Partial Response to:

Public Review of Energy Interoperation Version 1.0
Comments on EMIX Product Specification Standard
Paul Centolella
Commissioner, Public Utilities Commission of Ohio
SGIP Governing Board Member

By: Ed Cazalet, Co-Chair OASIS EMIX TC

First, thank you very much to Commissioner Paul Centolella and the PUCO for their excellent comments which address issues for both to EMX and Energy Interop.

Summary of Paul's Comments
Paul highlights significant changes in state and federal electricity rates. "These changes reflect a continuing trend towards better reflecting the value of electricity in prices, including dynamic prices.

Paul's comments are focused on "examples of emerging product and rate structures that the EMIX should accommodate".

"The purpose is to provide examples of emerging and innovative products and rate structures so that developers can work through how such products and rates might be expressed within the EMIX framework, determine what needs to be represented and to identify any gaps or ambiguities".

Paul's Recommended Examples:
Paul suggests the examples indicated in the diagram below.
The examples are described in Paul's comments. We will draw on the detailed descriptions of these examples in expressing these examples using EMIX/EI.

Paul also attached a number of supporting documents that provide valuable details for preparing our examples: The attachments are listed below:

A. Price Responsive Demand Integration into PJM Capacity and Energy Markets by PJM.
B. PJM Open Access Transmission Tariff: Changes for Price Response Demand "Substation".
C. PJM Proposed Market Rule Revisions Provisions : Integration of Price Responsive Demand in PJM Markets
D. Paper by Centolella and Ott (SVP Markets, PJM) on Integration of Price Responsive Demand into PJM Wholesale Markets
E. Tariff Changes on Price Responsive Demand (source not indicated)
F. Removing Barriers to Price Responsive Demand, Paul Centolella
G. PJM Compliance Filing on Scarcity Pricing
H. PJM Market Monitor Protest of PJM Filing
I. Comments of PUCO to FERC on PJM's Scarcity Pricing Proposal
J. Duke Peak Time Rebate Pilot Program
Introduction to Proposed EMIX/EI Examples in Response to Paul's Suggestions

In general, retail tariff prices will be communicated by the party directly serving a customer, typically called a load serving entity (LSE) or retail energy provider (REP). We will use the term REP for both. A REP may offer either cost of service prices or market prices; however the communication of the prices would be the same using EMIX/EI. We will address all of the retail examples in the above figure.

Paul also describes two aspects of the wholesale interface of REPs to the PJM markets. One aspect is price responsive demand (RPD) forecasting that would reduce the amount of forward capacity a REP would procure. This issues not about communicating prices, which we address below. It is about load forecasting that will take account of the impact of PRD. Since EMIX and EI will apply only to the communication of load forecasts and is not concerned with how a forecast is computed. However, an example showing communication of forecasted energy requirement to an ISO/RTO could be provided.

The second wholesale interface issue is shortage pricing. Shortage pricing is where in shortage conditions the ISO will increase the price in its market to better reflect shortage conditions. Despite the clear merits of this proposal, the method calculation of prices is out of scope for EMIX and the communication of prices by the ISO will be similar regardless of where prices are increased for shortages. Thus a wholesale example that illustrates how to communicate day-ahead and real-time energy prices and other charges to the REPs will likely be all we need at this point.

Retail Real-Time Tariff

There are several possible real-time tariffs. As examples, we could start with two interesting cases.

Example 1: A full-requirements service with hourly prices for 24 hours of the day, published the day before.
Example 2: A full-requirements service with a 5-minute price published shortly after each 5 minute interval.
EMIX-v1.0 : Table 6-4 (line 371) below describes the parameters of a requirements product:

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract Type</td>
<td>Enumerated String</td>
<td></td>
</tr>
<tr>
<td>Power Units</td>
<td>Power Units</td>
<td>As defined below</td>
</tr>
<tr>
<td>Service Location</td>
<td>Service Location</td>
<td>If response is for a single location, should be in gluon to apply to the entire sequence and be omitted in the intervals</td>
</tr>
<tr>
<td>Power Attributes</td>
<td>Power Attributes</td>
<td>As defined below</td>
</tr>
<tr>
<td>Price</td>
<td>From EMIX</td>
<td>Price per Unit during the Interval.</td>
</tr>
<tr>
<td>Demand Charge</td>
<td>Demand Charge.</td>
<td>Optional. See below. There may be multiple demand charges.</td>
</tr>
<tr>
<td>Maximum Power</td>
<td>Power</td>
<td>Buyer may not consume at more than this rate</td>
</tr>
<tr>
<td>Minimum Power</td>
<td>Power</td>
<td>If buyer consumes than this rate, the buyer is assessed a charge to bring it up to this rate.</td>
</tr>
<tr>
<td>Duration</td>
<td>From WS-Calendar</td>
<td>May be nil if all intervals have duration specified</td>
</tr>
<tr>
<td>Performance</td>
<td>From WS-Calendar</td>
<td>Indicates performance requirements such as fixed run-time, absolute end time, etc.</td>
</tr>
</tbody>
</table>

The demand charge, maximum power, minimum power and performance will not apply to our two example products.

From the EMIX.xsd we get

```xml
  <xs:element name="emix" type="emix:type-emix">
    <xs:annotation>
      <xs:documentation>Base element for Energy Market Information Exchange (EMIX)</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="type-emix">
    <xs:sequence>
      <xs:element name="uid" type="xs:string"/>
      <xs:element name="createdDateTime" type="xs:dateTime"/>
      <xs:element name="transactiveState" type="emix:type-transactiveState"/>
    </xs:sequence>
  </xs:complexType>
```
The createdDateTime will be at say 2 pm the day before for Example 1 or 30 seconds after each 5 min interval for Example 2. The transactiveState will be "Tender" which is an offer. When the actual usage is available the transactive state will be "Transaction". PackageDiscount, and envelopeContents can be nil. Currency is USD. marketContext is an URL to "yourREP".

From EMIX.xsd we also get the schema for EMIX terms as follows:

<xs:element name="terms" type="emix:type-emixTerms" maxOccurs="unbounded" />  
<xs:element name="extendedPrice" type="emix:type-price" nillable="true" />  
<xs:element name="packageDiscount" type="emix:type-price" nillable="true" />  
<xs:element name="currency" type="xs:string" />  
<xs:element name="marketContext" type="emix:type-marketContext" />  
<xs:element name="envelopeContents" type="emix:type-envelope" nillable="true" maxOccurs="1" />  
</xs:sequence>  
</xs:complexType>

<xs:complexType name="type-gluelon">
  <xs:annotation>
    <xs:documentation>This class is a reference to the WS-Calendar Gluon interface. Note that the artifact, shared by Interval and Gluon, has already been extended to include the EMIX Product Description.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element ref="wscal:x-calendargluon" />  
  </xs:sequence>  
</xs:complexType>

<xs:complexType name="type-interval">
  <xs:annotation>
    <xs:documentation>9.2 Interval - a duration, contract info, and maybe a starting time</xs:documentation>
  </xs:annotation>
</xs:complexType>

<xs:complexType name="type-emixTerms">
  <xs:annotation>
    <xs:documentation>9.3 Interval - a duration, contract info, and maybe a starting time</xs:documentation>
  </xs:annotation>
</xs:complexType>

<xs:complexType name="duration">
  <xs:annotation>
    <xs:documentation>9.4 Duration from WS-Calendar</xs:documentation>
  </xs:annotation>
</xs:complexType>
This class is a reference to the WS-Calendar Duration class.

9.5 Abstract Product Description

In EMIX, the WS-Calendar Artifact appears as a product description.

From PowerContract.xsd we get

Often used in retail residential rates. Simple prices, will supply all used. Demand Charges Optional.

For Example 1, we need a sequence of 24 intervals with the same one hour duration with a starting time of midnight and with different prices in each interval needed. As a Tender the price will be the same to a set of ServiceLocations. As a
Transaction based on actual usage, a different quantity also will be needed in each interval.

For Example 2, the structure is similar except a new Tender is created every 5 minutes. Usage Measurements creating a transaction may arrive every 5 minutes or once a day, for example.

I will leave it to Bill and/or Toby to apply the product descriptions to the emix-terms (sequences) to take these examples to Schemas.

**Two-Part Forward and Real-time Retail Real-Time Tariff**

Example 3: Forward Peak-Off-Peak Contract with Hourly Real Time Pricing
The REP offers one year tariff for a fixed rate of delivery of power during specified on-peak hours (week-days except certain holidays) from 8 am to 10 pm and a different rate of delivery for off-peak hours, weekends and holidays. The on-and-off peak prices are different.

The customer has the option of forward purchasing any amount of on and off-peak power, but the assumed delivery rate is constant over all on-peak hours and over the off-peak hours at a different rate. The forward purchase is an obligation that the customer pays for regardless of actual consumption.

If actual consumption in any hour is greater than the forward purchase then the customer will pay the hourly real time price for the difference. If the actual consumption is less, then the customer will get a credit at the real-time price for the difference.

The tender and transaction for the forward contract are modeled with a constant power product in EMIX. A constant power product is modeled in EMIX as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Constant Power</strong></td>
<td><strong>Quantity</strong></td>
<td>Defines Constant Power Intervals. Does not coexist with Starting and Final Power Quantities</td>
</tr>
<tr>
<td>Name</td>
<td>Definition</td>
<td>Note</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Starting Power Quantity</td>
<td>EMIX.Quantity</td>
<td>Defines Ramped Power Intervals. Requires matching Final Power Quantity. Does not coexist with Constant Quantity</td>
</tr>
<tr>
<td>Final Power Quantity</td>
<td>EMIX.Quantity</td>
<td>Defines Ramped Power Intervals. Requires matching Starting Power Quantity. Does not coexist with Constant Quantity</td>
</tr>
<tr>
<td>Power Units</td>
<td>Power Units</td>
<td>As defined below</td>
</tr>
<tr>
<td>Service Location</td>
<td>Service Location</td>
<td>Should normally be only in the Gluon and omitted from the intervals. If the Product is an aggregated response across multiple locations, one per interval, then it MAY appear in the interval instead</td>
</tr>
<tr>
<td>Power Attributes</td>
<td>Power Attributes</td>
<td>As defined below</td>
</tr>
<tr>
<td>UnitPrice</td>
<td>EMIX.Price</td>
<td>Price per Unit Quantity. Includes currency</td>
</tr>
<tr>
<td>Price</td>
<td>EMIX.Price</td>
<td>Extended price for interval. Includes quantity and currency</td>
</tr>
<tr>
<td>Duration</td>
<td>From WS-Calendar</td>
<td>May be nil if inherited from Gluon</td>
</tr>
<tr>
<td>Performance</td>
<td>From WS-Calendar</td>
<td>Indicates performance requirements such as fixed run-time, absolute end time, etc.</td>
</tr>
</tbody>
</table>

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

Table 0-2: Power Gluon Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Quantity</td>
<td>EMIX.Quantity</td>
<td>Defines Constant Power Intervals. Does not coexist with Starting and Final Power Quantities</td>
</tr>
<tr>
<td>Power Units</td>
<td>Power Units, enum</td>
<td>As defined below</td>
</tr>
<tr>
<td>Service Location</td>
<td>Service Location</td>
<td>If response is for a single location, should be in gluon to apply to the entire sequence and be omitted in the intervals</td>
</tr>
<tr>
<td>Power Attributes</td>
<td>Power Attributes</td>
<td>As defined below</td>
</tr>
<tr>
<td>Unit Price</td>
<td>EMIX.Price</td>
<td>Price per Unit Quantity. Includes currency</td>
</tr>
<tr>
<td>Name</td>
<td>Definition</td>
<td>Note</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>Price</td>
<td>EMIX.Price</td>
<td>Extended price for interval. Includes quantity and currency</td>
</tr>
<tr>
<td>Duration</td>
<td>From WS-Calendar</td>
<td>May be nil if all intervals have duration specified</td>
</tr>
<tr>
<td>Performance</td>
<td>From WS-Calendar</td>
<td>Indicates performance requirements such as fixed run-time, absolute end time, etc.</td>
</tr>
</tbody>
</table>

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

The real-time pricing schema for this two-party tariff will be the same as the first examples with the exception that the quantity in the real-time transaction will be the metered quantity less the forward quantity for the hour.

At this point I leave it to Bill and Toby to complete the example.

I will describe the other retail pricing examples in a second version of this memo.