

# DEMS Training for UNB

# DEMS Training for UNB

## Table of contents

**SIEMENS**



- DEMS Background & Dev History
- DEMS Product Overview
- DEMS Structural Overview
- OpenADR Overview
- OpenADR Reference Slides
- Practical demonstration
  - **Setting up a VEN**
  - **Dispatching a DR event**



# DEMS Background and Development History



## DEMS Development History

**DEMS, in its current form (ver 4.X), is the successor (and fusion) of previous products:**

**DRMS = Distributed Resource Management System**

and

**DEMS = Decentralized Energy Management System (ver 3.X and earlier)**

and

**EIP = Energy IP**

# DEMS Development History

## DRMS

- Siemens product built at Princeton Research Center
- Originally built for a small number of large assets
  - Demand Response wasn't always for the masses
  - Focused on very large loads

## DEMS

- Siemens VPP (Virtual Power Plant) product
- Built in Germany & Austria in 1999

## EIP

- Grid application platform for numerous utility-focused applications
- DR, MDM, CIS, Outage Management, Workforce Management, etc.
- Built in California by eMeter (a Meter Data Management company)
- Acquired by Siemens in 2011

# DEMS Development Timeline

- **1999-2012**

- Original development by Siemens Control System(Germany & Austria)

- **2011**

- Siemens acquires eMeter (Siemens MDM and EIP)
- DRMS 1.0 built (Princeton Research Center)

- **2012**

- DRMS 2.0 launched
- Siemens Fredericton office is setup
- Siemens decides that DRMS will be part of eMeter product (moved to EIP platform)

- **2013**

- Fredericton office builds DRMS 2.5
- Fredericton office starts working on DRMS 3.0 (Core product team)

- **2014**

- Fredericton office builds DRMS 2.6\* (Product Enhancement team)

\* *DRMS 2.6 still has many active users*

## DEMS Development Timeline

- **2015**

- Released DRMS 3.1 for customer use (DRMS on EIP)
- Fredericton office starts to extend DRMS 3.1 into DEMS functionality (Core Product team)

- **2016**

- DEMS 4.0 (on EIP) = DRMS + DEMS features(VPP but not all functionality)

- **2017**

- DRMS 2.12 (standalone)
- DRMS 4.1\*\*

*\*\* DRMS 4.1 is what UNB and NB Power have deployed, although NB Power does not use the VPP functionality*

# DEMS Development Timeline

- **Take-Aways**

- Several versions of DRMS / DEMS are actively deployed across the world
  - *DRMS (Stand-alone)*
  - *DEMS (on EIP)*
- Highly customized to each utility
- Numerous development branches
- UNB has been given the same version that NB Power currently has

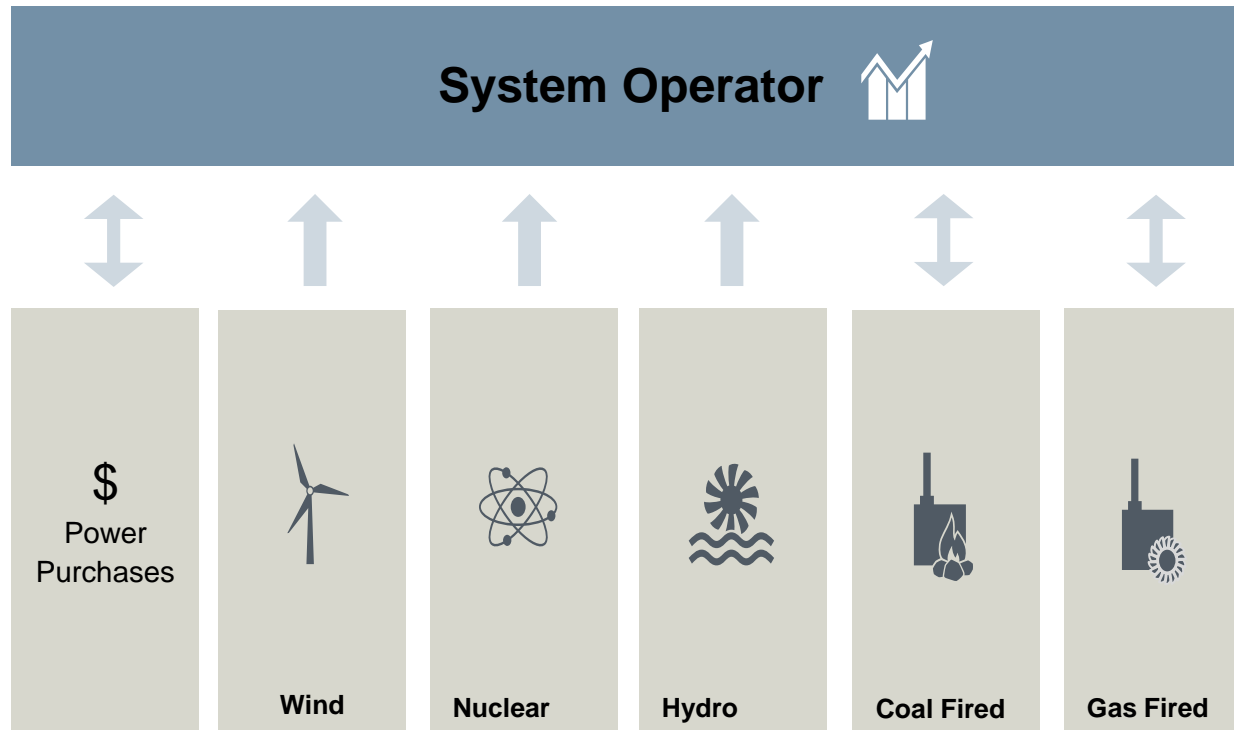


# DEMS Product Overview

**SIEMENS**

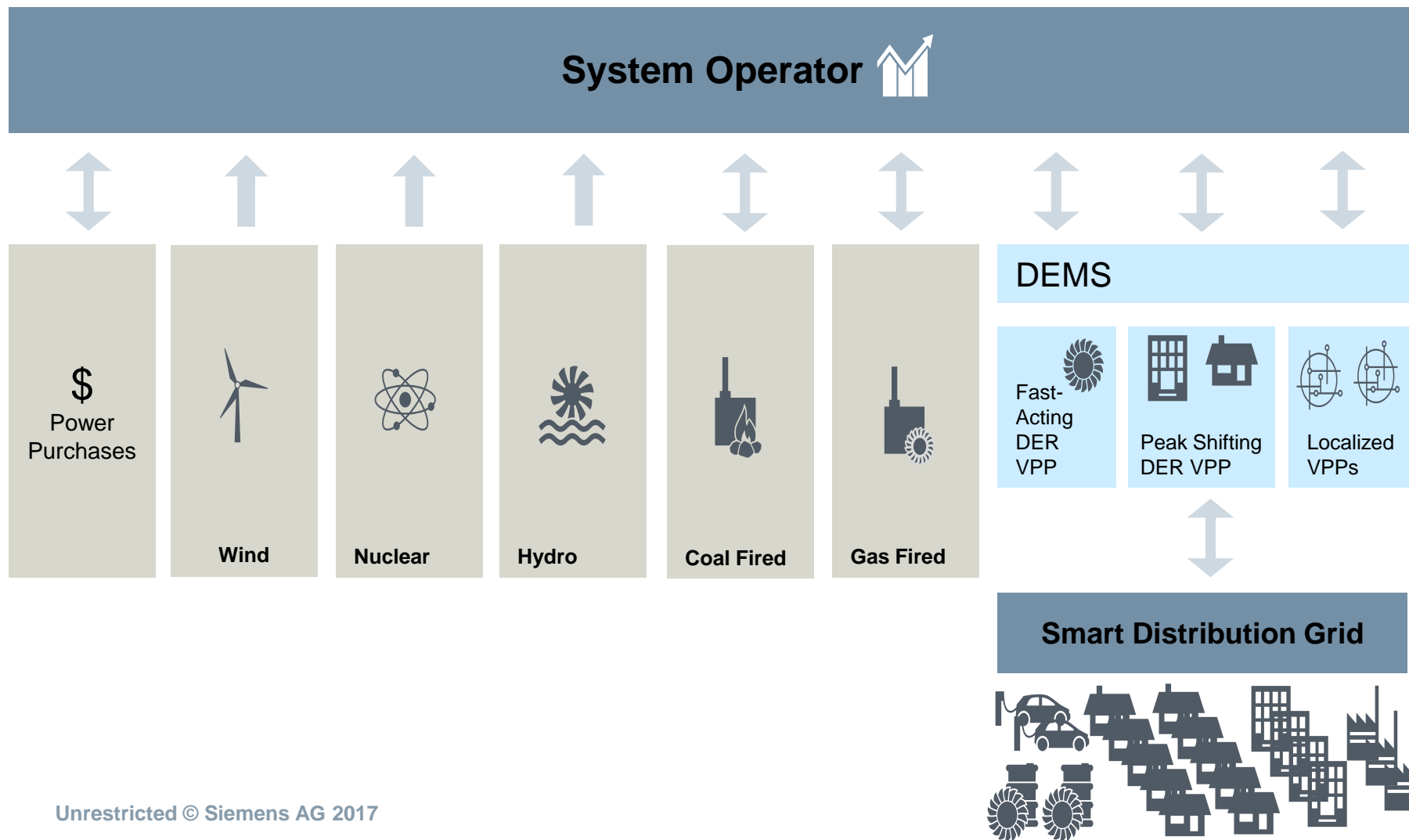


# Current System View



# Integrating DER

SIEMENS



# Users / Actors

## Large Investor Owned Utilities

- Have existing manual DR programs
- Looking to consolidate and manage all DR programs
- Looking to expand DR programs to include automation and E-car charging
- Desire to disintermediate aggregators between them and their customers

## Generation & Transmission Utilities

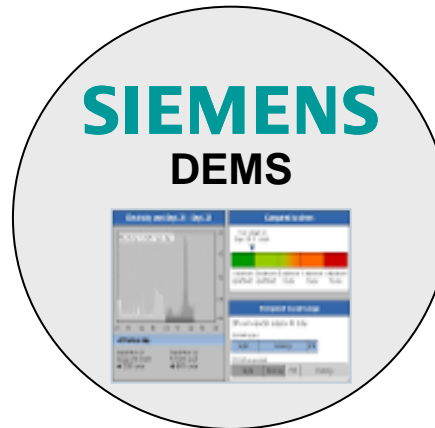
- Procure expensive power during peak demand to meet customer needs
- High benefits from leveraging DR as a virtual capacity resource within territory
- Seek integration between downstream metering & billing and energy management systems

## Local Distribution Utilities

- Offer a variety of demand-side management programs
- Looking to automate processes and take advantage of new wholesale market programs
- Desire to leverage AMI deployments to offer critical peak pricing and time of use rates

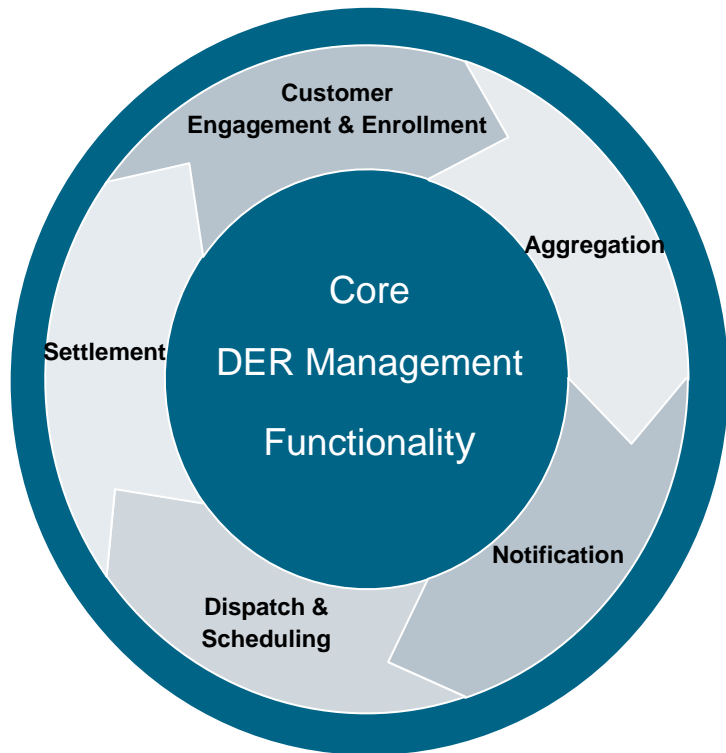
## Energy Retailers

- Actively procure capacity in energy markets, including DR markets
- Looking to take advantage of ancillary service and regulation markets
- Desire to expand breadth of services offer to customers and leverage DR services for customer satisfaction and retention



# Siemens DEMS: Harnessing the Value of Distributed Energy Resources

SIEMENS



## DER (Distributed Energy Resource)

Dispatchable and non-dispatchable resources on the grid, including Demand Response, Distributed Generation (solar, CHP, wind), electric vehicles, and storage technology. Energy efficiency and Micro Grids can also be considered to be DERs.

### Customer Engagement & Enrollment

- Integration with CIS or CRM Systems for managing customer enrollment
- Ability to interface with 3<sup>rd</sup> party enrollment tools and providers
- Ability to create and manage programs and constraints—for all customer classes, for both economic and reliability dispatch

### Aggregation

- Flexible ability to group consumer loads via a rules based engine
- Load groups can be based on feeder, substation, geography, customer class, etc.
- The heart of “surgical” DR

### Forecasting & Analytics

- Service point level load and DG forecasts
- Load shed and flexibility prediction based upon asset performance
- Configurable baseline calculations for settlement

### Dispatching & Scheduling

- Traditional Event Execution Model
- Resources can be dispatched from external systems (EMS/DMS, ISO, etc.)
- Event Wizard utilizing forecasting algorithms and analytics to provide load reduction recommendations
- Online control to track to schedule

### Notification

- Voice, Text, and Email based notifications
- Configurable messaging templates
- Notification for event scheduled, start, stop, cancelled, and extended

### Settlement

- Integration with MDM to automate customer billing and settlement
- Settlements performed in minutes when VEE'd data is available
- Baseline editing with audit trail



# DEMS “Big Picture”

- **Utilities are changing**

- Grid models are no longer just Top-Down
- Not just Generation following Load

- **Emerging Use-Case**

- How can utilities best tap into the resources of their customers
- Some of these resources are generating power
- How to make-up for lost revenue? (From customer generation or reduced consumption)
- How to rebalance the grid?

- **Current Methodology**

- NB Power baseline = 2000MW (approx)
- NB Power peak = 3500 MW (approx)
  - Peaking plants (e.g., fossil fuel) have fast ramp-up & ramp-down
  - Peaking plants are also very expensive to operate

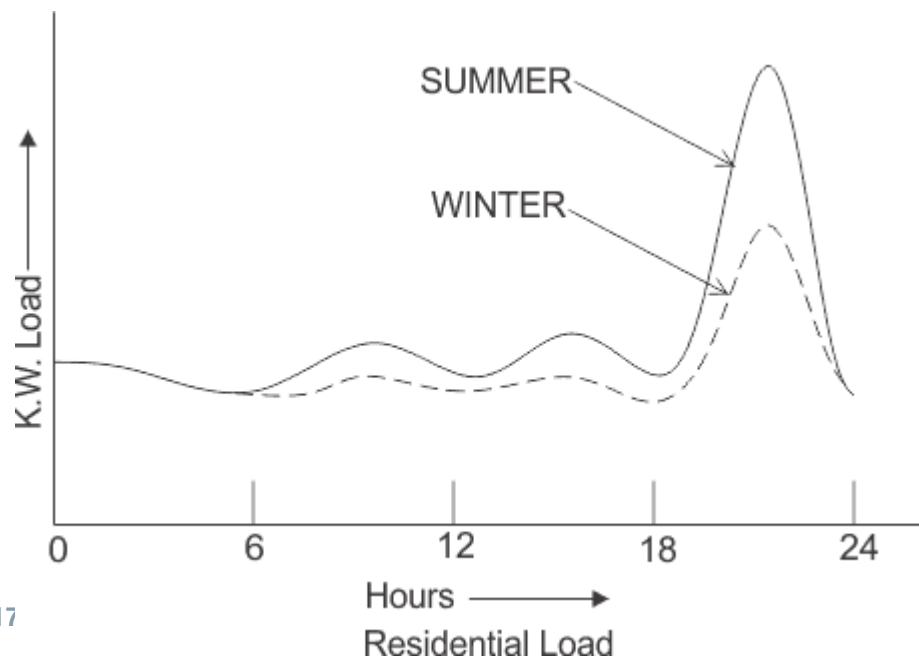
# DEMS “Big Picture”

## •Traditional Use-Case for DEMS

- Enable utilities to curtail peak consumption by shedding load

## •Business Case for NB Power

- They’ve already planned for peaking plants
  - Purchased fuel, allocated operational costs, etc.
- They don’t want to just curtail load, because that’s lost revenue
- They want to flatten the demand curve, by shifting load to off-peak times



# DEMS “Big Picture”

- **DEMS can load shed and shift**

- Load shedding must be assigned to a customer asset
- Customer (Account) information is critical
  - FlexSync = Siemens product to sync to CIS (Customer Information System). Used for bringing large amounts of assets into DEMS
- Data Model
  - Need to model a customer
  - Need to model a DER (Distributed Energy Resource)

# DEMS Structural Overview



# DEMS Structural Overview

- **Customer Model**

- Need to allocate Load Shed to a customer device
- So we need a data model for the customer (and their devices)
- Elements of the customer model
  - Account
  - Premise
  - Service Points (svcpts)
  - Consumer
  - Contacts
- Other elements
  - Programs
  - Enrollments
  - Constraints
  - Notifications



# DEMS Structural Overview

- **Service Point (SvcPt)**

- A logical representation of the point of entry for the utility's service
- i.e., the socket that a Smart Meter gets plugged into
- Could be a gateway, could be a meter
- Often representative of a device

- **Premise**

- Representative of a physical location (building) where svcpts are located

- **Program**

- A program that the utility wants to execute.
- Examples:
  - Summer Saver Program (targets Air conditioners, or pool pumps, etc)
  - Winter Peaking Program (targets thermostats and heating loads)
- Can only be created manually in DEMS

# DEMS Structural Overview

- **Constraints**

- Time of day / week
- Max duration
- Max number of DR dispatches
- Etc.

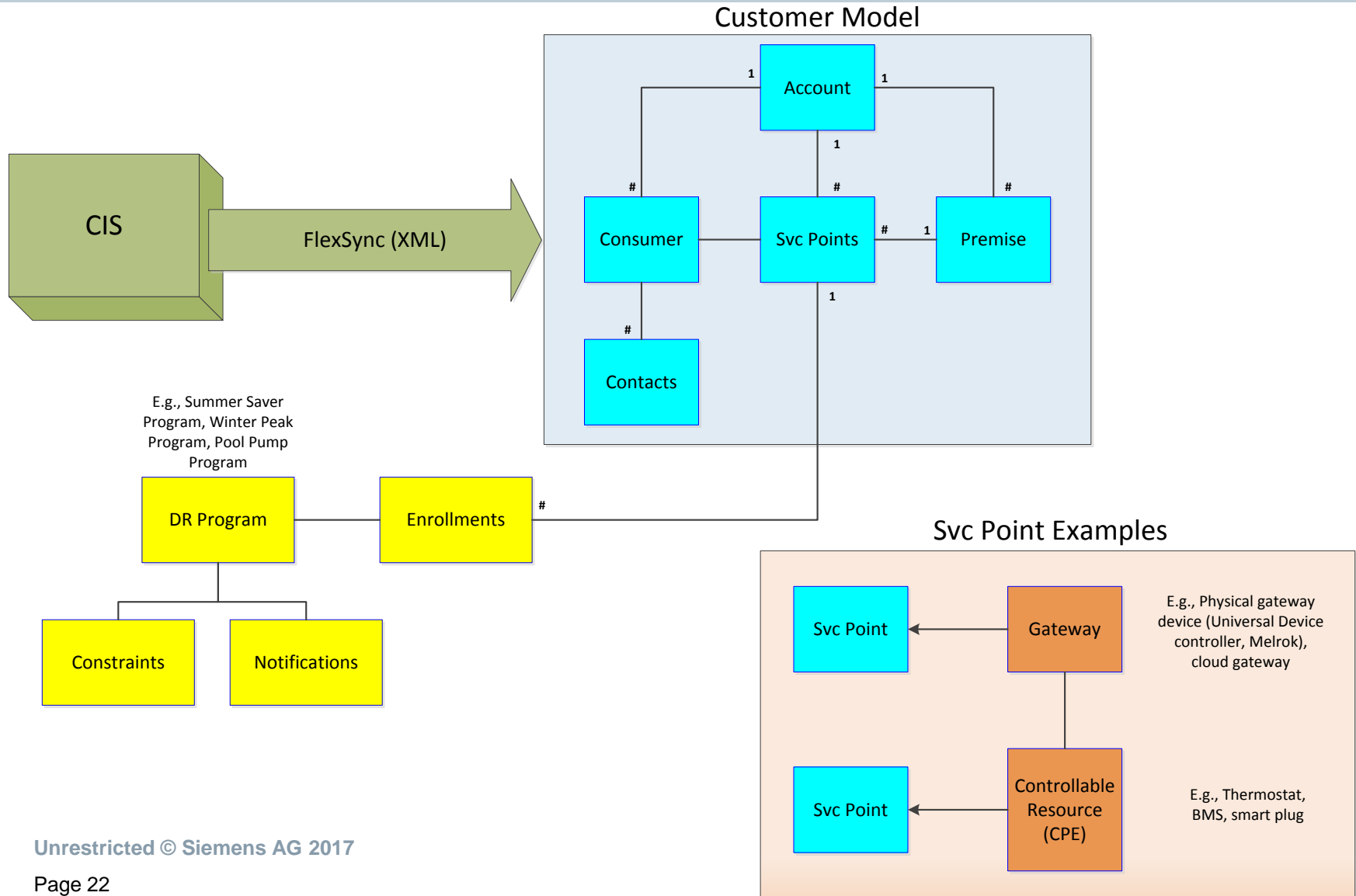
- **Notification Templates**

- Notify customers via email, SMS, voicemail, etc.
- Create templates and assign them to programs
- **Note: NB Power has chosen not to issue notifications to their customers**

# DEMS Structural Overview

- **Enrollments**
  - How a **Program** gets tied to the **Customer** model
  - **Svcpts** get enrolled into 1+ programs
  - There may be additional **Constraints** within an **Enrollment**
    - Subset of the already defined **Program Constraints**
    - Imported through RDU (XML system)

# DEMS Structural Overview



# DEMS Structural Overview

- **Aggregations / Service Point Groups**
  - A grouping of assets (**svcpts**) based on pre-defined business rules
    - Geographic region
    - Substation name
    - Postal Code
  - Can have nested levels of groups (groups within groups)
  - Parameters
    - Metadata (name-value pairs) – not relevant to how events get dispatched
  - Enables targeted Demand Response Programs



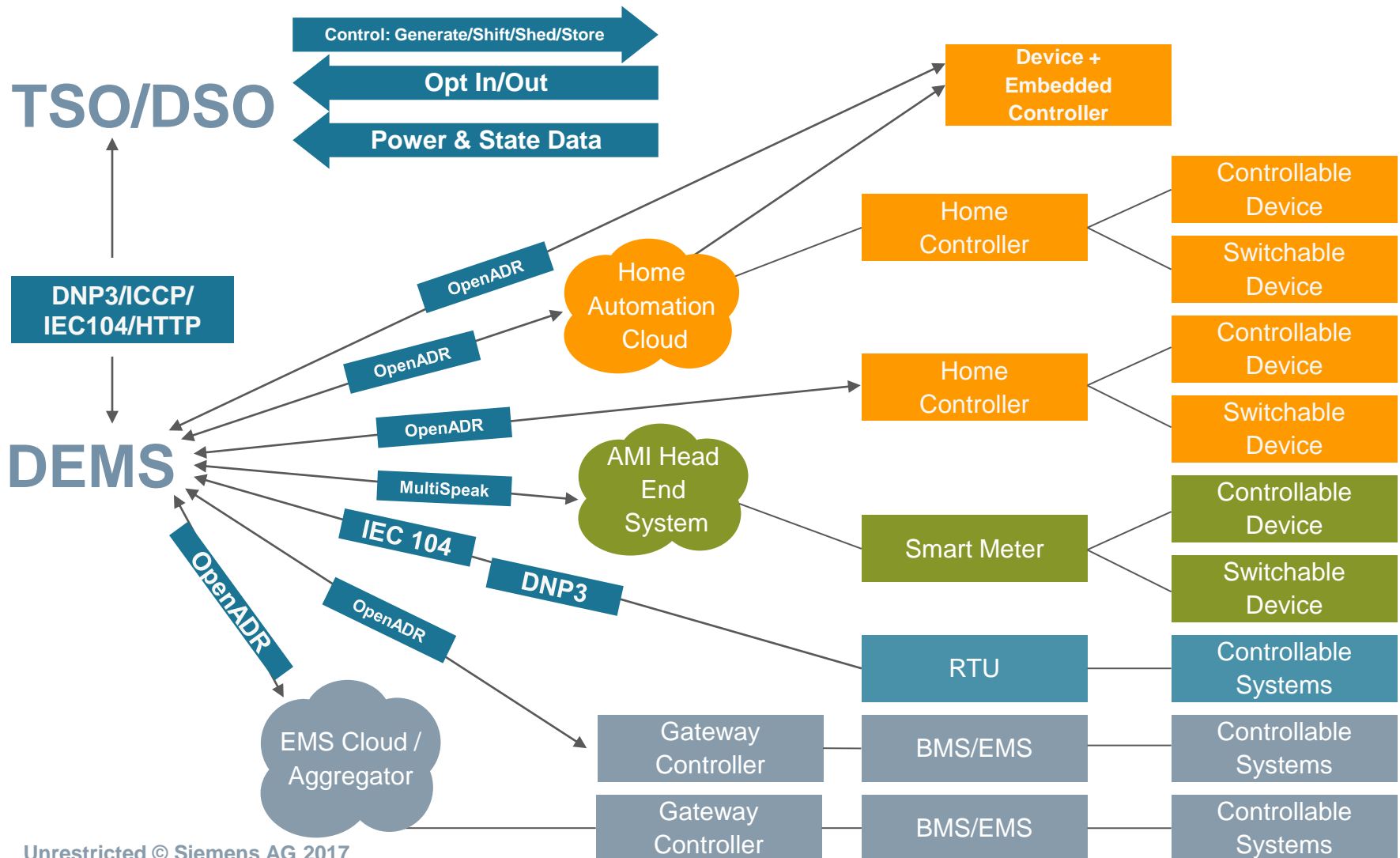
# DEMS Structural Overview

- **How to create Service Point Groups (SPGs)**
  - Explicit
    - Select specific **svcpts** to belong to the **SPG**
  - Rules-Based
    - Will create an **SPG** from existing **svcpts** that meet specified criteria
    - E.g., “All assets enrolled in program X”

# DEMS Structural Overview

- **Dispatching Demand Response Events**
  - DEMS dispatches DR (Demand Response) events to **SPGs** through **enrollments**
    - **SPGs** are aggregated groups of **svcpts**
    - **Enrollments** ties **svcpts** to **Programs**
    - **Programs** have **Constraints** which govern how they are run
- How does DEMS communicate with external devices and services?

# Protocols for Interoperability with DEMS



# OpenADR Overview

**SIEMENS**



# OpenADR Overview

An OpenADR network consists of two basic components:

## **A Virtual Top Node**

- or a VTN
- Server which publishes demand response-related information
- Example:  
Siemens DEMS

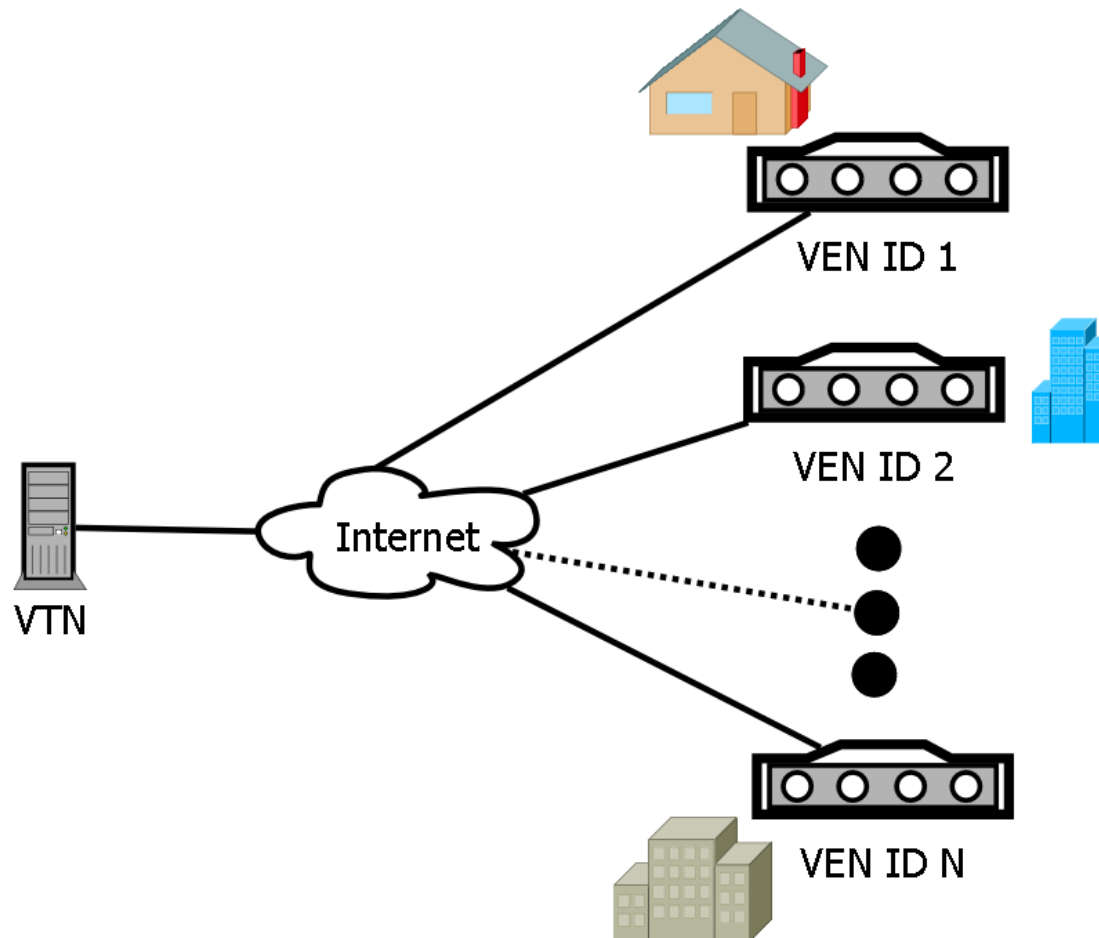
## **One or more Virtual End Node(s)**

- or VEN(s)
- Automated client which subscribes to demand response-related information published by the VTN
- Examples: Thermostats, Water Heaters, BMSs
- A VEN may also be a VTN (BMS, Home Controller, Cloud VEN)

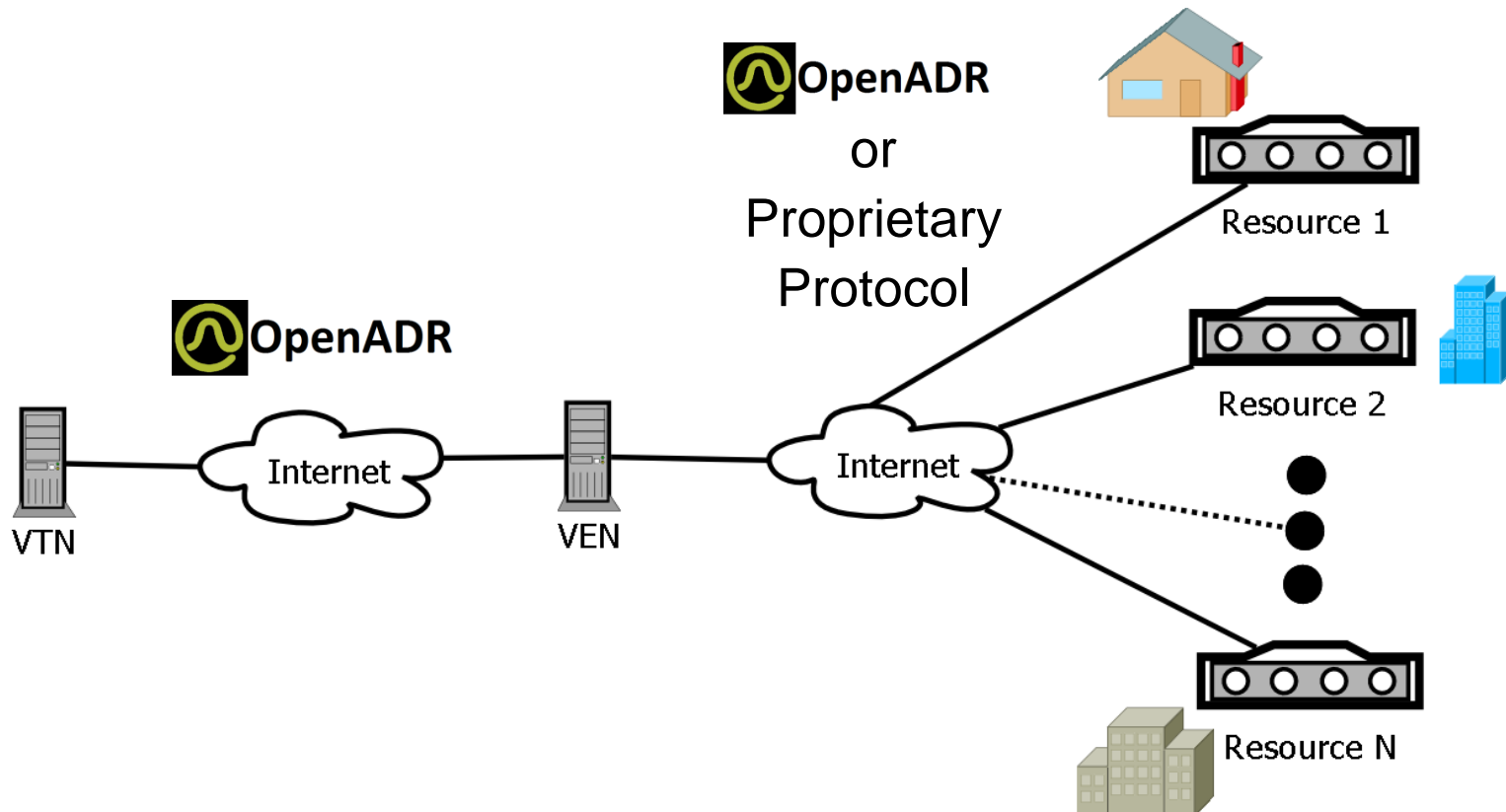
**All communication takes place over secure socket layer (SSL), with agreed certificates exchanged out-of-band.**



## OpenADR Overview :: Direct VTN to VEN Network



# OpenADR Overview :: VTN to Cloud VEN Network



## Siemens Contact Information



### **Hazem Muna Alsadi**

Tel: 1 506 260 5699

E: [hazem.munaalsadi@siemens.com](mailto:hazem.munaalsadi@siemens.com)

### **Joel Richard**

Tel: 1 506 260 8357

E: [joel.richard@siemens.com](mailto:joel.richard@siemens.com)

# OPENADR REFERENCE SLIDES

**SIEMENS**





# OpenADR Messaging

**SIEMENS**



# OpenADR Messaging :: Simple HTTP Application Protocol

- OpenADR supports both XMPP and Simple HTTP as an Application-layer Protocol for establishing and maintaining an OpenADR network
- Within Simple HTTP, both pull and push models of communication are possible
- However, DEMS exclusively uses the Simple HTTP pull model of communication
- VEN always POSTs messages to the VTN, and VTN always just responds within this already established connection

## OpenADR Messaging:: XML Message Contents

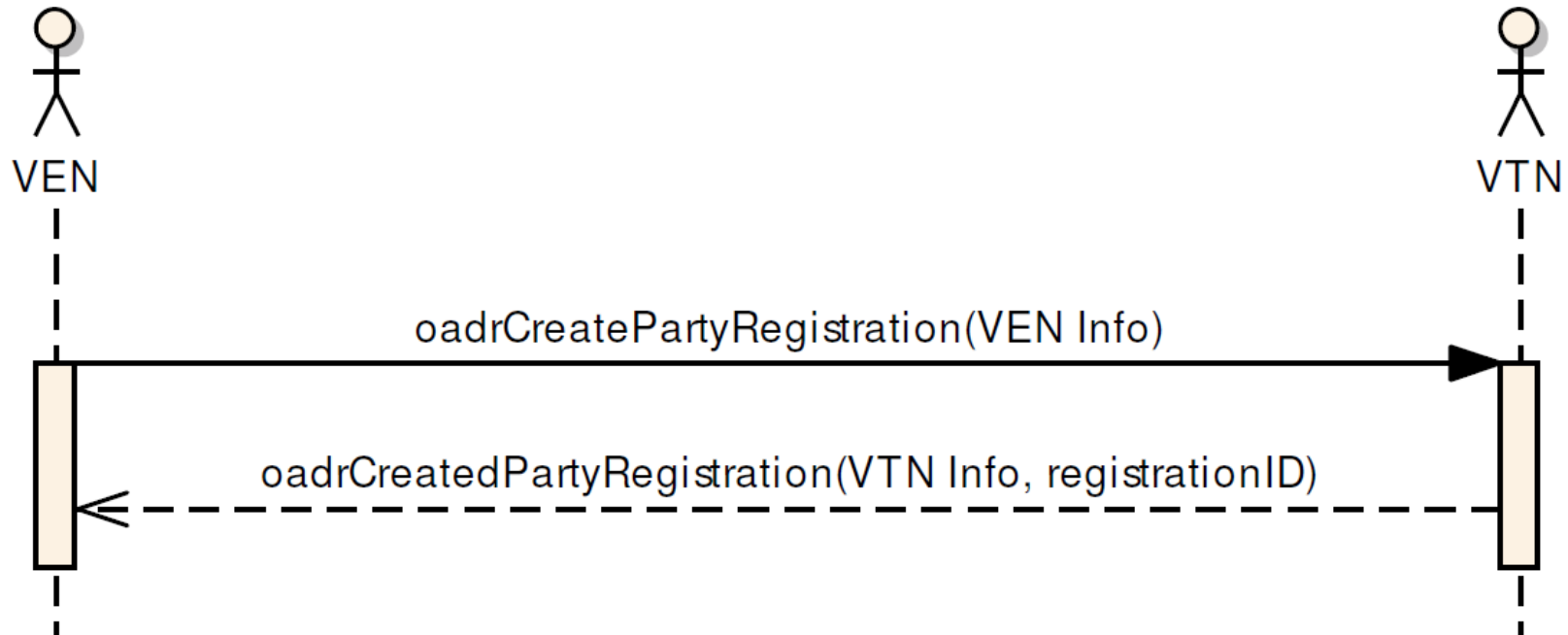
- OpenADR exclusively uses XML messages in communications
- XML messages conform to a publicly available schema
- Schema is composed of:
  - 1 new OpenADR schema, which in turn references:
    - 15 existing standard energy industry schemas
- All messages are wrapped in:
  - OadrSignedObject element
  - Which is then wrapped in an:
    - OadrPayload XML root element



## OpenADR Messaging:: Service Contexts

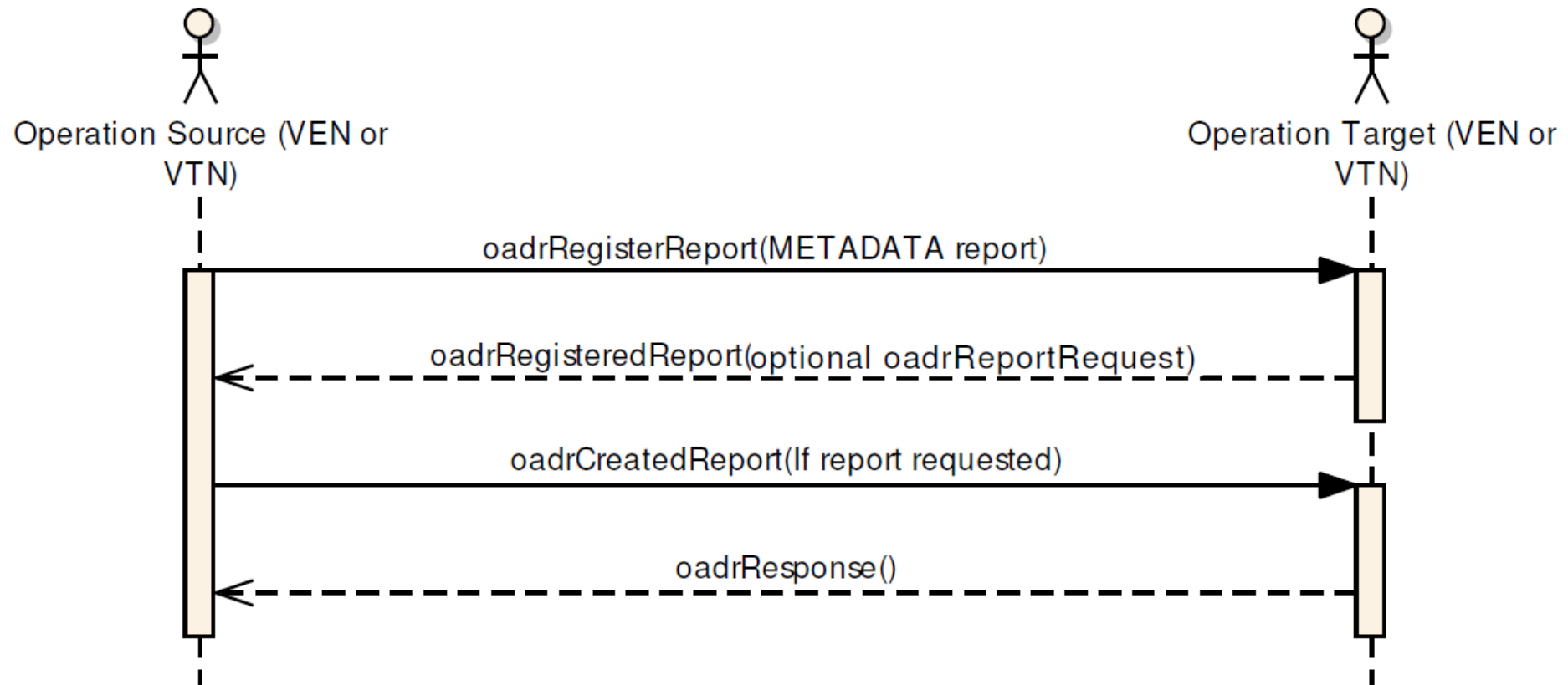
- OpenADR is split into several web application service contexts of which DEMS supports the following 4:
  - /OpenADR2/Simple/2.0b/**EiEvent**
    - Request Event(s) (NB: use Poll instead, except at boot)
    - OptIn/OptOut of Event(s)
  - /OpenADR2/Simple/2.0b/**EiReport**
    - Register Report(s) (define available reports with metadata)
    - Update Report(s) (update VTN with a requested reported)
  - /OpenADR2/Simple/2.0b/**EiRegisterParty**
    - Query Registration (Learn how to or if already registered)
    - Register a Party (Device announces presence in network)
    - Cancel Party Registration
  - /OpenADR2/Simple/2.0b/**OadrPoll**
    - Request queued information (Events, Report Requests, etc.)
    - VEN should call once per Poll Frequency

# OpenADR Messaging:: Registration (Required once, immediately following power on)



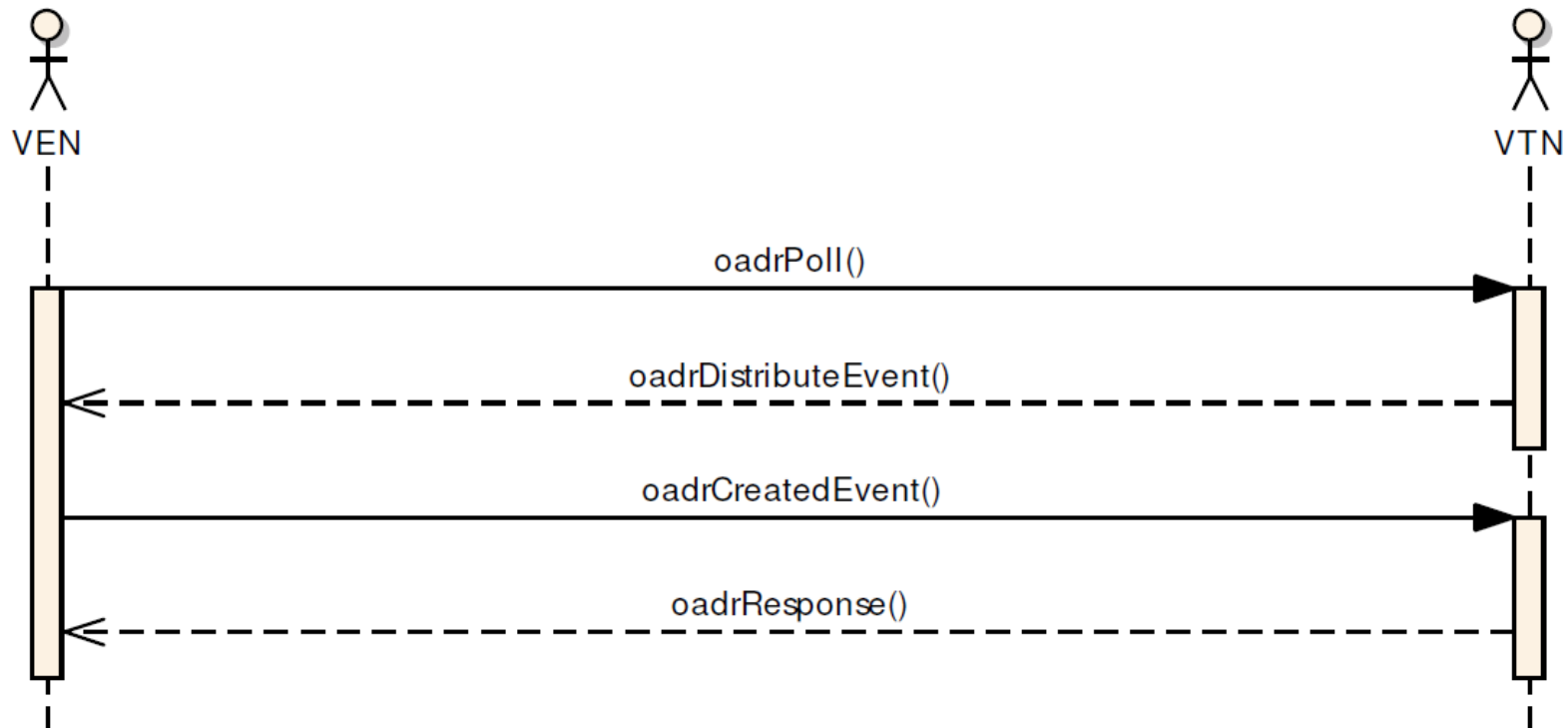
Source: OpenADR 2.0 Profile Specification B Profile, OpenADR Alliance

# OpenADR Messaging :: Report Registration (Required once, immediately following registration)



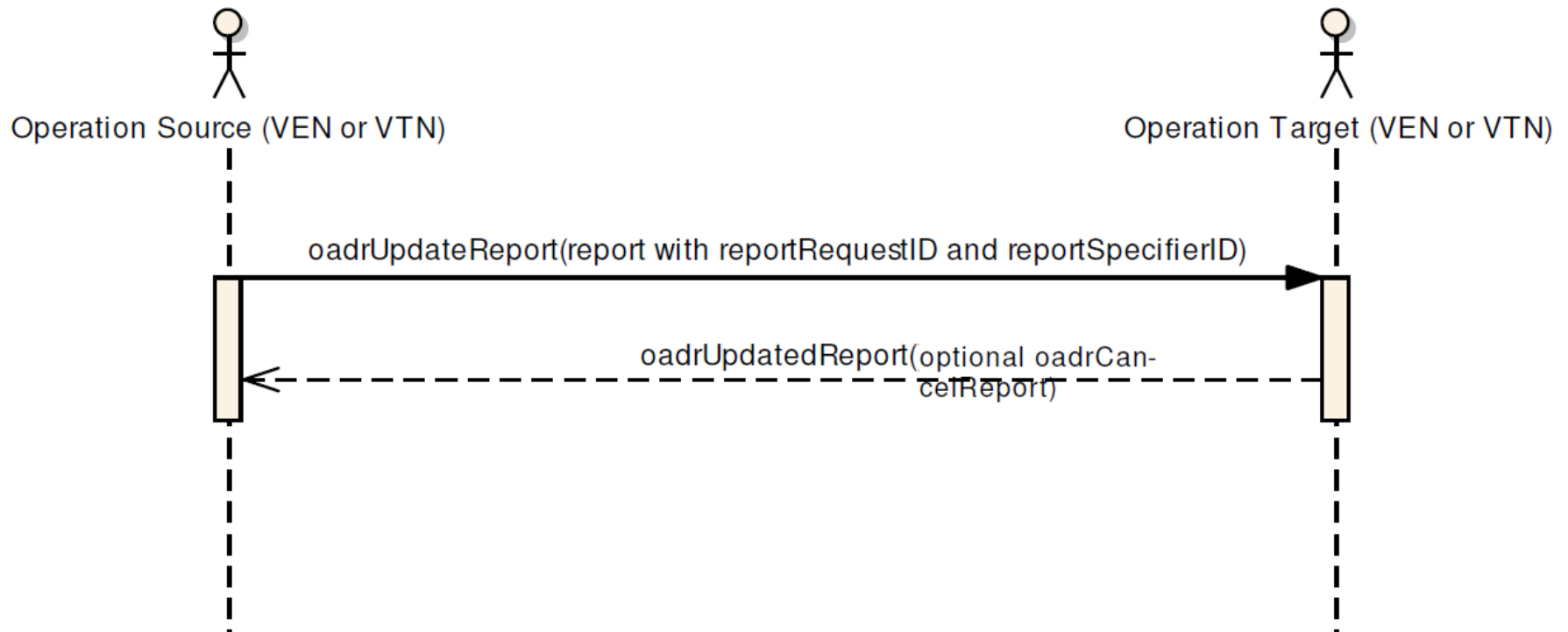
Source: OpenADR 2.0 Profile Specification B Profile, OpenADR Alliance

# OpenADR Messaging :: OadrPoll (main communication method) (Once per Poll Frequency time elapsing)



Source: OpenADR 2.0 Profile Specification B Profile, OpenADR Alliance

# OpenADR Messaging:: OadrUpdateReport (Once per ReportBackDuration time elapsing)



Source: OpenADR 2.0 Profile Specification B Profile, OpenADR Alliance

# OpenADR Security

SIEMENS



- VTN will provide to VEN out-of-band:
  - VEN Client Certificate and Key
    - Last 20 Hex Digits of this Certificate's SHA256 signature will be known as the VEN's fingerprint
    - Signed by VEN intermediate certificate, which is in turn signed by this OpenADR network's root certificate
  - VTN Intermediate Certificate
    - Certificate which signed the VTN's certificate
  - Network Root Certificate
    - Root authority certificate that has signed the VEN Intermediate Certificate and VTN Intermediate Certificate
  - A VEN ID



## OpenADR Security :: Authentication

- An TLS/SSL handshake will occur whenever the VEN attempts to POST to the VTN
- During handshake, VEN will receive a certificate request, and be told that the VTN trusts the Root certificate, and the VEN intermediate certificate
- VEN should respond with its certificate signed by the VEN intermediate certificate, that it received out-of-band
- (Optionally) VEN may verify that the VTN's certificate is signed by the VTN intermediate certificate, and that its hostname matches, to prevent a "man-in-the-middle"
- During registration, the VEN's certificate's fingerprint will be extracted and validated against the VTN's out-of-band received fingerprint for that VEN

# OpenADR Error Handling



## OpenADR Error Handling :: Detecting Errors

- VTN will indicate a successful POST from the VEN, by responding with an “application level error code” of 200 in the EiResponse sub-element of its response
- If an error occurs with the POST, and if possible, the VTN will attempt to respond, in the XML message body, with an application level error code, such as 452, and an error description message for human log consumption
- 452, in particular, indicates INVALID ID, such as the VTN believes the VEN gave an incorrect VEN ID, etc.
- 12 predefined error codes exist in the specification, and 8 are open for VTNs or VENs to define as they wish

## OpenADR Error Handling :: Handling Errors

- Almost all messages in OpenADR are idempotent
- If the VEN believes an error is a false-positive, it may simply resend the message that caused the error code
- However, the VEN should implement truncated exponential backoff from any error, in order to:
  - prevent flooding the network with retries
  - give VTN time to recover from any temporary overload
- Exponential backoff requires waiting randomly 0.9 to 1.1 seconds, and error reoccurs, waiting randomly 1.8 to 2.2 seconds, and if error reoccurs, 3.6 to 4.4 seconds, ...
- This backoff should truncate at a maximum of the Poll Frequency (typically 5 minutes), and resend each Poll Frequency until a successful response is received



# OpenADR Sample Messages



# OpenADR Sample Messages:: OadrQueryRegistration

```
<?xml version="1.0" encoding="UTF-8"?>
<oadr:oadrPayload>
  <oadr:oadrSignedObject>
    <oadr:oadrQueryRegistration ei:schemaVersion="2.0b">
      <pyld:requestID>12345</pyld:requestID>
    </oadr:oadrQueryRegistration>
  </oadr:oadrSignedObject>
</oadr:oadrPayload>
```

# OpenADR Sample Messages:: OadrCreatedPartyRegistration

```
<?xml version="1.0" encoding="UTF-8"?>
<oadr:oadrPayload>
<oadr:oadrSignedObject>
  <oadr:oadrCreatePartyRegistration ei:schemaVersion="2.0b">
    <pyld:requestID>REQ_123</pyld:requestID>
    <ei:venID/>
    <oadr:oadrProfileName>2.0b</oadr:oadrProfileName>
    <oadr:oadrTransportName>simpleHttp</oadr:oadrTransportName>
    <oadr:oadrTransportAddress>http://www.myaddress.com
    </oadr:oadrTransportAddress>
    <oadr:oadrReportOnly>>false</oadr:oadrReportOnly>
    <oadr:oadrXmlSignature>>false</oadr:oadrXmlSignature>
    <oadr:oadrVenName>MyVEN</oadr:oadrVenName>
    <oadr:oadrHttpPullModel>>true</oadr:oadrHttpPullModel>
  </oadr:oadrCreatePartyRegistration>
</oadr:oadrSignedObject>
</oadr:oadrPayload>
```