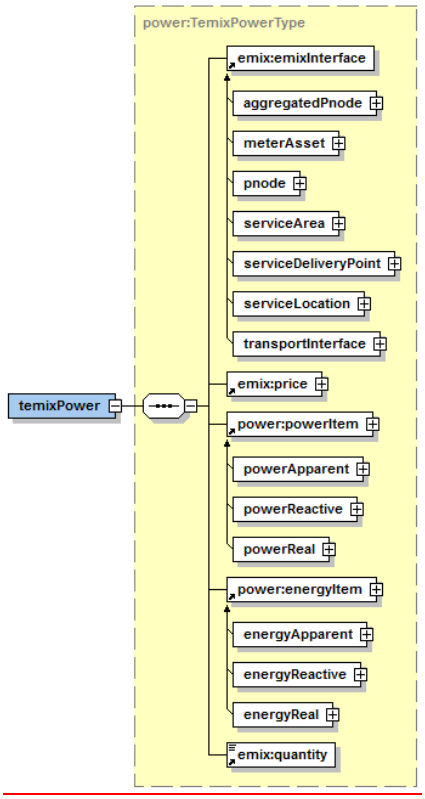


Figure 9-3: TeMIX Power

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

Formatted: Font: 8 pt

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
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.

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

9.5 Power Product Charges

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

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

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

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.

Formatted: Font: 8 pt

9.6 Enumerated Power Product Types

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

The following Power Product Types are enumerated:

Table 9-7: Requirements Power Products

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~~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

~~Contracted power~~ Power products such as these can ~~all~~ be described using the ~~Contracted Power Product Description~~

~~Table 6-4: Requirements~~ Power Product ~~Description~~

~~Name~~	~~Definition~~	~~Note~~
~~**Contract Type**~~	~~Enumerated String~~	
~~**Power Units**~~	~~Power Units~~	~~As defined below~~
~~**Service Location**~~	~~Service Location~~	~~If response is for a single location, should be in gluon to apply to the entire sequence and be omitted in the intervals~~
~~**Power Attributes**~~	~~Power Attributes~~	~~As defined below~~
~~**Price**~~	~~From EMIX~~	~~Price per Unit during the Interval.~~

Formatted: Font: 8 pt

Name	Definition	Note
Demand Charge	Demand Charge. Optional	See below. There may be multiple demand charges.
Maximum Power	Power	Buyer may not consume at more than this rate
Minimum Power	Power	If buyer consumes than this rate, the buyer is assessed a charge to bring it up to this rate.
Duration	From WS-Calendar	May be nil if all intervals have duration specified
Performance	From WS-Calendar	Indicates performance requirements such as fixed run-time, absolute end-time, etc.

Requirements Power may not match well with future smart energy scenarios. Requirements Power has no fixed forward obligation to take-and-pay for energy. Thus, there is no defined baseline for demand response or dynamic pricing. However, Requirements Power Descriptions are necessary for current legacy communications.

Table 6-5: Requirements Power Product Description

Name	Definition	Note
Contract Type	Enumerated String	
Block Power Price	Multiple occurs	Sequence of components defining the price of successive blocks of power. Each block has a Price, and a maximum energy quantity. If the contract is for an increasing block price, blocks are interpreted in order of increasing price, and for a decreasing block price contract, blocks are interpreted in order of decreasing price
Power Units	Power Units	As defined below
Service Location	Service Location	If response is for a single location, should be in gluon to apply to the entire sequence and be omitted in the intervals
Power Attributes	Power Attributes	As defined below
Price	From EMIX	Price per Unit during the Interval.
Demand Charge	Demand Charge. Optional	See below. There may be multiple demand charges.
Maximum Power	Power	Buyer may not consume at more than this rate
Minimum Power	Power	If buyer consumes than this rate, the buyer is assessed a charge to bring it up to this rate.
Duration	From WS-Calendar	May be nil if all intervals have duration specified
Performance	From WS-Calendar	Indicates performance requirements such as fixed run-time, absolute end-time, etc.

Demand Charges assess additional costs based peak rate of use by the buyer. Demand charges often extend beyond the current billing period.

Formatted: Font: 8 pt

744 *Table 6-6: Demand Charges Information Model*

Name	Definition	Note
Demand Charge Units	Power units	Single units used by all quantities
Demand Charge Floor	Quantity	Above this floor is exceeded, demand charges are applied
Demand Charge Rate	Price / Power	Incremental charge applied power if floor is exceeded.
Measurement Interval	Duration	Granularity or Power Use readings.
Collection Interval	Duration	Period during which power usage is summed for comparison to Demand Floor.
Collection Period	Duration	Usually the same as the billing period
Charge Duration	Duration	Period during which Demand Charges will be applied after incurred.

745 ## 6.3 Semantics of Power Products

746 The product descriptions refer to terms and data structures that had not yet been defined. These
747 elements are defined below.

748 First, there are simple base elements used again in defining power products, including those in the next
749 sections.

750 *Table 6-7: Simple Elements for use in Power Products*

Name	Definition	Note
Voltage	Decimal, May be measured or nominal	One of three elements hereafter referred to as the Power Attributes.
Hertz	Decimal, May be measured or nominal	One of three elements hereafter referred to as the Power Attributes. Always 0 for DC
AC	Boolean, true for AC, false for DC	One of three elements hereafter referred to as the Power Attributes.
Power Units	String	Enumeration of Power Units, e.g., MW
Energy Units	String	Enumeration of Energy Units, e.g., MWh
Voltage Units	String	Enumeration of Voltage Units, e.g., MV
VAR Units	String	Enumeration of volt amperes reactive (var) units, e.g., Kvar
Meter Asset	String	Identifier for an actual or virtual meter
Node	String	Grid Location identifier

751 Often, multiple simple units do or should appear together in specifications for constancy and
752 completeness. These are named and defined as below.

753

754 *Table 6-8: Compound Elements for use in Power Products*

Formatted: Font: 8 pt

Name	Definition	Note
Power Attributes	Voltage / Hertz / Ac	Group used in many definitions
Transaction Node	Node & Meter Asset	Location of a meter and the Service location the point of interconnection where capacity and/or energy transmitted by the provider is made available to the receiving party.
Pnode	Transaction Node	A pricing location for which market participants submit their bids, offers, buy/sell CRRs, and settle.
APnode	Transaction Node	Aggregated Pnode
Service Location	Transaction Node	For residential or most businesses, it is typically the location of the meter on the utility customer's premises. For transmission, it is the point(s) of interconnection on the transmission provider's transmission system where capacity and/or energy transmitted by the transmission provider is made available to the receiving party.
Service Place	Geo-location, i.e. kml:placemark	Typically a geo-referenced polygon that might contain many transaction nodes
Interface Pricing Point	Pnode or APnode or Service Location or Service Place	Typically the location of the meter on the customer's premises. For transmission, it is the point(s) of interconnection on the transmission provider's transmission system where capacity and/or energy transmitted by the transmission provider is made available to the receiving party. May also be a place containing nodes.

755

Formatted: Font: 8 pt

~~7 Resource Offer Descriptions~~

Formatted: Font: 8 pt

10 Energy Resources ~~offer~~

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

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

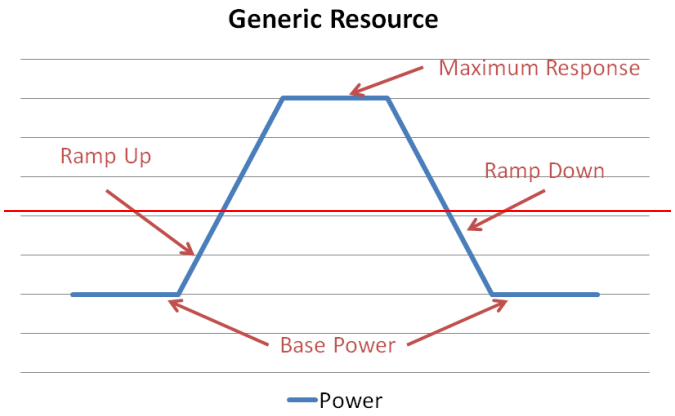
Resources often enter or are called to enter the market to meet specific needs. These needs can include a range of performance requirements; ~~resources~~Resources might be able to perform a range of capabilities. These performance capabilities are described using the information in ~~Resources Offers.~~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

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

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

~~7.1~~10.1 Resource Capabilities

Resources have capabilities rather than schedules. Resource descriptions describe what could be done, as distinguished from a transaction in which specific performance is requested or agreed to.



Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted: Font: 8 pt

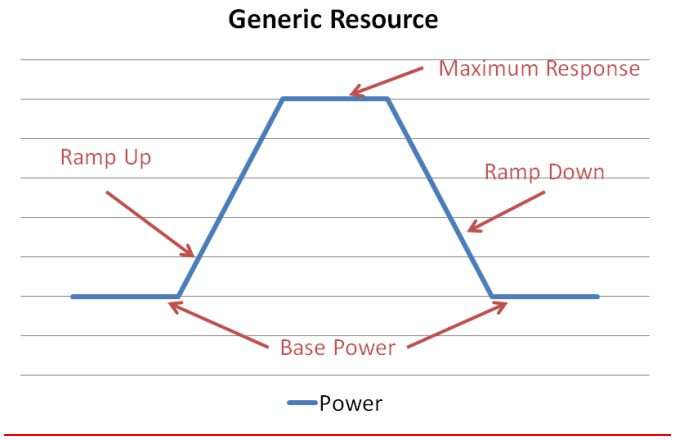
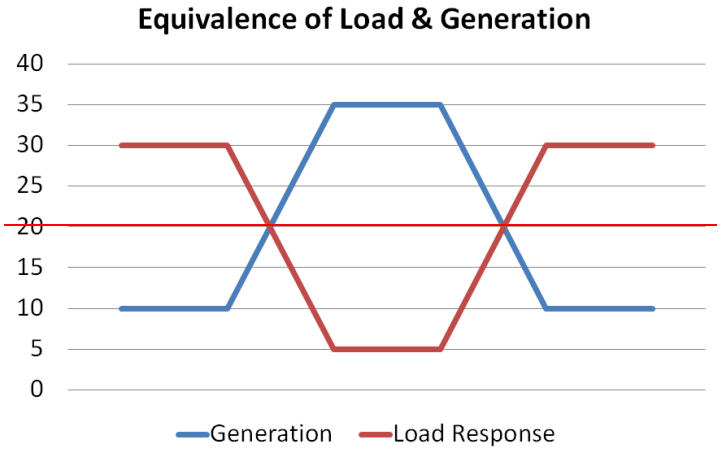


Figure 10-1: Attributes of a Generic Resource

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

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



Formatted: Font: 10 pt, Not Bold, Italic, Font color: Auto

Formatted: Font: 8 pt

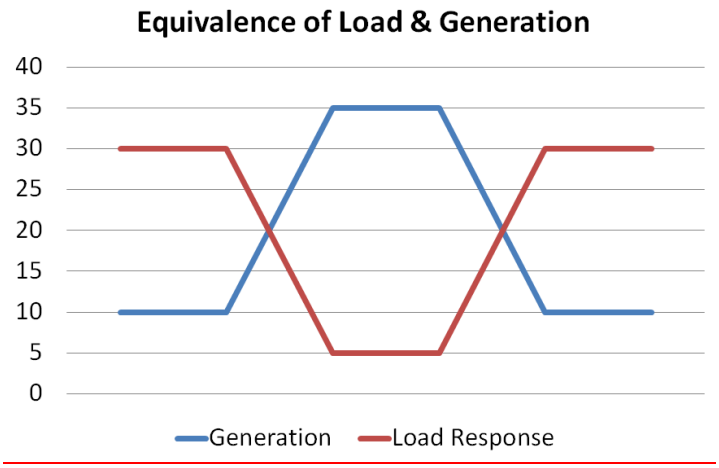
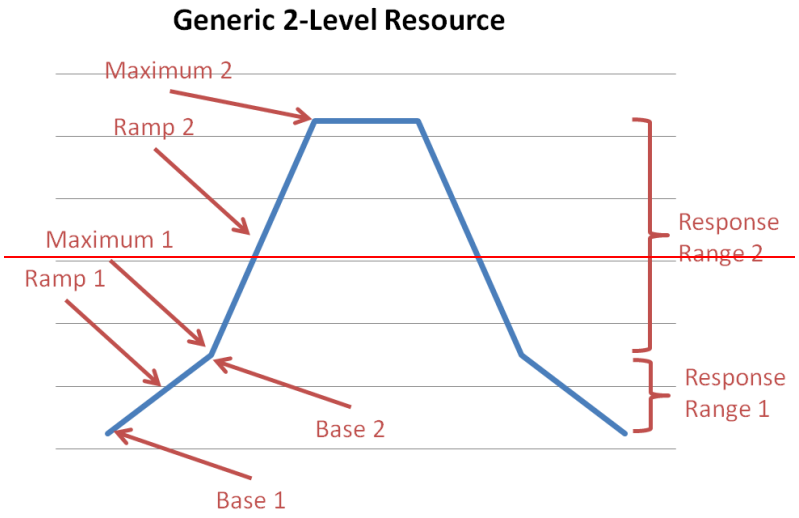


Figure 10-2: Equivalence of Load Shed and Generation

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

Many ~~resources~~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.



Formatted: Font: 8 pt

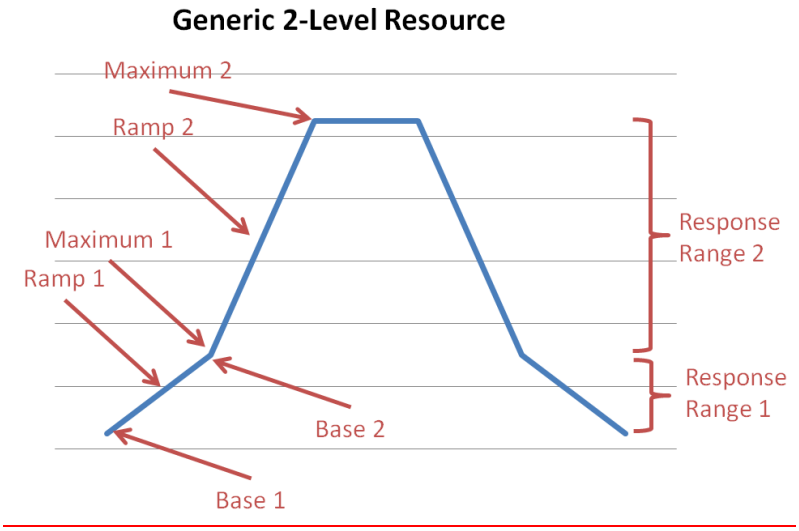
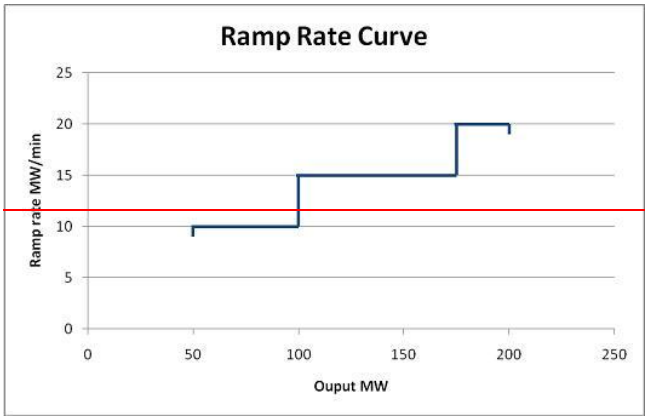


Figure 10-3: Combining Response Capabilities

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.

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

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



Formatted: Font: 8 pt

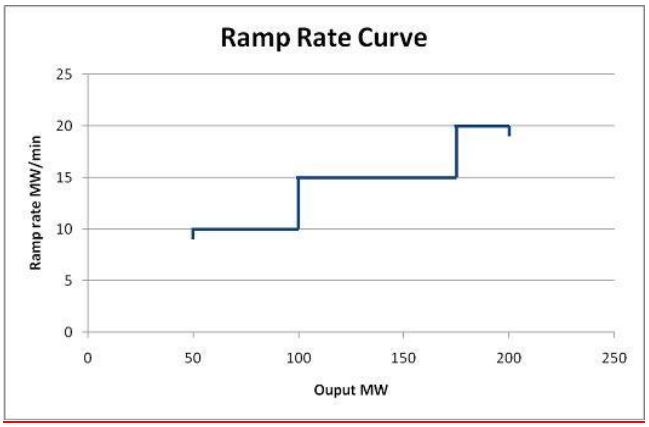


Figure 10-4: Ramp Rate Curve—CIM Style

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

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

~~7.2~~10.2 ~~Power~~ Resource Description Semantics

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

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

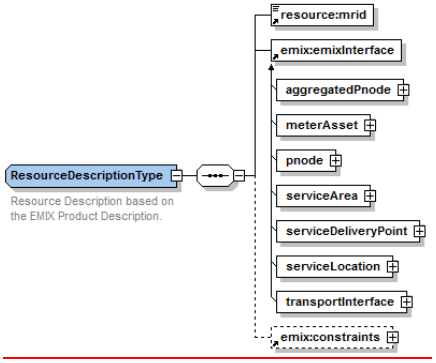


Figure 10-5: Resource Description base

The only aspects of a ~~resource~~Resource that matters to the energy market are the effects it can provide, the likelihood it will be able adequately to provide what it promises, and the financial incentives required to acquire them. The technology and process control details are many, and new ones may be required for each new power technology. Unless the market for the ~~resource~~Resource requires direct control, such details are irrelevant. The limited semantic set herein is sufficient to describe the capabilities of a ~~resource~~Resource.

Formatted: Font: 8 pt

EMIX bases its resource capability descriptions on the semantics in Table 7-1.

Table 7-1 Semantics for Power Resources

The EMIX Resource Description base consists of:

Table 10-1: Resource Description Elements

NameResource Description Element	Definition	Note
MridMRID	String	multiThe Multi-part resource id as defined in the ISO[IEC, TC57] CIM uniquely identifies each resource.
Notification Time	Duration	Time required for notification prior to beginning of response.
Response Time	Duration	Time required from notification to full response by the resource
Minimum Down Time	Duration	Minimum time interval between unit shut-down and start-up
Power Ramp RateEMIX Interface	Float & Power Units	Change up or down in units/minute between a starting power and an ending power.The Interface is where the Resource injects or extracts power. Note: for many transactions, reduced extraction is equivalent to injection.
Required Notice Time	Duration	Time period that is required from an order to reduce a load to the time that it takes to get to the minimum load reduction.
Shutdown Cost	Price	The fixed cost associated with committing a load reduction.
Offer Segment	Price, Maximum Power	Compound unit describing components of a tender. If multiple segments are offered (1st 50MW, next 100MW), Maximum Power is cumulative (50MW, 150MW). Offers are evaluated by sorting in order of increasing Maximum Power (for power) or decreasing Maximum Power (for load reduction) and must be purchased in order.
Minimum Resource Cost	Price per Duration	Resource requires this amount per period, i.e., a minimum requirement for $100 / hour at whatever power rate
Minimum Time Between Load Reductions	Duration	Shortest time that load must be left at normal levels before a new load reduction.
Minimum Load Reduction Interval	Duration	Shortest period load reduction must be maintained before load can be restored to normal levels.
Minimum Load Reduction	Power	Minimum units for a load reduction (e.g., MW rating of a discrete pump)
Minimum Load Reduction Cost	Price	Cost in currency at the minimum reduced load

Deleted Cells

Formatted Table

Formatted: Font: 10 pt, Font color: Auto

Deleted Cells

Formatted: Font: 8 pt

NameResource Description Element	Definition	Note
Maximum Operating Power	power quantity	The maximum operating power the purchaser can request from this unit
Maximum LoadConstraints	power quantity	Maximum load below which it may not be increasedAs well as all of the constraints listed for Product performance, Resources have additional constraints, listed below.
Minimum Load	power quantity	Minimum load below which it may not be reduced.
Power Ramp Rate	Power Quantity (rate), Duration, Begin Quantity, End Quantity	Between the Begin Quantity and End Quantity, Power can ramp at Quantity per Duration
Drop Ramp Rate	powerRampRate multipleoccurs	Maximum rate that load can be reduced. Begin Power must be greater than End Power
Raise Ramp Rate	powerRampRate multipleoccurs	Maximum rate that load may be restored Begin Power must be less than End Power
Is Controllable	Bool	Resource can be direct controlled. Warrant must be in envelope
Resource Class	Enumerated string	While a diverse set or resources can reduce risk, some resources may present covariant risk. For example, solar power in a region may ebb and flow in synchrony.

Formatted Table
Deleted Cells
Deleted Cells

In addition, voltage regulation services have their own semantics to specify voltvar.

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

Table 10-2 Semantics for Voltage Regulation Services

: Constraints unique to Power Resources

NamePower Constraint	Definition	Note	
Minimum LoadVMin	varQuantityConstraint on Minimum Load that a Resource can maintain	VMin is the IEEE 1547 minimum voltage level of 88% of nominal voltage where the PV inverter must disconnect	
Maximum PowerVMax	varQuantityConstraint on Maximum Power available from a resource	VMax is the IEEE 1547 maximum voltage level of 110% of nominal voltage where the PV inverter must disconnect.	

Formatted Table
Deleted Cells
Deleted Cells
Formatted: Font: 10 pt, Font color: Auto
Formatted: Font: 8 pt

NamePower Constraint	Definition	Note	
Maximum EnergyQMax		varQuantityConstraint on Maximum Energy available from a resource	Qmax is the inverter's current var capability and may be positive (capacitive) or negative (inductive). It would be the VA capability left after supporting the W demand.
Minimum Load ReductionvoltVar		voltageQuanity & varQuantityConstraint on Minimum Load Reduction resource can make	

7.310.3 Generic Power Resource Capability Descriptions

Resource Capability Products describe the capabilities of the resource using the semantics as above. The simpler of these interfaces mimic those found in traditional markets. Offer Load and Offer Generation describe more complete and abstract interfaces.

7.3.1 Load Curtailment Resource Capability Descriptions

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

Table 10-3 Responsive Load: Generic Power Response Resource – Simple Form

NameGeneric Resource Element	Definition	Note	
MridStaging Ramp		mridAn array Power Ramp Segments describing a Resource's ability to change level at the initiation of a Response	
Base LoadMinimum Response	powerQuantity	Load of system before requestThe least Response for which this resource will accept a request.	
Drop Ramp RateMaximum Response		rampDown multipleoccursThe greatest Response for which this resource will accept a request	Ramp rates are sorted by Descending maxima.
Minimum Load		powerQuantity	Load of system under full response
RaiseRecovery Ramp Rate	rampUp multipleoccurs	Ramp rates are sorted by ascending maxima.An array Power Ramp Segments describing how a Resource's returns to its original state following a response.	

The resource load is a simplified version of the market interface that appears in the TC57 CIM. Note that some of the terms are different because EMIX unifies terms across interfaces..

Formatted Table
Deleted Cells
Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"
Formatted: Not Highlight
Formatted: Not Highlight
Deleted Cells
Formatted Table
Deleted Cells
Deleted Cells
Deleted Cells
Formatted Table
Deleted Cells
Deleted Cells
Formatted: Highlight
Formatted: Font: 8 pt

Table 7-4 Offer Load Reduction

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

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

Power Ramp Segments consist of the following elements:

Table 10-4: Power Ramp

NamePower Ramp Element	Definition	Note	
Mrid		Mrid	
Drop Ramp Rate	dropRampRate, multipleoccurs	Ramp rates are sorted by descending maxima to assess response.Power Units for the Ramp	
Min LoadBegin Ramp		powerQuantityPower Quantity at the beginning of the Segment	Minimum Load system will accept
Min Load ReductionEnd Ramp		powerQuantityPower Quantity at the end of the Segment	Minimum reduction request resource will accept
Min Load Reduction Cost		Price	Minimum price to get resource to make minimal response
Min Load Reduction Interval	**Duration**	Minimum time for which resource will accept a load reductionThe time to get between the begin ramp and the end ramp.	
Min Time Bet Load RedIntegral Only		DurationIf true, one can't stop between the begin and end rates.	Shortest time that load must be left at normal levels before a new load reduction.
Raise Ramp Rate		raiseRampRate multipleoccurs	Ramp rates are sorted by ascending maxima to assess recovery.
Shutdown Cost		Price	Fixed cost associated with committing a load reduction

Deleted Cells
Formatted Table
Deleted Cells
Formatted: Left
Formatted Table
Deleted Cells
Deleted Cells
Formatted Table
Formatted: Font: Bold
Formatted: Font: Bold, Highlight
Deleted Cells

7.3.2 Generation Resource Capability Description

Generation resources are very similar to load resources. As to grid effect, adding 10 MW of generation and gaining 10 MW less of load are similar.

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

For a Generation Resource, Staging Ramps are processed in order of increasing End Power. The quantity of End Power MUST be greater than the quantity of the Begin Power for each Ramp in the

Formatted: Font: 8 pt

Staging Ramp. Recovery Ramps are processed in order of decreasing End Power. The quantity of End Power MUST be less than the quantity of Begin Power for each Ramp in the Recovery Ramp.

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

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

10.3.1 Offer Curves

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

Table 10-5 ~~Registered Generation Capabilities~~

: Resource Offer Segment

~~Name~~Resource Offer Element	~~Definition~~	Note	
~~Price~~~~Mrid~~		~~mrid~~Price required for this Segment	
~~Lower Ramp Rate~~		~~dropRampRate multipleoccurs~~	~~Regulation down response rate in power units / minute~~
Maximum ~~Operating Power~~Response	~~maxOperatingPower~~	~~Resource cannot be requested to operate at higher than maximum operating power~~Enumerator for the Power rate at the beginning of this Rampaximum Power change in this segment	
~~Minimum Operating Power~~Units	~~minOperatingPower~~	~~Resource cannot be requested to operate at lower than minimum operating power~~Power Quantity at which the Ramp Ends	
~~Raise Ramp Rate~~		~~raiseRampRate multipleoccurs~~	~~Apply ramp rates consecutively to find power capabilities.~~
~~Spin Reserve Ramp~~		~~powerRamp Rate~~	

Deleted Cells
Formatted Table
Formatted: Font: Bold
Deleted Cells
Deleted Cells

~~7.3.3 Power Offer Description~~

~~The Power Offer is the most complete and generic description of a power resource, including perfoamcne and economic requirements.~~

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

10.4 Reactive Power Resources

In addition, voltage regulation services have their own semantics.

Formatted: Font: 8 pt

890 *Table 10-6 ~~Power Offer Capabilities~~Semantics for Voltage Regulation Services*

~~Name~~Voltage Regulation Element	~~Definition~~	Note	
VMin~~Mrid~~		~~mrid~~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.	
~~Startup Cost~~		~~Price~~	~~Cost to initiate any resource~~
VMax~~Minimum Resource Cost~~	~~Price~~	VMax is the IEEE 1547 maximum voltage level of 110% of nominal voltage where the PV inverter must disconnect.~~Minimum cost to elicit response from Resource~~ Also defined as the Maximum reactive power of the Resource	
~~Raise Ramp Rate~~		~~raiseRampRate multipleoccurs~~	~~Apply ramp rates consecutively to find power capabilities.~~
QMax~~Maximum Power~~	~~maxOperatingPower~~	Qmax is the inverter's current var capability and may be positive (capacitive) or negative (inductive). ~~Resource cannot be requested to operate at higher than maximum operating power~~It is the VA capability left after supporting the W demand.	
voltVar~~Minimum Operating Power~~		~~minOperatingPower~~Reactive Power	~~Resource cannot be requested to operate at lower than minimum operating power~~
~~Lower Ramp Rate~~		~~dropRampRate multipleoccurs~~	~~Apply ramp rates consecutively to find power capabilities.~~
~~Offer Segment~~		~~offerSegment~~	~~Economic requirements for incremental power, sorted by maximum power rate ascending.~~

Formatted Table
Deleted Cells
Deleted Cells
Deleted Cells
Deleted Cells
Deleted Cells
Formatted Table
Formatted: Font: 8 pt

10.5 Summary of Resource Types

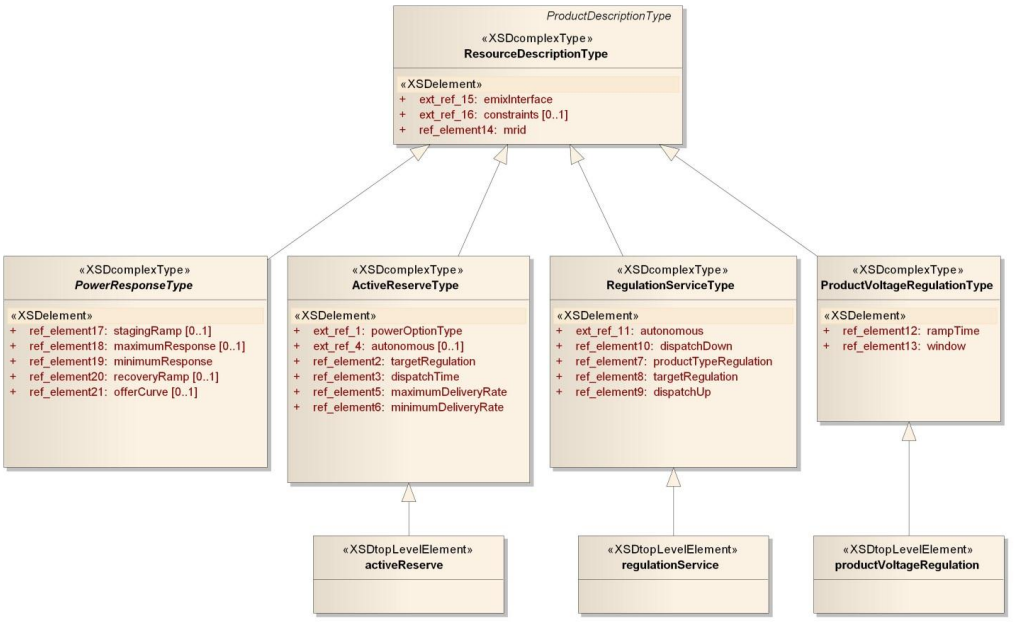


Figure 10-6: UML Summary of Resource Types

Formatted: Font: 8 pt

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.

Formatted: Font: 8 pt

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.

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

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

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

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.

Formatted: Font: 8 pt

TeMIX Element	Definition
Option Lead Time	The Minimum Notification Duration constraint
Option Schedule	The Availability Schedule constraint
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.

Formatted: Font: 8 pt

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

812 Ancillary Services ~~Products~~

Ancillary Services ~~Products~~ are typically products provided by a Resource Capability and ~~used by a system operator~~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 ~~and energy services~~ products in that they ~~must be~~are paid for availability, whether or not they perform. Of course, they must also perform when called~~. The ancillary services products described below are typical of ancillary service products defined by and procured by US ISO/RTO markets~~.

~~Ancillary Services descriptions are applied to a WS-Calendar Sequence to create the EMIX Terms used for exchange with other parties~~

~~8.1.1.1 Ancillary Services – Regulation Products~~

~~Regulation services are used to maintain accumulated frequency error within allowable bounds.~~

~~*Table 8-1 Power Regulation Product Description*~~

~~Name~~	~~Definition (Normative)~~	~~Note (Non-Normative)~~
~~**Product Type**~~	~~String, enumerated~~	~~Regulation Down Regulation Up Regulation Up & Down~~
~~**Availability Period**~~	~~ws-calendar interval~~	~~Interval during which the resource is warranted to be ready to perform.~~
~~**Autonomous Dispatch**~~	~~Bool~~	~~If true, service notes local conditions and dispatches itself. If false, it waits for dispatch request from Operator.~~
~~**Delivery Rate Units**~~	~~Typically kW or MW.~~	~~Unit is normally kilowatt-hours (kW) or megawatt-hours (MW)~~
~~**Dispatch Up**~~	~~Integer seconds~~	~~Time in which resource can respond to a request to increase energy provided. If zero, no dispatchUp available. Can also be startup delay for non-spinning reserve.~~
~~**Dispatch Down**~~	~~Integer seconds~~	~~Time in which resource can respond to a request to decrease energy provided. If zero, no dispatchDown available~~
~~**voltage**~~	~~Integer~~	~~Expressed in KV~~
~~**hertz**~~	~~Integer~~	
~~**Ac/dc**~~	~~AC or DC~~	

~~8.1.1.2 Ancillary Services Reserve Products~~

~~Ancillary Services are used for short term balancing of supply and demand by a system operator.~~

~~*Table 8-2 Reserves Product Description*~~

Formatted: Font: 8 pt

Name	Definition (Normative)	Note (Non-Normative)
Product Type	String, enumerated	Regulation Down Regulation Up Regulation Up & Down Spinning Reserve Non-Spinning Reserve
Availability Period	vcalendar	Interval during which the resource is warranted to be ready to perform
Maximum Delivery Rate	Integer	In home contracts this is limited by service size
Minimum Delivery Rate	Integer.	Determines minimum charges during period
Delivery Rate Units	Typically kWh or MWh.	Unit is normally kilowatt-hours (kWh) or megawatt-hours (MWh)
Maximum Delivery Time	Duration	When called on, for how long can this asest deliver
Cycle Time	Duration	When called on, how long until this asset can be called on again.

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.

Formatted: Font: 8 pt

913 Power Quality

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

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

9.1.113.1 Electrical Power Quality

Table 13-1: AC Power Quality

Name	DefinitionSpecification	Type	Note
Measurement Protocol	A string containing an identification of the standard or other protocol used to measure power quality	String	Text string with formal number of the standard used, e.g., "EN 50160", "IEEE 1549-2009"
Power Frequency	A floating point number describing the nominalmeasured 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.	Float	Measured rather than nominal value, e.g. 50.4, 59.9. 0 for DC
Supply Voltage Variations	An unsigned integer count of Supply Voltage Variations during the period	Float	See referenced standards for definition, measurement protocol and period. E.g., 7 in the billing period.
Rapid Voltage Changes	An unsigned integer count of Rapid Voltage Change events during the period	Ulong	See referenced standards for definition, measurement protocol and period. E.g., 0 in the billing period.
Flicker	An unsigned integer count of Flicker events during the period	Ulong	See referenced standards for definition, measurement protocol and period. E.g., 0 in the billing period.
Supply Voltage Dips	An unsigned integer count of Supply Voltage Dip events during the period	Ulong	See referenced standards for definition, measurement protocol and period. E.g., 0 in the billing period.

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Heading 2,H2, Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Deleted Cells

Deleted Cells

Formatted Table

Formatted: Font: 8 pt

Short Interruptions	An unsigned integer count of Short Interruption events during the period	~~Ulong~~	~~See referenced standards for definition, measurement protocol and period. E.g., 0 in the billing period.~~
Long Interruptions	An unsigned integer count of Long Interruption events during the period	~~Ulong~~	~~See referenced standards for definition, measurement protocol and period. E.g., 0 in the billing period.~~
Temp Overvoltage	An unsigned integer count of Temporary Overvoltage events during the period	~~Ulong~~	~~See referenced standards for definition, measurement protocol and period.~~
Supply Voltage Imbalance	An unsigned integer count of Supply Voltage Imbalance events during the period. ~~Optional, and not~~Not meaningful for DC.	~~Ulong~~	~~See referenced standards for definition, measurement protocol and period.~~
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	~~Float~~	~~See referenced standards for definition, measurement protocol and period. The period is usually much shorter than other power quality measures.~~
Mains Voltage	A floating point number Mains [Signaling] Voltage	~~Float~~	~~Nominal value, e.g, 110, 130, 220, 208. See referenced standards for definition and protocol.~~

987

Formatted: Font: 8 pt

~~10~~14 Power Transport ~~Products~~Product Descriptions

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

Transport costs affect the delivery of energy in all markets. Today's electrical power markets use different terms in transmission and delivery, but the underlying elements are the same. ~~Like the other products, aspects of transport charges may change over time, and so can be expressed as EMIX Terms by applying the Transport Description to the WS-Calendar~~ Future markets, including those for microgrids and virtual service providers, may not make the same distinctions between transmission and distribution as have been made in the past. Distributed Energy Resources (DER) may create new business models for use of the existing distribution networks. ~~**Sequence**~~.

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

Table 14-1: Transport Description

~~Name~~Transport Product Element		Definition ~~(Normative)~~	~~Note (Non-Normative)~~
Point of Receipt	~~Transaction Node~~	Where power enters a network or changes ownership	
Point of Delivery	~~Transaction Node~~	Where power exits a network or changes ownership	
Transport Access Fee	~~Price~~	Fixed Charge (not dependent on congestion) to access transport system	
Transport Congestion Fee	~~Price.~~	Congestion fee per unit of energy for energy flowing from receipt to delivery point. Can be a positive or negative price.~~ e.~~	
~~**Transport Congestion Fee Units**~~		~~Energy Units~~	
Marginal Loss Fee	~~Price~~	Marginal Loss Fee	
~~**Marginal Loss Fee Units**~~		~~Energy Units~~	
Transport Loss Factor	~~Float~~	Reduction in amount delivered due to loss during transport. (Loss Factor * purchase amount) = delivered amount	
Conversion Loss Factor	~~Float~~	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)	
~~**currency**~~		~~From CEFACT. Optional~~	~~Usually inherited, but allowed to permit stand-alone artifact~~

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Font: 10 pt, Font color: Auto

Formatted Table

Deleted Cells

Deleted Cells

Deleted Cells

Formatted: Font: 10 pt, Not Bold, Font color: Auto

Deleted Cells

Formatted: Font: 10 pt, Not Bold

Formatted: Font: 10 pt, Not Bold

Formatted: Font: 8 pt

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

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

Formatted: Font: 8 pt

~~11~~15 EMIX Warrants

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

Warrants are specific assertions about the extrinsic characteristics of ~~power~~ EMIX Products that may affect market pricing. Warrants are in effect Product artifacts as defined in EMIX. Warrants ~~start as~~ are extensions of the Product Descriptions type that are applied ~~to~~ the ~~intervals~~Intervals in a ~~sequence and to the gluon.~~Schedule. There may be zero ~~intervals~~Intervals in a ~~product~~Product if the unchanged product description applies to all.

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

~~Sometime~~Some warrants ~~are only~~ may be applicable ~~within~~ only in certain jurisdictions. For example, in ~~today's~~today's energy ~~markers (2010~~markets (2011) energy warranted as renewable in the Pacific Northwest can include ~~Hydro power~~hydropower. Energy markets in California exclude ~~Hydro Power~~hydropower from their definition of ~~Renewables. The means credits~~renewable power. Credits or mandates for renewable energy in California~~,~~ are not met by Products warranted as ~~Renewable~~renewable in the Northwest.

Some warrants may be separable from the underlying energy. ~~-~~For example, a warrant that ~~a source of~~ energy is generated by a source that is certified as "green" by an authority, may be issued a "green certificate". ~~Such a certificate can be separately traded, so the Warrant information for a product should specify if the "green certificate" is (1) accompanying the energy, (2) sold to a third party, or (3) the source is not green but a green certificate has been purchased and accompanies the energy.~~In some markets, such a certificate can be traded separately. The detailed specification of warrants is not part of version 1.0 of this specification.

~~11.1~~15.1 Warrant List Definition

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

Table 15-1: Warrant Types

Warrant ~~Element~~Type	~~Definition~~	Note	
Product Quality	~~A Product-specific assertion of 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.	
~~Warrant~~ **Environmental Warrant**		Quantifies the environmental burden created during the generation of the electric power.	~~These are as identified as per the artifact environmental.rdf~~
Content Warrant ~~Content~~		The proportion of the product defined that is from non-fossil fuel sources, including but not limited to “hydroelectric”, “nuclear”, “solar”, and “wind”.	~~The nature of the original input to storage is not altered when drawn from storage.~~
Source Warrant ~~Source~~	~~Individual source warrants~~	In aggregate may be the same as a ~~warrantContent~~Warrant Content	

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Outline numbered + Level: 2 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Indent at: 0.4"

Formatted Table

Deleted Cells

Formatted Table

Deleted Cells

Deleted Cells

Formatted: Font: 8 pt

Warrant ~~ElementType~~	~~Definition~~	Note	
~~Warrant~~ Controllability Warrant		Assertion that a ~~resource~~Resource referenced on the face of the envelope can be controlled and/or operated by or to some standard.	~~For example, some ISOs will accept a resource as direct load controllable if so asserted by a third party aggregator.~~

Deleted Cells
Formatted Table
Formatted Table
Deleted Cells

1036

Formatted: Font: 8 pt

~~12~~16 Conformance and Rules for EMIX and Referencing Specifications

~~If the first interval~~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

Formatted: Outline numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0" + Tab after: 0.3" + Indent at: 0.3"

Formatted: Font: 8 pt

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

16.1.2 Specific Attribute Inheritance in EMIX

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

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

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

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

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

- There are no exceptions.

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

In EMIX the following attributes MUST NOT be inherited

- UID (Gluons and Intervals)
- Temporal Relationships

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

16.2 Miscellaneous Business Rules not yet dealt with.

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

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

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

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

Formatted: Font: 8 pt

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)

Formatted: Font: 8 pt

B. Notes on Ancillary Services (non-normative)

Some markets, known as ancillary services, can offer substantially more for the same load than does the traditional market. Suitability of an offering for these diverse markets is determined by aspects of the response such as how fast the Product can offer the power, how long it can offer the power, how frequently the Product can offer the power, etc. Higher prices come with higher risks; the costs of non-performance in ancillary markets can be substantially higher as well.

Ancillary services require detailed interval metering. For the regulation product, 4 to 6 second interval metering and direct control of the generator is today required by the balancing system operator. However, there are current initiatives by FERC and many ISOs to allow loads and storage to provide ancillary services. One of the potential applications of the metering and communications infrastructure of the smart grid is to facilitate the participation of loads and distributed energy Products such as storage in providing balancing / ancillary services to the grid.

There is general agreement across North America on the names of ancillary services. There is general agreement on the performance profile for each ancillary service as well. There are minor differences in some of the actual performance profiles from region to region. Periodically, the performance requirements are changed for named services.

Ancillary service performance can be characterized as "meet or exceed" requirements. A given service level may meet the requirements for more than one named service. A power product that can be sold in more than one market has more potential value to the seller. Transparent service and performance requirements associated with market prices are likely to encourage sellers to make minor upgrades when they can thereby reach new markets.

For these reasons, we opted not to name the ancillary services in the standard, but instead to exchange the actual performance requirements either offered or required.

B.1 Common Requirements today

Formatted: Font: 8 pt

B. ~~For reference, here~~Extensibility and EMIX

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

B.1 Extensibility in Enumerated values

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

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

```
<xs:simpleType name="EMIXExtensionType">
   <xs:annotation>
         <xs:documentation>Pattern used for extending string enumeration,
where allowed</xs:documentation>
   </xs:annotation>
   <xs:restriction base="xs:string">
         <xs:pattern value="x-\S.*"/>
   </xs:restriction>
</xs:simpleType>
```

Non-extensible enumerated types look like this:

```
<xs:simpleType name="PowerOptionTypeEnumeratedType">
   <xs:annotation>
         <xs:documentation>Power Reserve Options</xs:documentation>
   </xs:annotation>
   <xs:restriction base="xs:string">
         <xs:enumeration value="SpinningReserve"/>
         <xs:enumeration value="NonSpinningReserve"/>
         <xs:enumeration value="OperatingReserve"/>
         <xs:enumeration value="DemandResponse"/>
   </xs:restriction>
</xs:simpleType>
```

The enumerations used in the specifications look like this.

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

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

Some extensible enumerated types are planned for extension. For example, the means of measurement for power quality are ~~some common performance requirements~~defined specific testing protocols. As of this writing, there are only two testing protocols in the specification.

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

Formatted: Font: 8 pt

We anticipate that other protocols will be used. In this case, we use ~~today. These are non-normative. They~~the suffix "EnumeratedType" to allow for the possibility of other Measurement Protocols that are ~~include here to assist the~~ not enumerated. Actual compliance, though, is based upon the type:

```
   <xs:simpleType name="MeasurementProtocolType">
         <xs:union memberTypes="power:MeasurementProtocolEnumeratedType
emix:EMIXExtensionType"/>
   </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.

```
<xs:complexType name="PowerQualityType">
   <xs:annotation>
         <xs:documentation>Power Quality consists of a number of measures,
based on contract, negotiation, and local regulation. Extend Power Quality to
incorporate new elements by creating additional elements based on
PowerQualityBaseType</xs:documentation>
   </xs:annotation>
   <xs:sequence>
   <xs:element name="measurementProtocol"
type="power:MeasurementProtocolType"/>
         <xs:element name="constraints" type="power:ArrayOfPowerQualities"/>
   </xs:sequence>
</xs:complexType>
```

A practitioner ~~in thinking~~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:

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

and the type extends the abstract base class BasePowerQualityMeaurementType:

```
<xs:complexType name="SupplyVoltageVariationsType" mixed="false">
   <xs:complexContent mixed="false">
         <xs:extension base="power:BasePowerQualityMeasurementType">
               <xs:sequence>
```

Formatted: Font: 8 pt

```
                        <xs:element name="count" type="xs:int"/>
                </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

The resulting schema, which references the approved EMIX schemas, but does not change them, can then be distributed to business partners to validate the resulting information exchanges. The core EMIX 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
- **Product Description Type**: In EMIX, the Product Description is the basis for all Resources and Product Descriptions.
- **Item Base**: Abstract base class for units for EMIX Product delivery, measurement, and warrants. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.
- **EMIX Interface**: Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

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

- **BasePowerQualityMeaurementType**: the basis for exchanging measurements of power quality
- **BaseConstraintType**: used to express constraints on the performance of equipment exposed to the market as Resources
- **BaseRequirementType**: used to express the market or business requirements of a trading partner.
- **BaseWarrantType**: the root for all warrants delivered with the energy product.

Formatted: Font: 8 pt

C. Semantics from WS-Calendar

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

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.

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

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

Formatted: Font: 8 pt

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

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

~~Regulation~~

~~Spinning Reserve~~

~~*Non-Spinning Reserve*~~*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.

Formatted: Font: 8 pt

Term	Definition
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.
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.

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

Formatted: Font: 8 pt

~~C.~~D. Electrical Power and Energy

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

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

- ~~Apparent Power:~~Apparent Energy: the production or consumption of Apparent Power over time; unit: volt-ampere hours, VAh
- Apparent Power (S): mathematical product of root-mean-square voltage and root-mean-square current, vector sum of Real Power and Reactive Power, ~~absolute value~~square root of ~~Complex~~sum of squares of Real Power~~,~~ and Reactive Power; unit: volt-ampere, VA
  - ~~Complex Power (S): square root of sum of squares of Real Power and Reactive Power, unit: volt-ampere, VA~~
- Current: flow of electric charge, or rate of flow of electric charge~~,~~; unit: ampere, A
- Energy: the production or consumption of ~~Real~~ Power over time~~, unit: Watt-hour, Wh (note: this is the contextual unit)~~.
- Power Factor (p.f.): ratio of Real Power to Complex Power, cosine of the phase angle between Current and Voltage, expressed as a number between 0 and 1, expressed as a percentage (i.e., 50% = 0.5~~),~~); unit: dimensionless
- Reactive Energy: the production or consumption of Reactive Power over time; unit: volt-ampere-reactive hours, VARh, VA-rh, varh
- Reactive Power (Q): mathematical product of the root-mean-square voltage and root-mean-square current multiplied by the sine of the angle between the voltage and current~~,~~; unit: volt-amperes reactive, VAR, VA-r, var
- Real Energy: the production or consumption of Real Power over time; unit: Watt-hour, Wh
- Real Power (P): rate at which electricity is produced or consumed, mathematical product of Voltage and Current~~,~~; unit: Watt, W
- Voltage: difference in electric potential between two points~~,~~; unit: volt, V
- Generically, the use of the term "Power" refers to "Real Power" and is expressed in Watts. Otherwise, one talks of Apparent Power in VA, or Reactive Power in VARs. Generically, the use of the term "Energy" refers to "Real Energy" and is expressed in Watt-hours. Otherwise, on talks of Apparent Energy in VAh, or Reactive Energy in VARh.

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

~~In the context of this specification, the price of Power and Energy will be expressed in the same unit, $/MWh. The argument for this comes from **[Stoft, p. 32]**. The use of Power is as a flow, and its total cost is measured in unit currency (i.e., dollars) per hour, not just unit currency. The price per unit cost of Power is measured in unit currency per hour per megawatt (MW) of Power flow, or unit currency/MWh. In the same manner, the total cost of a certain quantity of Energy is measured in unit currency. The price per unit cost of energy is measured in unit currency/MWh, which is the same as for Power.~~

Formatted: Bulleted + Level: 1 + Aligned at: 0" + Indent at: 0.5"

Formatted: Bulleted + Level: 1 + Aligned at: 0" + Indent at: 0.5"

Formatted: Bulleted + Level: 1 + Aligned at: 0" + Indent at: 0.5"

Formatted: Bulleted + Level: 1 + Aligned at: 0" + Indent at: 0.5"

Formatted: Font: 8 pt

E. ~~To clear up confusion~~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.

Formatted: Font: 8 pt

F. Schemas (Non-Normative)

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

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

F.1 EMIX Schemas

The EMIX Schema is in three parts

F.1.1 EMIX.XSD

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- emix.xsd
Schema Set for OASIS EMIX 1.0 WD23 (20110411)
Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<!-- 1.0 EMIX: Energy Market Information Exchange-->
<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:gmlsf="http://www.opengis.net/gmlsf/2.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://docs.oasis-
open.org/ns/emix" elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:include schemaLocation="emix-requirements.xsd"/>
   <xs:include schemaLocation="emix-warrants.xsd"/>
   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
spec/v1.0/csprd02/xsd/iCalendar.xsd"/>
   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
spec/v1.0/csprd02/xsd/iCalendar-wscal-extensions.xsd"/>
   <xs:import namespace="urn:ietf:params:xml:ns:icalendar-2.0"
schemaLocation="http://docs.oasis-open.org/ws-calendar/ws-calendar-
spec/v1.0/csprd02/xsd/iCalendar-availability-extension.xsd"/>
   <xs:import namespace="urn:un:unece:uncefact:codelist:standard:5:ISO42173A:2010-04-07"
schemaLocation="http://www.unece.org/uncefact/codelist/standard/ISO_ISO3AlphaCurrencyCod
e_20100407.xsd"/>
   <xs:import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
   <!-- 1.0 Core EMIX objects-->
   <xs:annotation>
            <xs:appinfo
source="http://schemas.opengis.net/gml/3.2.1/profiles/gmlsfProfile/2.0/gmlsfLevels.xsd">
                  <gmlsf:ComplianceLevel>0</gmlsf:ComplianceLevel>

   <gmlsf:GMLProfileSchema>http://schemas.opengis.net/gml/3.2.1/profiles/gmlsfProfile/2.
0/gmlsf.xsd</gmlsf:GMLProfileSchema>
            </xs:appinfo>
   </xs:annotation>
   <!-- 1.1 EMIX Product -->
   <xs:element name="product" type="emix:ProductType">
            <xs:annotation>
                  <xs:documentation>Emix Product, .i.e., a Product Description applied
to a schedule.</xs:documentation>
```


Formatted: Font: 8 pt

```
            </xs:annotation>
      </xs:element>
      <xs:complexType name="ProductType" mixed="false">
            <xs:annotation>
                  <xs:documentation>EMIX Product Type, i.e. a Product Description
applied to a Schedule</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:EmixBaseType">
                        <xs:sequence>
                              <xs:element ref="emix:currency"/>
                              <xs:element ref="emix:marketContext"/>
                              <xs:element ref="emix:transactiveState"/>
                              <xs:element ref="emix:side"/>
                              <xs:element ref="emix:constraints"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <!-- 1.2 EMIX Option -->
      <xs:element name="emixOption" type="emix:EmixOptionType">
            <xs:annotation>
                  <xs:documentation>Option to buy an Emix Product</xs:documentation>
            </xs:annotation>
      </xs:element>
      <xs:complexType name="EmixOptionType" mixed="false">
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:EmixBaseType">
                        <xs:sequence>
                              <xs:element ref="emix:transactiveState"/>
                              <xs:element name="optionExerciseSchedule"
type="emix:BusinessScheduleType"/>
                              <xs:element name="optionHolderSide"
type="emix:SideType"/>
                              <xs:element name="optionPremium"
type="emix:PriceType"/>
                              <xs:element name="optionExerciseLeadTime"
type="xcal:DurationPropType"/>
                              <xs:element name="optionStrikePrice"
type="emix:PriceType"/>
                              <xs:element ref="emix:optionType"/>
                              <xs:element ref="emix:side"/>
                              <xs:element ref="emix:marketContext"/>
                              <xs:element ref="emix:currency"/>
                              <xs:element ref="emix:constraints"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <!-- 1.3 EMIX TEMIX -->
      <xs:element name="temix" type="emix:TemixType">
            <xs:annotation>
                  <xs:documentation>minimalist Energy Market Information Exchange (EMIX)
Type</xs:documentation>
            </xs:annotation>
      </xs:element>
      <xs:complexType name="TemixType" mixed="false">
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:EmixBaseType">
                        <xs:sequence>
                              <xs:element ref="emix:currency"/>
                              <xs:element ref="emix:transactiveState"/>
                              <xs:element ref="emix:constraints"/>
                              <xs:element ref="emix:side"/>
                              <xs:element name="expires" type="xs:dateTime"
minOccurs="0" maxOccurs="1"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <!-- 1.4 Delivery -->
```

Formatted: Font: 8 pt

```
   <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  -->
```

Formatted: Font: 8 pt

```
   <xs:element name="price" type="emix:PriceType" substitutionGroup="emix:priceBase"/>
   <xs:complexType name="PriceType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Simple Price</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:PriceBaseType">
                        <xs:sequence>
                               <xs:element ref="emix:value" minOccurs="1"
maxOccurs="1"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 8.3.2 Multiplier Price - multiplier on base amount  -->
   <xs:element name="priceMultiplier" type="emix:PriceMultiplierType"
substitutionGroup="emix:priceBase"/>
   <xs:complexType name="PriceMultiplierType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Multiplier times market price, 1 for same as
market</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:PriceBaseType">
                        <xs:sequence>
                               <xs:element name="multiplier" type="xs:float"
minOccurs="1" maxOccurs="1"/>
                               <xs:element ref="emix:marketContext" minOccurs="0"
maxOccurs="1">
                                      <xs:annotation>
                                             <xs:documentation>Market Context for
base price. If blank, Market Context from hosting artifact.</xs:documentation>
                                      </xs:annotation>
                               </xs:element>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 8.3.4 Price Offset (additive or subtractive) over base amount  -->
   <xs:element name="priceRelative" type="emix:PriceRelativeType"
substitutionGroup="emix:priceBase"/>
   <xs:complexType name="PriceRelativeType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Price Relative is a fixed charge (positive or
negative) apllied to base price</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:PriceBaseType">
                        <xs:sequence>
                               <xs:element ref="emix:value" minOccurs="1"
maxOccurs="1"/>
                               <xs:element ref="emix:marketContext" minOccurs="0"
maxOccurs="1">
                                      <xs:annotation>
                                             <xs:documentation>Market Context for
base price. If blank, Market Context from hosting artifact.</xs:documentation>
                                      </xs:annotation>
                               </xs:element>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 8.3.6 Simple Price  -->
   <xs:element name="value" type="emix:valueType"/>
   <xs:simpleType name="valueType">
          <xs:restriction base="xs:decimal"/>
   </xs:simpleType>
   <!-- 8.5 Quantity -->
   <xs:element name="integralOnly" type="emix:IntegralOnlyType"/>
   <xs:simpleType name="IntegralOnlyType">
          <xs:annotation>
```

Formatted: Font: 8 pt

```
                  <xs:documentation>integralOnly is an element used in many EMIX objects
distinguishing between an (amount, response, ramp) that is all (true) or nothing
(false)</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:boolean"/>
   </xs:simpleType>
   <xs:element name="autonomous" type="emix:AutonomousType"/>
   <xs:simpleType name="AutonomousType">
            <xs:annotation>
                  <xs:documentation>An autonomous resource or service (true) is able to
respond or maintain service independently. A non autonomous service (false) must await
dispatch.</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:boolean"/>
   </xs:simpleType>
   <!-- 8.7 Enumeration for Option Types -->
   <xs:element name="optionType" type="emix:OptionTypeType"/>
   <xs:simpleType name="OptionTypeType">
            <xs:union memberTypes="emix:OptionTypeEnumeratedType
emix:EmixExtensionType"/>
   </xs:simpleType>
   <xs:simpleType name="OptionTypeEnumeratedType">
            <xs:annotation>
                  <xs:documentation>Enumerated Option Types</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:string"/>
   </xs:simpleType>
   <!-- 8.8 Performance Constraints -->
   <!-- 9.2 Abstract EMIX Base(product applied to a schedule) -->
   <xs:element name="emixBase" type="emix:EmixBaseType"/>
   <xs:complexType name="EmixBaseType" abstract="true">
            <xs:annotation>
                  <xs:documentation>iCalendar-derived object to host EMIX
elements</xs:documentation>
            </xs:annotation>
            <xs:complexContent>
                  <xs:extension base="xcal:VcalendarType">
                        <xs:sequence>
                              <xs:element ref="emix:envelopeContents" minOccurs="0"
maxOccurs="1"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
   </xs:complexType>
   <!-- 9.3 Abstract Product  Description -->
   <xs:element name="productDescription" type="emix:ProductDescriptionType"
substitutionGroup="xcal:artifactBase"/>
   <xs:complexType name="ProductDescriptionType" abstract="true">
            <xs:annotation>
                  <xs:documentation>In EMIX, the Product Description is placed in the
Interval or Gluon attachment. The respective product schemas extend this abstract
class.</xs:documentation>
            </xs:annotation>
            <xs:complexContent>
                  <xs:extension base="xcal:ArtifactBaseType"/>
            </xs:complexContent>
   </xs:complexType>
   <!-- 9.4 Interfaces  -->
   <xs:element name="serviceArea" type="emix:ServiceAreaType"
substitutionGroup="emix:emixInterface"/>
   <xs:complexType name="ServiceAreaType">
            <xs:annotation>
                  <xs:documentation>The Service Area is the geographic region that is
affected by the EMIX market condition</xs:documentation>
            </xs:annotation>
            <xs:complexContent>
                  <xs:extension base="emix:EmixInterfaceType">
                        <xs:sequence>
                              <xs:element ref="emix:geographicArea"/>
                        </xs:sequence>
                  </xs:extension>
```

Formatted: Font: 8 pt

```
            </xs:complexContent>
      </xs:complexType>
      <xs:element name="emixInterface" type="emix:EmixInterfaceType"/>
      <xs:complexType name="EmixInterfaceType" abstract="true" mixed="false">
            <xs:annotation>
                  <xs:documentation>Abstract base class for the interfaces for EMIX
Product delivery, measurement, and/or pricing</xs:documentation>
            </xs:annotation>
      </xs:complexType>
      <!-- 9.5 Geographic Area -->
      <xs:element name="geographicArea" type="emix:GeographicAreaType"
substitutionGroup="gml:AbstractFeature"/>
      <xs:complexType name="GeographicAreaType" mixed="false">
            <xs:annotation>
                  <xs:documentation>A service area is a geographic region that may be
affected by the same EMIX market condition.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="gml:AbstractFeatureType">
                        <xs:sequence>
                              <xs:element name="foo"
type="gml:GeometryPropertyType"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <!-- 9.6 Business Schedule  -->
      <xs:element name="businessSchedule" type="emix:BusinessScheduleType"/>
      <xs:complexType name="BusinessScheduleType" mixed="false">
            <xs:annotation>
                  <xs:documentation>iCalendar-derived business schedule, more variant
than allowed in sequences</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="xcal:VavailabilityType"/>
            </xs:complexContent>
      </xs:complexType>
      <xs:element name="duration" type="emix:DurationType"/>
      <xs:complexType name="DurationType" mixed="false">
            <xs:annotation>
                  <xs:documentation>iCalendar-derived duration. THis brings in
xcal:duration and xcal:parameters. </xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="xcal:DurationPropType"/>
            </xs:complexContent>
      </xs:complexType>
      <!-- 9.7 emix Items -->
      <xs:element name="measurement" type="emix:MeasurementType"
substitutionGroup="emix:productDescription"/>
      <xs:complexType name="MeasurementType">
            <xs:annotation>
                  <xs:documentation>Type of Measurement</xs:documentation>
            </xs:annotation>
            <xs:complexContent>
                  <xs:extension base="emix:ProductDescriptionType">
                        <xs:sequence>
                              <xs:element ref="emix:quantity"/>
                              <xs:element ref="emix:itemBase"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <xs:element name="emixGranularity" type="emix:EmixGranularityType"/>
      <xs:complexType name="EmixGranularityType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Abstract base class used for graularity of market
indications of interest and tenders</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element ref="emix:quantity"/>
```

Formatted: Font: 8 pt

```
                <xs:element ref="emix:itemBase"/>
            </xs:sequence>
    </xs:complexType>
    <xs:element name="itemBase" type="emix:ItemBaseType" abstract="true"/>
    <xs:complexType name="ItemBaseType" abstract="true" mixed="false">
            <xs:annotation>
                <xs:documentation>Abstract base class for units for pricing, refer to
definitions on pp. 30-33 in [Stoft].EMIX Product delivery, measurement, and warrants.
Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include
Quantity or Price, because a single product description or transaction may have multiple
quantities or prices associated with a single item.</xs:documentation>
            </xs:annotation>
    </xs:complexType>
    <!-- 9.8 Units and Measurement Abstractions -->
    <xs:element name="quantity" type="emix:QuantityType"/>
    <xs:simpleType name="QuantityType">
            <xs:annotation>
                <xs:documentation>Base type for all quanties in
EMIX.</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:float"/>
    </xs:simpleType>
    <xs:element name="scale" type="emix:SiScaleType"/>
    <xs:simpleType name="SiScaleType">
            <xs:annotation>
                <xs:documentation>Scale based on representations of SI scale as
expressed in the unit multipliers defined for the CIM</xs:documentation>
                <xs:documentation xml:lang="en">enumeration</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:string">
                <xs:enumeration value="n">
                    <xs:annotation>
                        <xs:documentation>Nano 10**-9</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="micro">
                    <xs:annotation>
                        <xs:documentation>Micro 10**-6</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="m">
                    <xs:annotation>
                        <xs:documentation>Milli 10**-3</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="c">
                    <xs:annotation>
                        <xs:documentation>Centi 10**-2</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="d">
                    <xs:annotation>
                        <xs:documentation>Deci 10**-1</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="k">
                    <xs:annotation>
                        <xs:documentation>Kilo 10**3</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="M">
                    <xs:annotation>
                        <xs:documentation>Mega 10**6</xs:documentation>
                        <xs:documentation xml:lang="en">enum</xs:documentation>
                    </xs:annotation>
```

Formatted: Example small

Formatted: Font: 8 pt

```
                </xs:enumeration>
                <xs:enumeration value="G">
                        <xs:annotation>
                                <xs:documentation>Giga 10**9</xs:documentation>
                                <xs:documentation xml:lang="en">enum</xs:documentation>
                        </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="T">
                        <xs:annotation>
                                <xs:documentation>Tera 10**12</xs:documentation>
                                <xs:documentation xml:lang="en">enum</xs:documentation>
                        </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="none">
                        <xs:annotation>
                                <xs:documentation xml:lang="en">enum</xs:documentation>
                        </xs:annotation>
                </xs:enumeration>
                <xs:enumeration value="p">
                        <xs:annotation>
                                <xs:documentation>Pico 10**-12</xs:documentation>
                                <xs:documentation xml:lang="en">enum</xs:documentation>
                        </xs:annotation>
                </xs:enumeration>
        </xs:restriction>
   </xs:simpleType>
   <!-- 9.9 Extension Type  -->
   <xs:simpleType name="EmixExtensionType">
        <xs:annotation>
                <xs:documentation>Pattern used for extending string enumeration, where
allowed</xs:documentation>
        </xs:annotation>
        <xs:restriction base="xs:string">
                <xs:pattern value="x-\S.*"/>
        </xs:restriction>
   </xs:simpleType>
</xs:schema>
```

F.1.2 EMIX-Requirements

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- emix.xsd
Schema Set for OASIS EMIX 1.0 WD23 (20110411)
Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<!-- 8.9 Constraints & Requirements -->
<xs:schema xmlns:emix="http://docs.oasis-open.org/ns/emix"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
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" targetNamespace="http://docs.oasis-
open.org/ns/emix" elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:include schemaLocation="emix.xsd"/>
   <xs:element name="constraints" type="emix:ConstraintsType"/>
   <xs:complexType name="ConstraintsType">
        <xs:annotation>
                <xs:documentation>Constraints are extrinsic to the product delivery
but effect how a partner may request performance of a service. Performance constraints
may be tied to 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. It is
possible for a given underlying resource to be offered to the market with different
constraints and therefor different values.</xs:documentation>
        </xs:annotation>
```

Formatted: Font: 8 pt

```
            <xs:sequence>
                  <xs:element name="constraints" type="emix:ArrayOfConstraints"/>
                  <xs:element name="requirements" type="emix:ArrayOfRequirements"/>
            </xs:sequence>
   </xs:complexType>
   <!-- 8.9.1 Core EMIX Constraints-->
   <xs:element name="baseConstraint" type="emix:BaseConstraintType" abstract="true">
            <xs:annotation>
                  <xs:documentation>Abstract extension object for Emix
Constraints</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:complexType name="ArrayOfConstraints">
            <xs:annotation>
                  <xs:documentation>Collection of Emix Constraints</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element ref="emix:baseConstraint" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
   </xs:complexType>
   <xs:complexType name="BaseConstraintType" abstract="true">
            <xs:annotation>
                  <xs:documentation>Type of Abstract extension object for Emix
Constraints</xs:documentation>
            </xs:annotation>
   </xs:complexType>
   <xs:element name="minimumResponseDuration" type="emix:MinimumResponseDurationType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The shortest Duration for which the resource will
accept a request  to maintain a response before returning to pre-request
levels.</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="maximumResponseDuration" type="emix:MaximumResponseDurationType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The longest Duration for which the resource will
accept a request.</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="minimumRecoveryDuration" type="emix:MinimumRecoveryDurationType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The minimum Duration that the Resource requires
after the end of a response the resource has is ready to respond to a new
request.</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="minimumDurationBetweenInvocations"
type="emix:MinimumDurationBetweenInvocationsType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The minimum Duration that the Resource requires
after receiving a request before the resource has is ready to respond to a new
request.</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="minimumNotificationDuration"
type="emix:MinimumNotificationDurationType" substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The minimum Duration that the Resource requires for
Notification before initiating a response to a request.</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="maximumNotificationDuration"
type="emix:MaximumNotificationDurationType" substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>The maximum Duration in advance of a requested
response that the resource is willing to accept a request.</xs:documentation>
```

Formatted: Font: 8 pt

```
            </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>
```

Formatted: Font: 8 pt

```
   <!-- 8.9.1.1 Minimum Response-->
   <xs:complexType name="MinimumResponseDurationType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of the shortest Duration for which the resource
will accept a request  to maintain a response before returning to pre-request
levels.</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>
   <!-- 8.9.1.2 Maximum Response-->
   <xs:complexType name="MaximumResponseDurationType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of the longest Duration for which the resource
will accept a request.</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="MinimumRecoveryDurationType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of the minimum Duration that the Resource
requires after the end of a response the resource has is ready to respond to a new
request.</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="MinimumDurationBetweenInvocationsType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of the minimum Duration that the Resource
requires after receiving a request before the resource has is ready to respond to a new
request.</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="MinimumNotificationDurationType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of the minimum Duration that the Resource
requires for Notification before initiating a response to a request.</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseConstraintType">
                        <xs:sequence>
                               <xs:element ref="emix:duration" minOccurs="1"
maxOccurs="1"/>
```

Formatted: Font: 8 pt

```
                        </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>
```

Formatted: Font: 8 pt

```
                <xs:documentation>Type of 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:complexContent mixed="false">
                <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="MaximumRunDurationType" mixed="false">
            <xs:annotation>
                <xs:documentation>Type of the Maximum duration for which a resource
will accept a request</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="MinimumRunDurationType" mixed="false">
            <xs:annotation>
                <xs:documentation>Type of the Minimum duration for which a resource
will accept a request</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>
    <!-- Business Schedules -->
    <xs:complexType name="AvailabilityScheduleType" mixed="false">
            <xs:annotation>
                <xs:documentation>Type of the schedule of time for 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:complexContent mixed="false">
                <xs:extension base="emix:BaseConstraintType">
                    <xs:sequence>
                        <xs:element ref="emix:businessSchedule"
maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="NotificationScheduleType" mixed="false">
            <xs:annotation>
                <xs:documentation>Type of the schedule of time during which a resource
will accept requests. The schedule may include multiple notofication windows. The
request must occur within a notification window.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                <xs:extension base="emix:BaseConstraintType">
                    <xs:sequence>
                        <xs:element ref="emix:businessSchedule"
maxOccurs="unbounded"/>
                    </xs:sequence>
```

Formatted: Font: 8 pt

```
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <xs:complexType name="UnavailabilityScheduleType" mixed="false">
            <xs:annotation>
                <xs:documentation>Type of the schedule of time for 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:complexContent mixed="false">
                <xs:extension base="emix:BaseConstraintType">
                    <xs:sequence>
                        <xs:element ref="emix:businessSchedule"
maxOccurs="unbounded"/>
                    </xs:sequence>
                </xs:extension>
            </xs:complexContent>
        </xs:complexType>
        <!-- 8.2 EMIX Requirements-->
        <xs:element name="marketRequirements" type="emix:MarketRequirementsType">
            <xs:annotation>
                <xs:documentation>Market Requirements are the market portion of
Constraints, i.e., they are used to state the offeror's expectations about a tender.It
is possible for a given underlying resource to be offered to the market with different
Requirements and therefor different values.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:complexType name="MarketRequirementsType">
            <xs:annotation>
                <xs:documentation>Market Requirements are the market portion of
Constraints, i.e., they are used to state the offeror's expectations about a tender.It
is possible for a given underlying resource to be offered to the market with different
Requirements and therefor different values.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                <xs:element name="requirements" type="emix:ArrayOfRequirements"/>
            </xs:sequence>
        </xs:complexType>
        <xs:complexType name="BaseRequirementType" abstract="true">
            <xs:annotation>
                <xs:documentation>Abstract base for all
Requirements</xs:documentation>
            </xs:annotation>
        </xs:complexType>
        <xs:element name="baseRequirement" type="emix:BaseRequirementType">
            <xs:annotation>
                <xs:documentation>Abstract base for all
Requirements</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:complexType name="ArrayOfRequirements">
            <xs:annotation>
                <xs:documentation>Abstract base for a collection of
requirements</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                <xs:element ref="emix:baseRequirement" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
        <xs:element name="minimumEconomicRequirement"
type="emix:MinimumEconomicRequirementType" substitutionGroup="emix:baseRequirement">
            <xs:annotation>
                <xs:documentation> Minimum net remuneration this resource requires
from a total response</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="requiredStartupCost" type="emix:RequiredStartupCostType"
substitutionGroup="emix:baseRequirement">
```

Formatted: Font: 8 pt

```
            <xs:annotation>
                  <xs:documentation>Minimum remunuration required from start-up of this
service. </xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:element name="minimumResourceCost" type="emix:MinimumResourceCostType"
substitutionGroup="emix:baseRequirement">
            <xs:annotation>
                  <xs:documentation>Resource requires this amount per period, i.e., a
minimum requirement for $100 / hour at whatever rate</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:complexType name="MinimumEconomicRequirementType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Minimum net remuneration this resource requires from
a total response</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseRequirementType">
                        <xs:sequence>
                              <xs:element ref="emix:price"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="RequiredStartupCostType" mixed="false">
            <xs:annotation>
                  <xs:documentation> Minimum remunuration required from start-up of this
service. </xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseRequirementType">
                        <xs:sequence>
                              <xs:element ref="emix:price" minOccurs="1"
maxOccurs="1"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
   </xs:complexType>
   <!-- 4.3.2 Minimum Resource Cost  -->
   <xs:complexType name="MinimumResourceCostType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Resource requires this amount per period, i.e., a
minimum requirement for $100 / hour at whatever rate</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseRequirementType">
                        <xs:sequence>
                              <xs:element ref="emix:price"/>
                              <xs:element ref="emix:duration"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
   </xs:complexType>
</xs:schema>
```

F.1.3 EMIX Warrants

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- emix.xsd
Schema Set for OASIS EMIX 1.0 WD23 (20110411)
Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
```


Formatted: Font: 8 pt

```
<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>
```

Formatted: Font: 8 pt

```
                              <xs:element ref="emix:product" minOccurs="1"
maxOccurs="unbounded"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="EnvironmentalWarrantType" abstract="true" mixed="false">
            <xs:annotation>
                  <xs:documentation>An Environmental Warrant asserts what environmental
cost was created by the product.</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseWarrantType">
                        <xs:sequence>
                              <xs:element ref="emix:product" minOccurs="1"
maxOccurs="unbounded"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="SourceWarrantType" abstract="true" mixed="false">
            <xs:annotation>
                  <xs:documentation>A source warrant consists of assertions about
through what proces energy originated. </xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseWarrantType">
                        <xs:sequence>
                              <xs:element ref="emix:product" minOccurs="1"
maxOccurs="unbounded"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="ContentWarrantType" abstract="true" mixed="false">
            <xs:annotation>
                  <xs:documentation>A content warrant consists of assertions about where
energy originated. </xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseWarrantType">
                        <xs:sequence>
                              <xs:element ref="emix:product" minOccurs="1"
maxOccurs="unbounded"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="ControllabilityWarrantType" abstract="true" mixed="false">
            <xs:annotation>
                  <xs:documentation>A Controllability Warrant makes certifies that the
resource is controolable byt the market buyer. </xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:BaseWarrantType">
                        <xs:sequence>
                              <xs:element ref="emix:product" minOccurs="1"
maxOccurs="unbounded"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
</xs:schema>
```

F.2 Power Schemas

The Power Schema is in 3 parts

Formatted: Font: 8 pt

F.2.1 Power.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)

Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
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/power"
elementFormDefault="unqualified" attributeFormDefault="unqualified">
   <xs:include schemaLocation="power-products.xsd"/>
   <xs:include schemaLocation="power-quality.xsd"/>
   <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
   <!-- 1.0 Core EMIX Power objects-->
   <!-- 1.1 Power Product-->
   <!-- 1.2 Reserves and Power Options-->
   <!-- 2.0  Contract Power Products -->
   <!-- 4.0 Resource Semantics -->
   <!-- 6.0 Power Quality -->
   <!-- 9.1.2 Unit Energy Price -->
   <xs:element name="unitEnergyPrice" type="power:UnitEnergyPriceType"/>
   <xs:complexType name="UnitEnergyPriceType">
          <xs:annotation>
                 <xs:documentation>Price per Unit of Energy, i.e., Power times
Duration</xs:documentation>
          </xs:annotation>
          <xs:sequence>
                 <xs:element ref="emix:priceBase"/>
                 <xs:element ref="power:energyItem"/>
          </xs:sequence>
   </xs:complexType>
   <xs:element name="energyQuantity" type="power:EnergyQuantityType"/>
   <xs:complexType name="EnergyQuantityType">
          <xs:annotation>
                 <xs:documentation>Level of Energy</xs:documentation>
          </xs:annotation>
          <xs:sequence>
                 <xs:element ref="emix:quantity"/>
                 <xs:element ref="power:energyItem"/>
          </xs:sequence>
   </xs:complexType>
   <!-- 9.1.3 Power Delivery Rate -->
   <xs:element name="powerQuantity" type="power:PowerQuantityType"/>
   <xs:complexType name="PowerQuantityType">
          <xs:annotation>
                 <xs:documentation>Quantity of Power</xs:documentation>
          </xs:annotation>
          <xs:sequence>
                 <xs:element ref="emix:quantity"/>
                 <xs:element ref="power:powerItem"/>
          </xs:sequence>
   </xs:complexType>
   <!-- 9.1.5 Reactive Power -->
   <xs:element name="varQuantity" type="power:VarQuantityType"/>
   <xs:complexType name="VarQuantityType">
          <xs:sequence>
                 <xs:element ref="emix:quantity"/>
                 <xs:element ref="power:powerReactive"/>
          </xs:sequence>
   </xs:complexType>
   <!-- 9.8 Interface Types -->
   <!-- 9.8.1 EndDevices -->
```

Formatted: Font: 8 pt

```
   <!-- updated name of this section to reflect the more generic EndDevice rather than
Meters specifically -->
   <xs:element name="endDeviceAsset" type="power:EndDeviceAssetType"
substitutionGroup="emix:emixInterface">
          <xs:annotation>
                 <xs:documentation>One type of EndDeviceAsset is a MeterAsset which can
perform metering, load management, connect/disconnect, accounting functions, etc. Some
EndDeviceAssets may be connected to a MeterAsset.</xs:documentation>
          </xs:annotation>
   </xs:element>
   <xs:complexType name="EndDeviceAssetType">
          <xs:annotation>
                 <xs:documentation>The EndDeviceAssets are the physical device or
devices which could be meters or other types of devices that may be of
interest</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="emix:EmixInterfaceType">
                        <xs:sequence>
                               <xs:element ref="power:mRID"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 9.8.1.1 Meters -->
   <xs:element name="meterAsset" type="power:MeterAssetType"
substitutionGroup="emix:emixInterface"/>
   <xs:complexType name="MeterAssetType">
          <xs:annotation>
                 <xs:documentation>The MeterAsset is the physical device or devices
that performs the role of the meter</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="emix:EmixInterfaceType">
                        <xs:sequence>
                               <xs:element ref="power:mRID"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 9.8.2 Nodes -->
   <xs:element name="pnode" type="power:PnodeType"
substitutionGroup="emix:emixInterface"/>
   <xs:complexType name="PnodeType" mixed="false">
          <xs:annotation>
                 <xs:documentation>A pricing node is directly associated with a
connectivity node.  It is a pricing location for which market participants submit their
bids, offers, buy/sell CRRs, and settle.</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:EmixInterfaceType">
                        <xs:sequence>
                               <xs:element ref="power:node"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <xs:element name="aggregatedPnode" type="power:AggregatedPnodeType"
substitutionGroup="emix:emixInterface"/>
   <xs:complexType name="AggregatedPnodeType" mixed="false">
          <xs:annotation>
                 <xs:documentation>An aggregated pricing node is a specialized type of
pricing node used to model items such as System Zone, Default Price Zone, Custom Price
Zone, Control Area, Aggregated Generation, Aggregated Particpating Load, Aggregated Non-
Participating Load, Trading Hub, DCA Zone</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:EmixInterfaceType">
                        <xs:sequence>
                               <xs:element ref="power:node"/>
                        </xs:sequence>
```

Formatted: Font: 8 pt

```
                </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">
```

Formatted: Font: 8 pt

```
            <xs:annotation>
                  <xs:documentation>The mRID identifies the physical device that may be
a CustomerMeter or other types of EndDevices.</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:string"/>
      </xs:simpleType>
      <!-- 9.9 Enumerations -->
      <!--  9.9.1 Voltage   -->
      <xs:element name="voltage" type="power:VoltageType"
substitutionGroup="emix:itemBase"/>
      <xs:complexType name="VoltageType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Voltage</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:extension base="emix:ItemBaseType">
                        <xs:sequence>
                              <xs:element name="itemDescription" type="xs:string"
fixed="Voltage"/>
                              <xs:element name="itemUnits" type="xs:string"
fixed="V"/>
                              <xs:element ref="emix:scale"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
      </xs:complexType>
      <!-- 9.9.2 Energy Units -->
      <xs:element name="energyApparent" type="power:EnergyApparentType"
substitutionGroup="power:energyItem"/>
      <xs:complexType name="EnergyApparentType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Apparent Energy, measured in volt-ampere hours
(VAh)</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:restriction base="power:EnergyItemType">
                        <xs:sequence>
                              <xs:element name="itemDescription" type="xs:string"
fixed="ApparentEnergy"/>
                              <xs:element name="itemUnits" type="xs:string"
fixed="VAh"/>
                              <xs:element name="scale" type="emix:SiScaleType"/>
                        </xs:sequence>
                  </xs:restriction>
            </xs:complexContent>
      </xs:complexType>
      <xs:element name="energyReactive" type="power:EnergyReactiveType"
substitutionGroup="power:energyItem"/>
      <xs:complexType name="EnergyReactiveType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Reactive Energy, volt-amperes reactive hours
(VARh)</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                  <xs:restriction base="power:EnergyItemType">
                        <xs:sequence>
                              <xs:element name="itemDescription" type="xs:string"
fixed="ReactiveEnergy"/>
                              <xs:element name="itemUnits" type="xs:string"
fixed="VARh"/>
                              <xs:element name="scale" type="emix:SiScaleType"/>
                        </xs:sequence>
                  </xs:restriction>
            </xs:complexContent>
      </xs:complexType>
      <xs:element name="energyReal" type="power:EnergyRealType"
substitutionGroup="power:energyItem"/>
      <xs:complexType name="EnergyRealType" mixed="false">
            <xs:annotation>
                  <xs:documentation>Real Energy, Watt Hours (Wh)</xs:documentation>
            </xs:annotation>
```

Formatted: Font: 8 pt

```
            <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"/>
```

Formatted: Font: 8 pt

```
                                    <xs:element name="scale" type="emix:SiScaleType"/>
                                    <xs:element ref="power:powerAttributes"/>
                            </xs:sequence>
                    </xs:restriction>
            </xs:complexContent>
    </xs:complexType>
    <!-- ================================================== -->
    <xs:element name="powerReal" type="power:PowerRealType"
substitutionGroup="power:powerItem"/>
    <xs:complexType name="PowerRealType" mixed="false">
            <xs:annotation>
                    <xs:documentation>Real power measured in Watts (W) or Joules/second
(J/s)</xs:documentation>
            </xs:annotation>
            <xs:complexContent mixed="false">
                    <xs:restriction base="power:PowerItemType">
                            <xs:sequence>
                                    <xs:element name="itemDescription" type="xs:string"
fixed="RealPower"/>
                                    <xs:element name="itemUnits">
                                            <xs:simpleType>
                                                    <xs:restriction base="xs:token">
                                                            <xs:enumeration value="W"/>
                                                            <xs:enumeration value="J/s"/>
                                                    </xs:restriction>
                                            </xs:simpleType>
                                    </xs:element>
                                    <xs:element name="scale" type="emix:SiScaleType"/>
                                    <xs:element ref="power:powerAttributes"/>
                            </xs:sequence>
                    </xs:restriction>
            </xs:complexContent>
    </xs:complexType>
    <!-- ================================================== -->
    <!-- 9.9.5 Base Power Item Type -->
    <!-- ================================================== -->
    <xs:element name="powerItem" type="power:PowerItemType"
substitutionGroup="emix:itemBase"/>
    <xs:complexType name="PowerItemType" abstract="true" mixed="false">
            <xs:annotation>
                    <xs:documentation>Base for the measurement of Power</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:element ref="power:powerAttributes"/>
                            </xs:sequence>
                    </xs:extension>
            </xs:complexContent>
    </xs:complexType>
    <!-- ================================================== -->
    <xs:element name="powerAttributes" type="power:PowerAttributesType"/>
    <xs:complexType name="PowerAttributesType">
            <xs:sequence>
                    <xs:element name="hertz" type="xs:decimal"/>
                    <xs:element name="voltage" type="xs:decimal"/>
                    <xs:element name="ac" type="xs:boolean"/>
            </xs:sequence>
    </xs:complexType>
    <!-- 9.9.5 Enumeration for Reserves and other Power Options -->
    <xs:element name="powerOptionType" type="power:PowerOptionTypeType"/>
    <xs:simpleType name="PowerOptionTypeType">
            <xs:union memberTypes="power:PowerOptionTypeEnumeratedType
emix:EmixExtensionType"/>
    </xs:simpleType>
    <xs:simpleType name="PowerOptionTypeEnumeratedType">
            <xs:annotation>
                    <xs:documentation>Power Reserve Options</xs:documentation>
```

Formatted: Font: 8 pt

```
            </xs:annotation>
            <xs:restriction base="xs:string">
                  <xs:enumeration value="SpinningReserve"/>
                  <xs:enumeration value="NonSpinningReserve"/>
                  <xs:enumeration value="OperatingReserve"/>
                  <xs:enumeration value="DemandResponse"/>
            </xs:restriction>
   </xs:simpleType>
</xs:schema>
```

F.2.2 Power Quality

Demonstrates extensibility of base Warrant classes, as well.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)

Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
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/power"
elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:include schemaLocation="power.xsd"/>
   <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
   <!-- 6.0 Quality Warrants -->
   <xs:element name="powerQualityWarrant" type="power:PowerQualityWarrantType"
substitutionGroup="emix:baseWarrant"/>
   <xs:complexType name="PowerQualityWarrantType" mixed="false">
            <xs:annotation>
                  <xs:documentation>A Power 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>
                              <xs:element name="measurementProtocol"
type="power:MeasurementProtocolType"/>
                              <xs:element name="constraints"
type="power:ArrayOfPowerQualities"/>
                        </xs:sequence>
                  </xs:extension>
            </xs:complexContent>
   </xs:complexType>
   <!-- 6.1 Power Quality -->
   <xs:element name="powerQuality" type="power:PowerQualityType">
            <xs:annotation>
                  <xs:documentation>Power Quality warrant</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:complexType name="PowerQualityType">
            <xs:annotation>
                  <xs:documentation>Power Quality consists of a number of measures,
based on contract, negotiation, and local regulation. Extend Power Qulity to incorporate
new elements by creating additional elements based on
PowerQualityBaseType</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="measurementProtocol"
type="power:MeasurementProtocolType"/>
                  <xs:element name="constraints" type="power:ArrayOfPowerQualities"/>
            </xs:sequence>
   </xs:complexType>
```

Formatted: Font: 8 pt

```
   <xs:element name="basePowerQualityMeasurement"
type="power:BasePowerQualityMeasurementType" abstract="true">
          <xs:annotation>
                 <xs:documentation>Abstract extension object for Power
Qualities</xs:documentation>
          </xs:annotation>
   </xs:element>
   <xs:complexType name="ArrayOfPowerQualities">
          <xs:annotation>
                 <xs:documentation>Collection of Power Qualities</xs:documentation>
          </xs:annotation>
          <xs:sequence>
                 <xs:element ref="power:basePowerQualityMeasurement" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
   </xs:complexType>
   <xs:complexType name="BasePowerQualityMeasurementType" abstract="true">
          <xs:annotation>
                 <xs:documentation>An identification of the standard or other protocol
used to measure power quality. Sets definition for all other power attributes. Type of
Abstract extension object  for Power Qualities</xs:documentation>
          </xs:annotation>
   </xs:complexType>
   <!-- 6.1 Defined Power Qualities -->
   <xs:element name="powerFrequency" type="power:PowerFrequencyType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="supplyVoltageVariations" type="power:SupplyVoltageVariationsType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="rapidVoltageChanges" type="power:RapidVoltageChangesType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="flicker" type="power:FlickerType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="supplyVoltageDips" type="power:SupplyVoltageDipsType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="shortInterruptions" type="power:ShortInterruptionsType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="longInterruptions" type="power:LongInterruptionsType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="temporaryOvervoltage" type="power:TemporaryOvervoltageType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="supplyVoltageImbalance" type="power:SupplyVoltageImbalanceType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="harmonicVoltage" type="power:HarmonicVoltageType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <xs:element name="mainsVoltage" type="power:MainsVoltageType"
substitutionGroup="power:basePowerQualityMeasurement"/>
   <!-- 6.2 Defines Power Quality Measuress -->
   <xs:complexType name="PowerFrequencyType" mixed="false">
          <xs:annotation>
                 <xs:documentation>measured Power frequency, e.g. 50.4, 59.9, ,
measured as per referenced measurement protocol. 0 for DC
                                </xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BasePowerQualityMeasurementType">
                        <xs:sequence>
                               <xs:element name="frequency" type="xs:float"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="SupplyVoltageVariationsType" mixed="false">
          <xs:annotation>
                 <xs:documentation>count of Supply Voltage Variations 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"/>
```

Formatted: Font: 8 pt

```
                        </xs:sequence>
                </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="RapidVoltageChangesType" mixed="false">
        <xs:annotation>
                <xs:documentation>count of Rapid Voltage Changes 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="FlickerType" mixed="false">
        <xs:annotation>
                <xs:documentation>count of Flicker 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="SupplyVoltageDipsType" mixed="false">
        <xs:annotation>
                <xs:documentation>count of Supply Voltage Dips 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="ShortInterruptionsType" mixed="false">
        <xs:annotation>
                <xs:documentation>count of Short Interruptions 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="LongInterruptionsType" mixed="false">
        <xs:annotation>
                <xs:documentation>count of Long Interuptions 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>
```

Formatted: Font: 8 pt

```
                </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">
```

Formatted: Font: 8 pt

```
                <xs:enumeration value="EN 50160"/>
                <xs:enumeration value="IEEE 1549-2009"/>
            </xs:restriction>
        </xs:simpleType>
</xs:schema>
```

F.2.3 Power Products.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- power-quality.xsd - Power Products for OASIS EMIX 1.0 WD23 (20110411)

Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)

This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<xs:schema xmlns:power="http://docs.oasis-open.org/ns/emix/power"
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/power"
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <xs:include schemaLocation="power.xsd"/>
    <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
    <!-- 2.0 Power Products -->
    <xs:element name="powerProductDescription" type="power:PowerProductDescriptionType"
substitutionGroup="emix:productDescription"/>
    <xs:complexType name="PowerProductDescriptionType" abstract="true">
        <xs:annotation>
            <xs:documentation>Type of Product Description for simple transactions.
Also used as template for other Power Product Description Types. A product is advertised
(or bought) with a constant power, which dictates the rate of delivery. After a
specified duration, energy has been delivered, at a price per energy, price per unit
energy</xs:documentation>
        </xs:annotation>
        <xs:complexContent>
            <xs:extension base="emix:ProductDescriptionType">
                <xs:sequence>
                    <xs:element ref="power:productType"/>
                    <xs:element ref="emix:emixInterface"/>
                    <xs:element ref="power:unitEnergyPrice"/>
                    <xs:element ref="power:powerItem"/>
                    <xs:element name="charges"
type="power:ArrayOfCharges"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!-- 2.1 Full Requirements Power -->
    <xs:element name="fullRequirementsPower" type="power:FullRequirementsPowerType"
substitutionGroup="power:powerProductDescription"/>
    <xs:complexType name="FullRequirementsPowerType">
        <xs:annotation>
            <xs:documentation>Type of Product Description for Supplier to provide
for full requirements of buyer. Simple prices, will supply all used. Demand Charges
Optional. Often used in retail residential rates.</xs:documentation>
        </xs:annotation>
        <xs:complexContent>
            <xs:extension base="power:PowerProductDescriptionType">
                <xs:sequence>
                    <xs:element ref="emix:priceBase"/>
                    <xs:element name="maximumPower"
type="emix:QuantityType"/>
                    <xs:element name="minimumPower"
type="emix:QuantityType"/>
                </xs:sequence>
            </xs:extension>
```

Formatted: Font: 8 pt

```
            </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>
```

Formatted: Font: 8 pt

```
   </xs:complexType>
   <!--
================================================================================
====== -->
   <!-- Charge Defintions -->
   <!--
================================================================================
====== -->
   <!-- 2.5 Charge Abstractions -->
   <xs:element name="baseCharge" type="power:BaseChargeType" abstract="true">
            <xs:annotation>
                     <xs:documentation>Abstract extension object for Emix Power Product
Charges</xs:documentation>
            </xs:annotation>
   </xs:element>
   <xs:complexType name="ArrayOfCharges">
            <xs:annotation>
                     <xs:documentation>Collection of Emix Power Product
Charges</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                     <xs:element ref="power:baseCharge" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
   </xs:complexType>
   <xs:complexType name="BaseChargeType" abstract="true">
            <xs:annotation>
                     <xs:documentation>Type of Abstract extension object for  Emix Power
Product Charges</xs:documentation>
            </xs:annotation>
   </xs:complexType>
   <!--
================================================================================
====== -->
   <!-- 2.6 General Charges -->
   <!-- 2.6.1 Blocks for use in Block Power -->
   <xs:element name="blockPowerPrice" type="power:BlockPowerPriceType"
substitutionGroup="power:baseCharge"/>
   <xs:complexType name="BlockPowerPriceType" mixed="false">
            <xs:complexContent mixed="false">
                     <xs:extension base="power:BaseChargeType">
                              <xs:sequence>
                                       <xs:element ref="emix:priceBase"/>
                                       <xs:element name="maximumEnergyQuantity"
type="emix:QuantityType"/>
                              </xs:sequence>
                     </xs:extension>
            </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="ArrayOfBlockPowerPrices">
            <xs:annotation>
                     <xs:documentation>Collection of Emix Block Power
Prices</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                     <xs:element ref="power:blockPowerPrice" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
   </xs:complexType>
   <!--
================================================================================
====== -->
   <!-- 2.6.2 Demand Charges -->
   <xs:element name="demandCharge" type="power:DemandChargeType"
substitutionGroup="power:baseCharge"/>
   <xs:complexType name="DemandChargeType" mixed="false">
            <xs:complexContent mixed="false">
                     <xs:extension base="power:BaseChargeType">
                              <xs:sequence>
                                       <xs:element name="demandChargeUnits"
type="power:PowerItemType"/>
```

Formatted: Font: 8 pt

```
                                <xs:element name="demandChargeFloor"
type="emix:QuantityType"/>
                                <xs:element name="demandChargeRate"
type="emix:PriceBaseType"/>
                                <xs:element name="measurementInterval"
type="emix:DurationType"/>
                                <xs:element name="collectionInterval"
type="emix:DurationType"/>
                                <xs:element name="collectionPeriod"
type="emix:DurationType"/>
                                <xs:element name="chargeDuration"
type="emix:DurationType"/>
                        </xs:sequence>
                </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--
================================================================================
====== -->
    <!-- Transport Charges and Losses Types -->
    <!--
================================================================================
====== -->
    <!-- 2.7 Transport Abstract Types -->
    <xs:element name="baseTransportCharge" type="power:BaseTransportChargeType"
abstract="true" substitutionGroup="power:baseCharge">
        <xs:annotation>
                <xs:documentation>Abstract extension object for Emix Power Product
Charges</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:complexType name="ArrayOfTransportCharges">
        <xs:annotation>
                <xs:documentation>Collection of Emix Power Transport Product
Charges</xs:documentation>
        </xs:annotation>
        <xs:sequence>
                <xs:element ref="power:baseTransportCharge" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="BaseTransportChargeType" abstract="true" mixed="false">
        <xs:annotation>
                <xs:documentation>Type of Abstract extension object for  Emix
Transport Charges</xs:documentation>
        </xs:annotation>
        <xs:complexContent mixed="false">
                <xs:extension base="power:BaseChargeType"/>
        </xs:complexContent>
    </xs:complexType>
    <!--
================================================================================
====== -->
    <!-- 2.8 Congestion and Loss Charges -->
    <!-- 2.8.1 Congestion Revenue Rights Charge-->
    <xs:element name="congestionRevenueRights" type="power:CongestionRevenueRightsType"
substitutionGroup="power:baseTransportCharge"/>
    <xs:complexType name="CongestionRevenueRightsType" mixed="false">
        <xs:annotation>
                <xs:documentation>Financial Hedge for Congestion, a forward contract
for congestion revenues to potentially offset congestion charges. Also known as
Financial Transmission Rights or Congestion Revenue Rights</xs:documentation>
        </xs:annotation>
        <xs:complexContent mixed="false">
                <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                                <xs:element ref="power:transportInterface"/>
                                <xs:element ref="power:transportCongestionFee"/>
                        </xs:sequence>
                </xs:extension>
        </xs:complexContent>
```

Formatted: Font: 8 pt

```
   </xs:complexType>
   <!-- 2.8.2 Congestion Charge-->
   <xs:element name="congestionCharge" type="power:CongestionChargeType"
substitutionGroup="power:baseTransportCharge"/>
   <xs:complexType name="CongestionChargeType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Congestion Charge is the cost of purchasing the
right to transfer power over a given segment of the grid.</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                               <xs:element ref="power:transportInterface"/>
                               <xs:element ref="power:transportCongestionFee"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 2.8.3 Marginal Loss Charge -->
   <xs:element name="marginalLossCharge" type="power:MarginalLossChargeType"
substitutionGroup="power:baseTransportCharge"/>
   <xs:complexType name="MarginalLossChargeType" mixed="false">
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                               <xs:element ref="power:marginalLossFee"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 2.8.4 Marginal Loss -->
   <xs:element name="marginalLoss" type="power:MarginalLossType"
substitutionGroup="power:baseTransportCharge"/>
   <xs:complexType name="MarginalLossType" mixed="false">
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                               <xs:element ref="power:lossFactor"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 2.8.5 Conversion Loss -->
   <xs:element name="conversionLoss" type="power:ConversionLossType"
substitutionGroup="power:baseTransportCharge"/>
   <xs:complexType name="ConversionLossType" mixed="false">
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                               <xs:element ref="power:pnode"/>
                               <xs:element ref="power:lossFactor"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <xs:element name="transportAccessFee" type="power:TransportAccessFeeType"
substitutionGroup="power:baseTransportCharge"/>
   <xs:complexType name="TransportAccessFeeType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Transport Access Fee is a Fixed Charge (not
dependent on congestion or quantity) to access a transport system.</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="power:BaseTransportChargeType">
                        <xs:sequence>
                               <xs:element ref="power:transportInterface"/>
                               <xs:element ref="emix:price"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
```

Formatted: Font: 8 pt

```
   <!--
================================================================================================
====== -->
   <!-- 2.9 Elemental Charge and Loss Types -->
   <!--
================================================================================================
====== -->
   <!-- 2.9.3 Loss Fee  -->
   <xs:element name="marginalLossFee" type="power:MarginalLossFeeType"/>
   <xs:simpleType name="MarginalLossFeeType">
          <xs:annotation>
                 <xs:documentation>Marginal Loss Fee</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:decimal"/>
   </xs:simpleType>
   <!-- 2.9.4 Transport Congestion Fee -->
   <xs:element name="transportCongestionFee" type="power:TransportCongestionFeeType"/>
   <xs:simpleType name="TransportCongestionFeeType">
          <xs:annotation>
                 <xs:documentation>Financial Transmission Rights (FTR) regarding
transmission capacity.</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:decimal"/>
   </xs:simpleType>
   <!-- 2.9.5 Loss Factor -->
   <xs:element name="lossFactor" type="power:LossFactorType"/>
   <xs:simpleType name="LossFactorType">
          <xs:annotation>
                 <xs:documentation>Reduction in amount delivered as product travels.
(lossFactor * purchase amount) = delivered amount</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:float">
                 <xs:maxInclusive value="1"/>
          </xs:restriction>
   </xs:simpleType>
   <!-- 2.9.6 Enumeration & Simple Types for Products -->
   <xs:element name="productType" type="power:ProductTypeType"/>
   <xs:simpleType name="ProductTypeType">
          <xs:union memberTypes="power:ProductTypeEnumeratedType emix:EmixExtensionType
power:PowerOptionTypeType"/>
   </xs:simpleType>
   <xs:simpleType name="ProductTypeEnumeratedType">
          <xs:restriction base="xs:string">
                 <xs:enumeration value="Energy"/>
                 <xs:enumeration value="Transport"/>
                 <xs:enumeration value="EnergyOption"/>
                 <xs:enumeration value="TransportOption"/>
                 <xs:enumeration value="FullRequirementsPower"/>
                 <xs:enumeration value="FullRequirementsPowerWithDemandCharge"/>
                 <xs:enumeration value="FullRequirementsPowerWithMaximumAndMinimum"/>
                 <xs:enumeration value="HourlyDayAhead"/>
                 <xs:enumeration value="Ex-AnteRealTimePrice"/>
                 <xs:enumeration value="TimeOfUsePricing"/>
                 <xs:enumeration value="Transport"/>
                 <xs:enumeration value="CongestionRevenueRights"/>
          </xs:restriction>
   </xs:simpleType>
</xs:schema>
```

F.3 Resource.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2011 rel. 2 (x64) (http://www.altova.com) by Toby Considine
(TC9, Inc) -->
<!-- resource.xsd - Resource Descriptions for OASIS EMIX 1.0 WD23 (20110411)

Set includes:
    EMIX, EMIX-Requirements, EMIX-Warrants (emix)
    Power, Power-Contracts, Power-Quality (power)
    Resource (resource)
```

Formatted: Font: 8 pt

```
This set built on the WS-Calendar v1.0 PRD02 Schemas.
-->
<xs:schema xmlns:resource="http://docs.oasis-open.org/ns/emix/power/resource"
xmlns:power="http://docs.oasis-open.org/ns/emix/power" 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/power/resource"
elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:import namespace="http://docs.oasis-open.org/ns/emix" schemaLocation="emix.xsd"/>
   <xs:import namespace="http://docs.oasis-open.org/ns/emix/power"
schemaLocation="power.xsd"/>
   <!-- 3.0 Resource are described in terms of their capabilities Capabilities to aid in
the matching of need and supplier -->
   <xs:element name="loadReduction" type="resource:LoadReductionType"
substitutionGroup="emix:productDescription"/>
   <xs:element name="generation" type="resource:GenerationType"
substitutionGroup="emix:productDescription"/>
   <xs:element name="activeReserve" type="resource:ActiveReserveType"
substitutionGroup="emix:productDescription"/>
   <xs:element name="regulationService" type="resource:RegulationServiceType"
substitutionGroup="emix:productDescription"/>
   <xs:element name="productVoltageRegulation"
type="resource:ProductVoltageRegulationType"
substitutionGroup="emix:productDescription"/>
   <!-- 3.1 Load resource -->
   <xs:complexType name="LoadReductionType">
          <xs:annotation>
                 <xs:documentation>A Load Reduction Resource ramps down, stays down,
and then ramps up. For stagingRamps, endRamp is less than beginRamp. For recoveryRamps,
endRamp is greater than beginRamp.</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="resource:PowerResponseType"/>
          </xs:complexContent>
   </xs:complexType>
   <!-- 3.2 Generation Resource -->
   <xs:complexType name="GenerationType">
          <xs:annotation>
                 <xs:documentation>A Generation Resource ramps up, stays up, and then
ramps down. For stagingRamps, endRamp is greater than beginRamp. For recoveryRamps,
endRamp is less than beginRamp.</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="resource:PowerResponseType">
                        <xs:sequence>
                               <xs:element name="Type"
type="resource:ResourceTypeType" minOccurs="0" maxOccurs="1"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 3.5 Active Reserve -->
   <xs:complexType name="ActiveReserveType">
          <xs:annotation>
                 <xs:documentation>Active Reserve</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="resource:ResourceDescriptionType">
                        <xs:sequence>
                               <xs:element ref="power:powerOptionType"/>
                               <xs:element ref="resource:targetRegulation"/>
                               <xs:element ref="resource:dispatchTime"/>
                               <xs:element ref="emix:autonomous" minOccurs="0">
                                      <xs:annotation>
                                             <xs:documentation>Resource provides
autonomous management of its local circuits. If true, service notes local conditions and
dispatches itself. If false, it waits for dispatch request from VTN.</xs:documentation>
                                      </xs:annotation>
                               </xs:element>
```


Formatted: Font: 8 pt

```
                                <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>
```


Formatted: Font: 8 pt

```
   <!-- 3.9.1 Resource Types -->
   <xs:element name="resourceType" type="resource:ResourceTypeType"/>
   <xs:simpleType name="ResourceTypeType">
          <xs:union memberTypes="resource:ResourceTypeEnumeratedType
emix:EmixExtensionType"/>
   </xs:simpleType>
   <xs:simpleType name="ResourceTypeEnumeratedType">
          <xs:annotation>
                 <xs:documentation>Resource types share common responsiveness and
predictability characteristics, sometimes covarying across resources in the same class.
(Example: Solar in the same region failing at the same time)</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:token">
                 <xs:enumeration value="DispatchableHydro"/>
                 <xs:enumeration value="NonDispatchableHydro"/>
                 <xs:enumeration value="WindGeneration"/>
                 <xs:enumeration value="SolarGeneration"/>
                 <xs:enumeration value="TollingContract"/>
                 <xs:enumeration value="AggregateResource"/>
                 <xs:enumeration value="DispatchableStorage"/>
          </xs:restriction>
   </xs:simpleType>
   <!-- 3.9.2 Regulation Products -->
   <xs:element name="productTypeRegulation" type="resource:ProductTypeRegulationType"/>
   <xs:simpleType name="ProductTypeRegulationType">
          <xs:annotation>
                 <xs:documentation>enumerates the Voltage Regulation
Products</xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:string">
                 <xs:enumeration value="RegulationUp"/>
                 <xs:enumeration value="RegulationDn"/>
                 <xs:enumeration value="RegulationUp-Dn"/>
          </xs:restriction>
   </xs:simpleType>
   <!-- 4.0 Resource Semantics -->
   <!-- 4.1 Resource Capability -->
   <xs:element name="powerResponse" type="resource:PowerResponseType"/>
   <xs:complexType name="PowerResponseType" abstract="true">
          <xs:annotation>
                 <xs:documentation>Generic model describing the power response
capabilities of a resource</xs:documentation>
          </xs:annotation>
          <xs:complexContent>
                 <xs:extension base="resource:ResourceDescriptionType">
                        <xs:sequence>
                               <xs:element ref="resource:stagingRamp" minOccurs="0"
maxOccurs="1"/>
                               <xs:element ref="resource:maximumResponse"
minOccurs="0" maxOccurs="1"/>
                               <xs:element ref="resource:minimumResponse"
maxOccurs="1"/>
                               <xs:element ref="resource:recoveryRamp" minOccurs="0"
maxOccurs="1"/>
                               <xs:element ref="resource:offerCurve" minOccurs="0"
maxOccurs="1"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 4.1 Ramp Rates -->
   <!-- 4.1.3 Power Ramp Rate -->
   <xs:element name="stagingRamp" type="resource:ArrayOfRampSegments"/>
   <xs:element name="recoveryRamp" type="resource:ArrayOfRampSegments"/>
   <xs:element name="powerRamp" type="resource:ArrayOfRampSegments"/>
   <xs:complexType name="PowerRampType">
          <xs:annotation>
                 <xs:documentation>A Power Ramp is an Array of of Ramp Segments that
describing a Resource's ability to change level. A Power Ramp is either monotonically
increasing or monotonically decreasing.</xs:documentation>
          </xs:annotation>
```

Formatted: Font: 8 pt

```
            <xs:sequence>
                  <xs:element ref="resource:rampSegments"/>
            </xs:sequence>
      </xs:complexType>
      <xs:element name="rampSegments" type="resource:ArrayOfRampSegments"/>
      <xs:complexType name="ArrayOfRampSegments">
            <xs:annotation>
                  <xs:documentation>Collection of Power Ramp Segments</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element ref="resource:powerRampSegment" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:element name="powerRampSegment" type="resource:PowerRampSegmentType"/>
      <xs:complexType name="PowerRampSegmentType">
            <xs:annotation>
                  <xs:documentation>A Power Ramp Segment describes a change up or down
in units/duration. A ramp rate holds for the duration between beginRamp to
endRamp</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="rate" type="power:PowerQuantityType"/>
                  <xs:element ref="emix:duration"/>
                  <xs:element ref="resource:beginRamp"/>
                  <xs:element ref="resource:endRamp"/>
                  <xs:element ref="emix:integralOnly"/>
            </xs:sequence>
      </xs:complexType>
      <xs:element name="beginRamp" type="xs:int"/>
      <xs:element name="endRamp" type="xs:int"/>
      <!-- 4.1.4 Power Ramp Rate -->
      <xs:element name="percentRampRate" type="resource:PercentRampRateType"/>
      <xs:complexType name="PercentRampRateType">
            <xs:annotation>
                  <xs:documentation>Change up or down in percent of total
response.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element ref="resource:rate"/>
                  <xs:element ref="emix:duration"/>
            </xs:sequence>
      </xs:complexType>
      <!-- 4.2 Constraints and Requirements unique to Power Resources-->
      <xs:element name="minimumLoad" type="resource:MinimumLoadType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>Constraint on Minimum Load that a Resource can
maintain</xs:documentation>
            </xs:annotation>
      </xs:element>
      <xs:element name="maximumPower" type="resource:MaximumPowerType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>Constraint on Maximum Power available from a
resource</xs:documentation>
            </xs:annotation>
      </xs:element>
      <xs:element name="maximumEnergy" type="resource:MaximumEnergyType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>Constraint on Maximum Energy available from a
resource</xs:documentation>
            </xs:annotation>
      </xs:element>
      <xs:element name="minimumLoadReduction" type="resource:MinimumLoadReductionType"
substitutionGroup="emix:baseConstraint">
            <xs:annotation>
                  <xs:documentation>Constraint on Minimum Load Reduction resource can
make</xs:documentation>
            </xs:annotation>
```

Formatted: Font: 8 pt

```
   </xs:element>
   <xs:complexType name="MinimumLoadType" mixed="false">
          <xs:annotation>
                 <xs:documentation>ype of Constraint on Minimum Load that a Resource
can maintain</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseConstraintType">
                        <xs:sequence>
                               <xs:element ref="power:powerQuantity"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="MaximumPowerType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of Constraint on Maximum Power available from a
resource</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseConstraintType">
                        <xs:sequence>
                               <xs:element ref="power:powerQuantity" minOccurs="1"
maxOccurs="1"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="MaximumEnergyType" mixed="false">
          <xs:annotation>
                 <xs:documentation> Type of Constraint on Maximum Energy available from
a resource</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseConstraintType">
                        <xs:sequence>
                               <xs:element ref="power:energyQuantity"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 4.2.5 Minimum Load Reduction -->
   <xs:complexType name="MinimumLoadReductionType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Minimum units for a load reduction (e.g., MW rating
of a discrete pump)</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseConstraintType">
                        <xs:sequence>
                               <xs:element ref="power:powerQuantity" minOccurs="1"
maxOccurs="1"/>
                        </xs:sequence>
                 </xs:extension>
          </xs:complexContent>
   </xs:complexType>
   <!-- 4.3.1 Offer Segment elements  -->
   <xs:element name="offerCurve" type="resource:OfferCurveType"
substitutionGroup="emix:baseRequirement"/>
   <xs:complexType name="OfferCurveType" mixed="false">
          <xs:annotation>
                 <xs:documentation>Type of a collection of Offer Segments used to
compute cost requirements across a range of power.</xs:documentation>
          </xs:annotation>
          <xs:complexContent mixed="false">
                 <xs:extension base="emix:BaseRequirementType">
                        <xs:sequence>
                               <xs:element name="offerSegment"
type="resource:OfferSegmentType" maxOccurs="unbounded"/>
                        </xs:sequence>
                 </xs:extension>
```

Formatted: Font: 8 pt

```
            </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>
```

Formatted: Font: 8 pt

```
   </xs:complexType>
   <!-- 4.9 Miscelenous Semantic elementsvolt-var -->
   <xs:element name="dispatchTime" type="emix:DurationType"/>
   <xs:element name="maximumDeliveryRate" type="emix:QuantityType"/>
   <xs:element name="minimumDeliveryRate" type="emix:QuantityType"/>
   <xs:element name="maximumResponse" type="emix:QuantityType"/>
   <xs:element name="minimumResponse" type="emix:QuantityType"/>
   <xs:element name="rate" type="emix:QuantityType"/>
   <xs:element name="targetRegulation" type="power:PowerAttributesType"/>
   <xs:element name="dispatchUp" type="emix:DurationType">
         <xs:annotation>
               <xs:documentation>Time in which resource can respond to a request to
increase energy provided. If zero, no dispatchUp available. Can also be startup delay
for non-spinning reserve.</xs:documentation>
         </xs:annotation>
   </xs:element>
   <xs:element name="dispatchDown" type="emix:DurationType">
         <xs:annotation>
               <xs:documentation>Time in which resource can respond to a request to
decrease energy provided. If zero, no dispatch Down available.</xs:documentation>
         </xs:annotation>
   </xs:element>
   <xs:element name="rampTime" type="emix:DurationType">
         <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 name="window" type="emix:DurationType">
         <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:schema>
```


Formatted: Font: 8 pt

G. An Example

24 Hours of pricing on a full requirements contract.

```
<?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>
```

Formatted: Font: 8 pt

```
                            <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>
```

Formatted: Font: 8 pt

```
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>3fbb3ccb-a38e-43b2-968b-
0117c57a1d24@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>78839070-98bc-42c4-b216-
3665efdeaef4@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>-
0.13</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>6a5d23b2-0aa9-4309-b2d3-
b04f630add99@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>3fbb3ccb-a38e-43b2-968b-
0117c57a1d24@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>0.15</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
```


Formatted: Font: 8 pt

```
                    <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>
```

Formatted: Font: 8 pt

```
                    <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>
```

Formatted: Font: 8 pt

```
<emix:priceEnumeration>1.12</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>2097d7d8-f554-469c-8c92-
9eb8551c086d@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>a843d8d0-28f8-4a31-a7ee-
25b14c701036@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>1.14</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>f2f630f6-f673-47ae-8a89-
3d2b23f379c6@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>2097d7d8-f554-469c-8c92-
9eb8551c086d@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>1.15</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
```

Formatted: Font: 8 pt

```
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>1d9ad168-433e-4504-97f5-
05a8df1d1f20@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>f2f630f6-f673-47ae-8a89-
3d2b23f379c6@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>2.74</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>f65aa213-3176-47e3-88c3-
0f7dbe0ec99e@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>1d9ad168-433e-4504-97f5-
05a8df1d1f20@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>1.25</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>920c3e26-cea2-4456-84a7-
9f5f369cf491@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
```

Formatted: Font: 8 pt

```
                    </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>
```

Formatted: Font: 8 pt

```
<emix:priceEnumeration>1.31</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>01063041-4d93-46a6-9fd6-
7b4f92a938e0@examples.oasis-open.org</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>2482a65c-bd0c-4a33-b777-
cf89d637d1b8@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>1.00</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
                <xcal:properties>
                    <xcal:uid>
                        <xcal:text>2bb61d4e-a3ea-423a-98ef-
395e25b35674@examples.oasis-open.org</xcal:text>
                    </xcal:uid>
                    <xcal:related-to>
                        <xcal:parameters>
                            <xcal:reltype>
                            </xcal:reltype>
                        </xcal:parameters>
                        <xcal:uid>01063041-4d93-46a6-9fd6-
7b4f92a938e0@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>0.99</emix:priceEnumeration>
                                </emix:priceAbsolute>
                            </power:unitEnergyPrice>
                        </emix:productDescription>
                    </xcal:x-wsCalendar-attach>
                </xcal:properties>
            </xcal:interval>
            <xcal:interval>
```

Formatted: Font: 8 pt

```
            <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>
```

Formatted: Font: 8 pt

```
                    </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>
```

Formatted: Font: 8 pt

```
<emix:priceEnumeration>0.87</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
        <xcal:interval>
            <xcal:properties>
                <xcal:uid>
                    <xcal:text>9e89696d-6511-427b-9086-
b595ef25f23c@examples.oasis-open.org</xcal:text>
                </xcal:uid>
                <xcal:related-to>
                    <xcal:parameters>
                        <xcal:reltype>
                        </xcal:reltype>
                    </xcal:parameters>
                    <xcal:uid>3d0b45e4-382b-4964-8779-
1b7b8c3d7133@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>0.76</emix:priceEnumeration>
                            </emix:priceAbsolute>
                        </power:unitEnergyPrice>
                    </emix:productDescription>
                </xcal:x-wsCalendar-attach>
            </xcal:properties>
        </xcal:interval>
    </xcal:components>
    <emix:currency>USD</emix:currency>
    <emix:marketContext>market.context@examples.com</emix:marketContext>
    <emix:transactiveState>Tender</emix:transactiveState>
</emix:product>
```


Formatted: Font: 8 pt

~~D.~~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

Formatted: Font: 8 pt

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 ~~Terms.~~Base. General tightening. Addition of Load and Power Offers, including 3-part bids for each.
~~CD01~~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

4871

Formatted: Font: 8 pt