



The Green Button Initiative

white house inspired.
industry led.

In response to the Obama Administration's call to action, 150+ utilities and service providers commit to provide 60+ million homes and businesses with their own energy usage information in the consensus industry standard Green Button format
www.greenbuttondata.org



Photo credit: Silver Spring Networks CustomerIQ™ Energy Portal

John Teeter, john.teeter@nist.gov

Marty Burns, martin.burns@nist.gov

Smart Grid and Cyber-Physical Systems Program Office
Engineering Laboratory, NIST



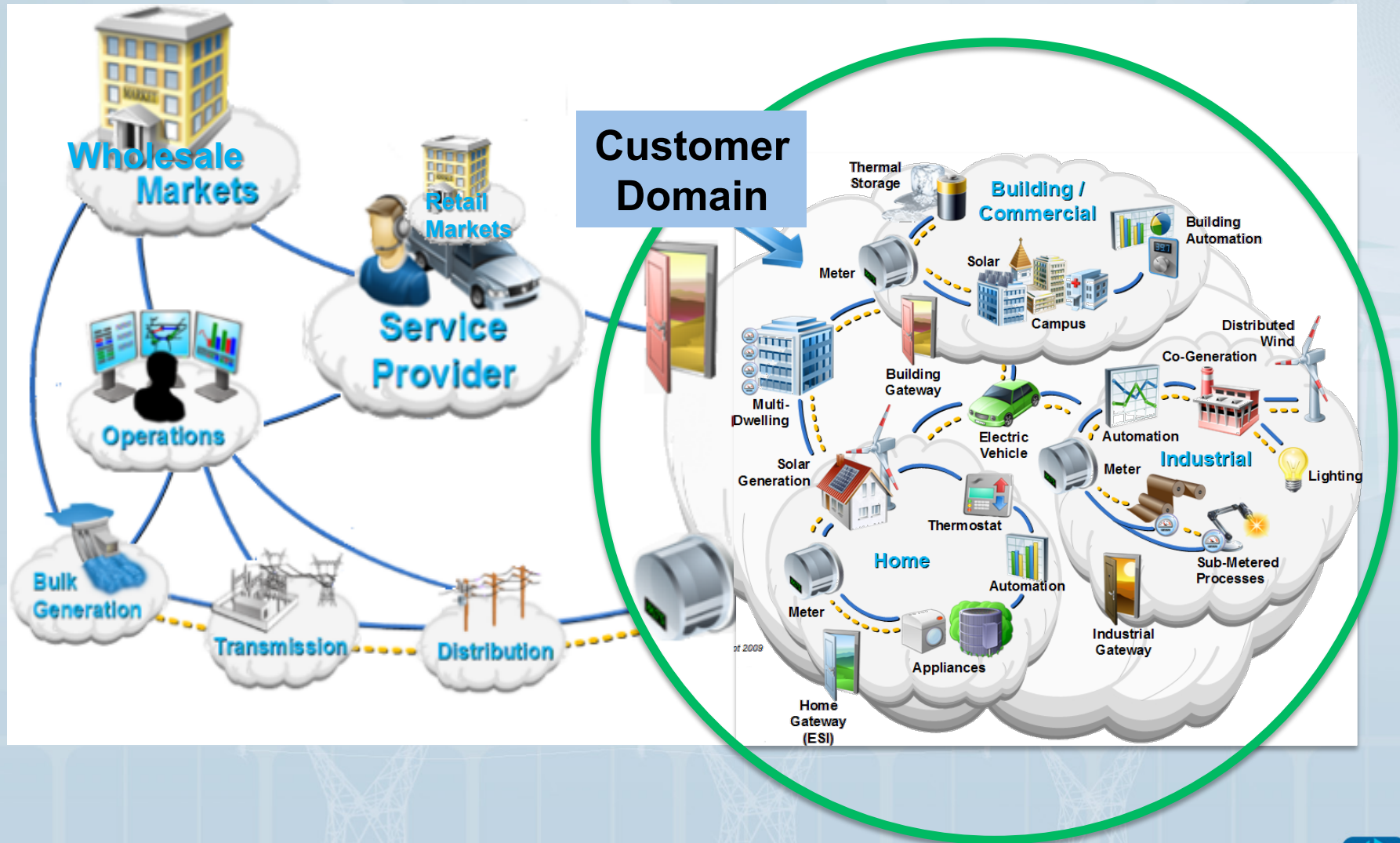
Green Button Technology

- Consistency
- Interoperability
- Scaling to Market





Customer Domain





Current Status

- Green Button XML representation is stable.
Review it at: [Ref: espiDerived.xsd](#)
- Download My Data Testing and Certification is ready.
 - [Validate your XML Data](#)
 - [Review the DMD Test Plan](#)
- Connect My Data implementations are underway.
Implementation Agreements:
 - [GreenButtonAuthorization.docx](#)
 - [GreenButtonAtomLinks.docx](#)
 - [espiDerived.xsd](#)
- Green Button Sandbox
 - <https://services.greenbuttondata.org>
 - [Deploy your own sandbox](#)
- [Open Data by Design Challenge](#) (\$17,000 in Prizes)





Definitions

EUI - One or more components of energy usage information including UsagePoint, MeterReading, IntervalBlock, ReadingType, LocalTimeParameters, ElectricPowerUsageSummary, and ElectricPowerQualitySummary

ApplicationInformation - A single data structure that contains definitions of all information shared between a Data Custodian and Authorized Third Party that govern their communications relationship.

Authorization - A single data structure that represents the three-party agreement between Retail Customer, Data Custodian, and Authorized Third Party which covers the life of the authorization for EUI exchange.

[Ref: GreenButtonAuthorization.docx Section 1.1](#)





Diversity of information in EUI

Dimensions

Kinds of Data

- Measurements of power, energy, gas, water, ...
- Quality: Raw, validated, estimated, ...
- Source: Meter near real-time, utility back end, third party
- Economics: Consumers need to know the cost of their consumed power (but we did not construct a pricing model)
- Identification: by customer, device, location
- Readings
- Interval data
- Summary Information
- Power Quality Metrics

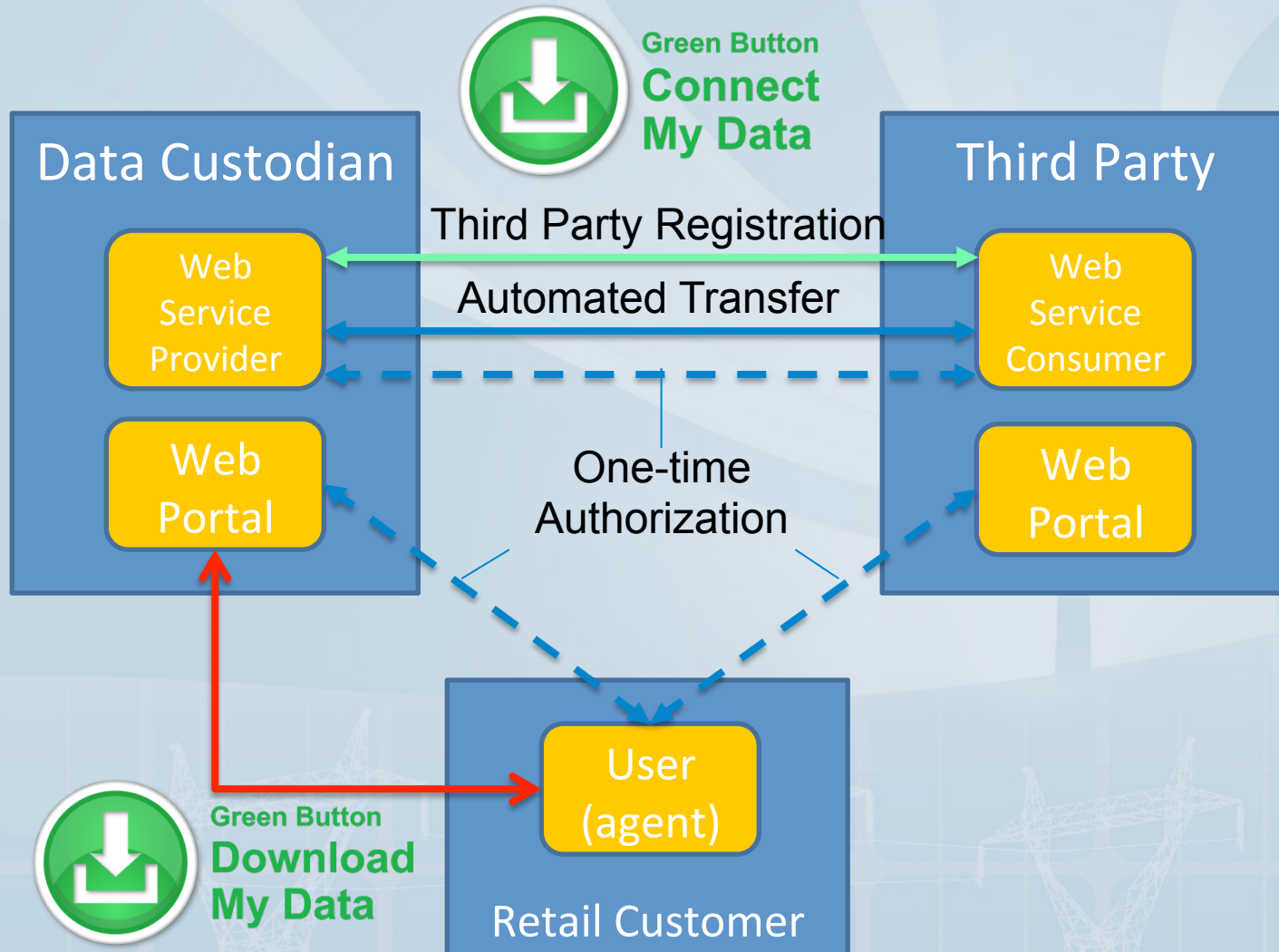
The screenshot displays a web interface with several sections: 'Energy Usage Information' showing location details, 'Water Reading Information' showing meter type, 'Summary of Electric Power Usage Information' with billing and consumption data, and 'Hourly Electricity Usage' with a table of hourly data.

Energy consumption time period	Usage (kWh)	Cost (\$)	Events occurred
2011-09-01 00:00:00 to 2011-09-01 01:00:00	1	0.12	
2011-09-01 01:00:00 to 2011-09-01 02:00:00	1	0.12	
2011-09-01 02:00:00 to 2011-09-01 03:00:00	1	0.12	
2011-09-01 03:00:00 to 2011-09-01 04:00:00	1	0.12	
2011-09-01 04:00:00 to 2011-09-01 05:00:00	1	0.12	
2011-09-01 05:00:00 to 2011-09-01 06:00:00	3	0.36	
2011-09-01 06:00:00 to 2011-09-01 07:00:00	3	0.36	



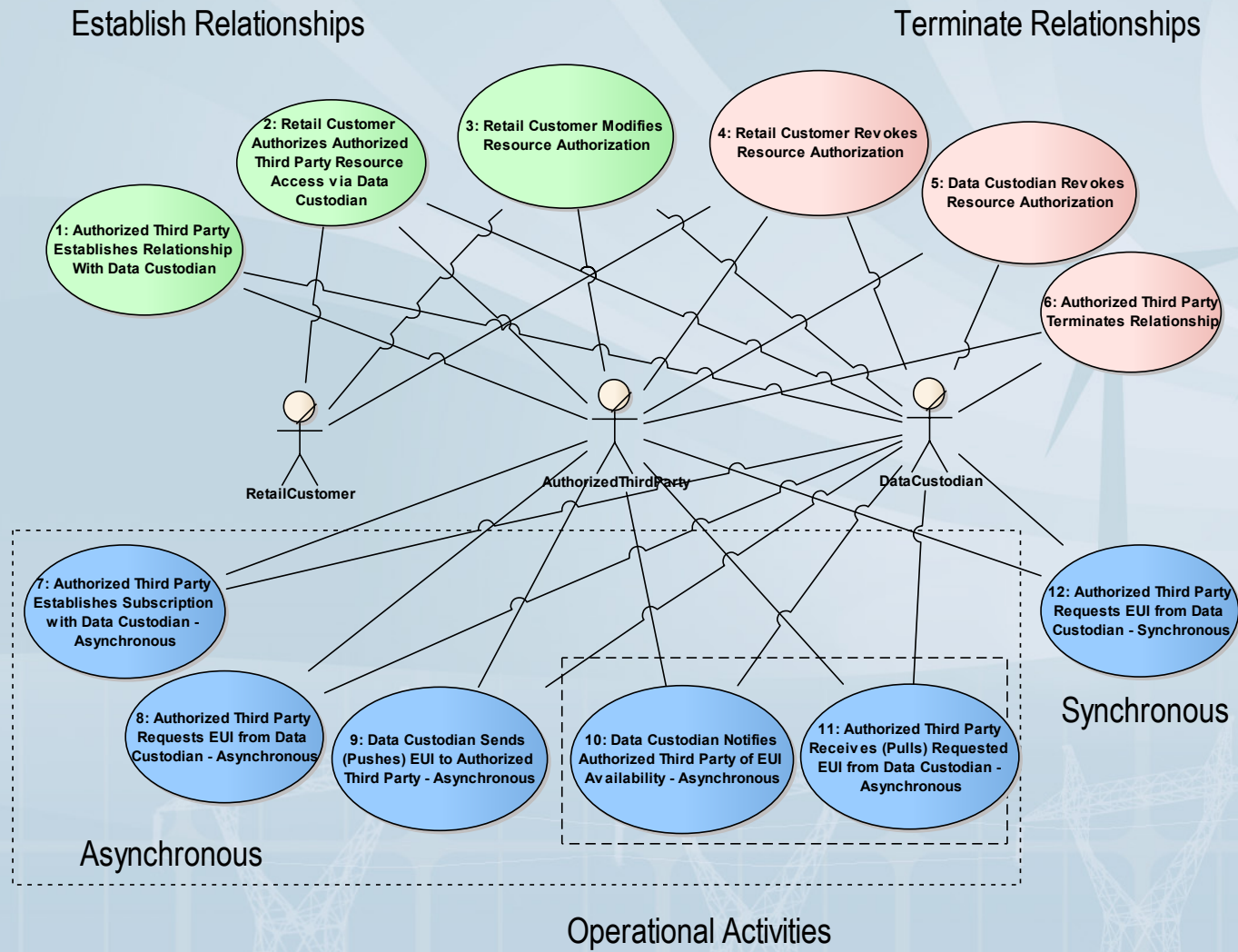


Green Button Data Exchange





Green Button Connect My Data





Green Button Sandbox

Open source implementation

- [RESTful API functions](#)
- Determined by “Function Block”
- Focus today on Batch transfers

[Services.greenbuttondata.org](https://services.greenbuttondata.org)

- Individual Retail Customer Accounts
 - single and multiple Usage Points
- Portfolio of Accounts⁽¹⁾
- Scope Parameters⁽¹⁾

Making a private sandbox

- [Download/Run the Virtual Machine](#)
- [Clone/Build/Run the Github repositories](#)
- Deploy the Amazon Machine Image (AMI) ⁽¹⁾





Third Party Perspective

Registration of Third Party ←←

- Manual and not yet standardized
- Each Data Custodian will have unique process
- Moving toward [OAuth2 Dynamic Client Registration](#)

Authorization of Data Access

- Individual Retail Customer Accounts
 - single and multiple Usage Points
- Scope Parameters
- Portfolio of Accounts

Process of Data Exchange

- Notifications
- OAuth2 access_tokens
- Pull Scenarios



Registration of Third Party

Manual and not yet standardized

- Relationship between Data Custodian and Third Party
- Information captured in ApplicationInformation Resource
- Updates and Termination

Each Data Custodian will have unique process

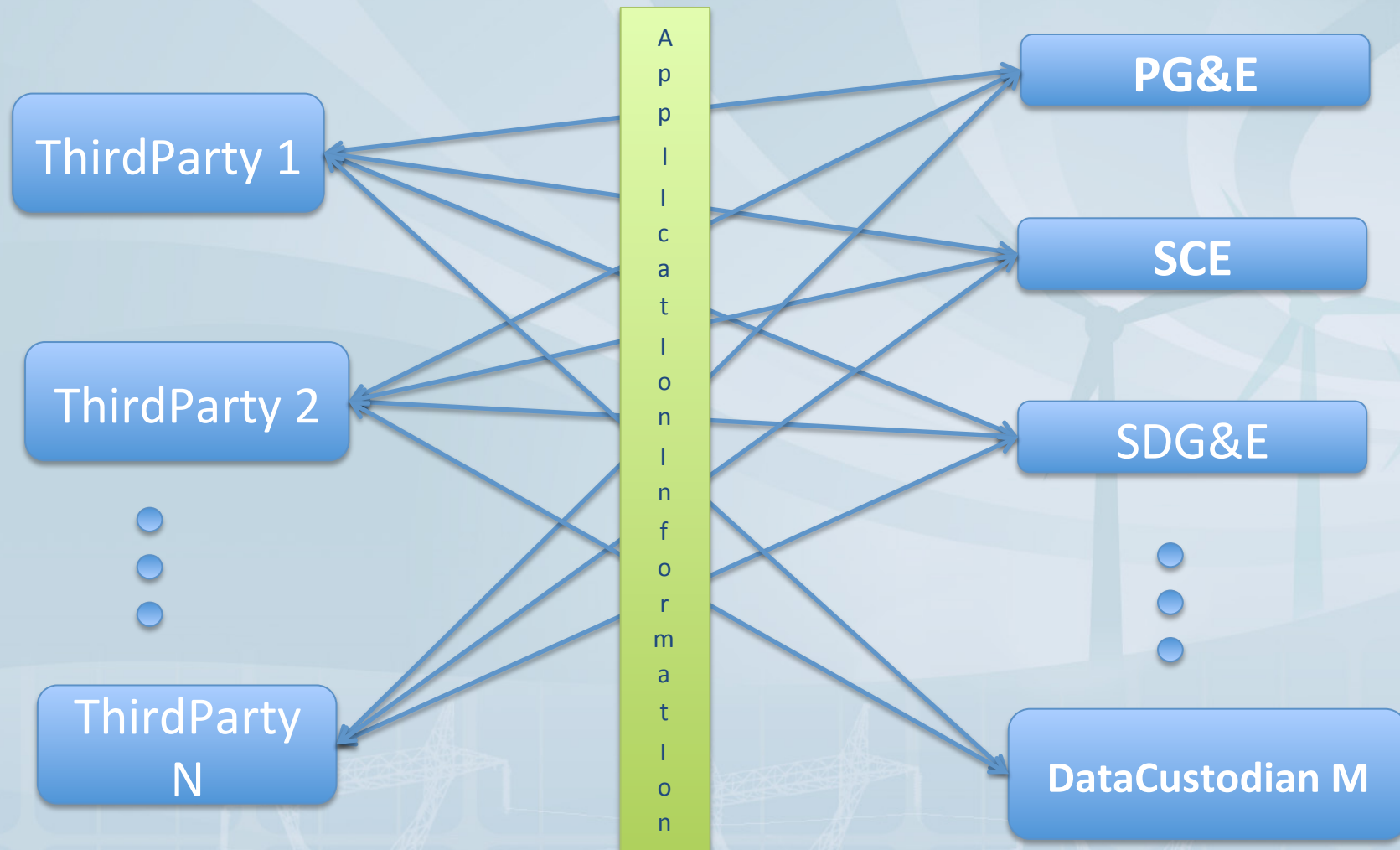
- May involve manual steps
- Accountability and Validation
- Collecting ApplicationInformation data

Moving toward [OAuth2 Dynamic Client Registration](#)

- Not yet IETF RFC – Working Draft
- Rapid movement to support when ratified
- Open source implementation soon



ApplicationInformation



Each Relationship shares an
ApplicationInformation Resource



ApplicationInformation

- The result of the registration process is the creation of an ApplicationInformation object (with @40 Attributes)
- A shared view of the capabilities and relevant endpoints maintained by both parties (the contract b/t the actors)
- Updates and access (via RESTful https requests) are available. (manageable for dynamic configuration)
- Detail in references

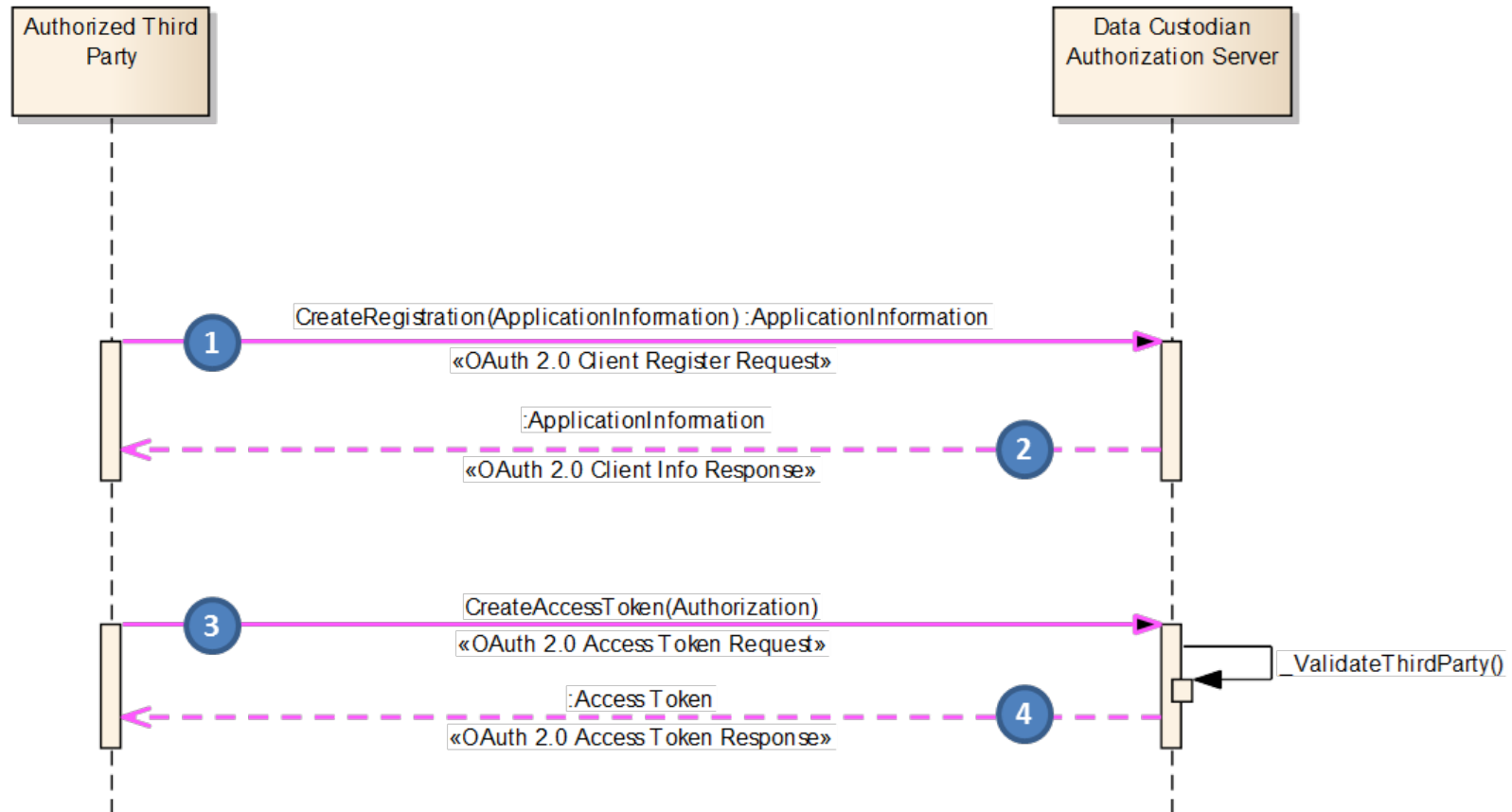
[Ref: Authorization.docx Appendix D](#)

[Ref: espiDerived.xsd](#)



Green Button Connect My Data Registration

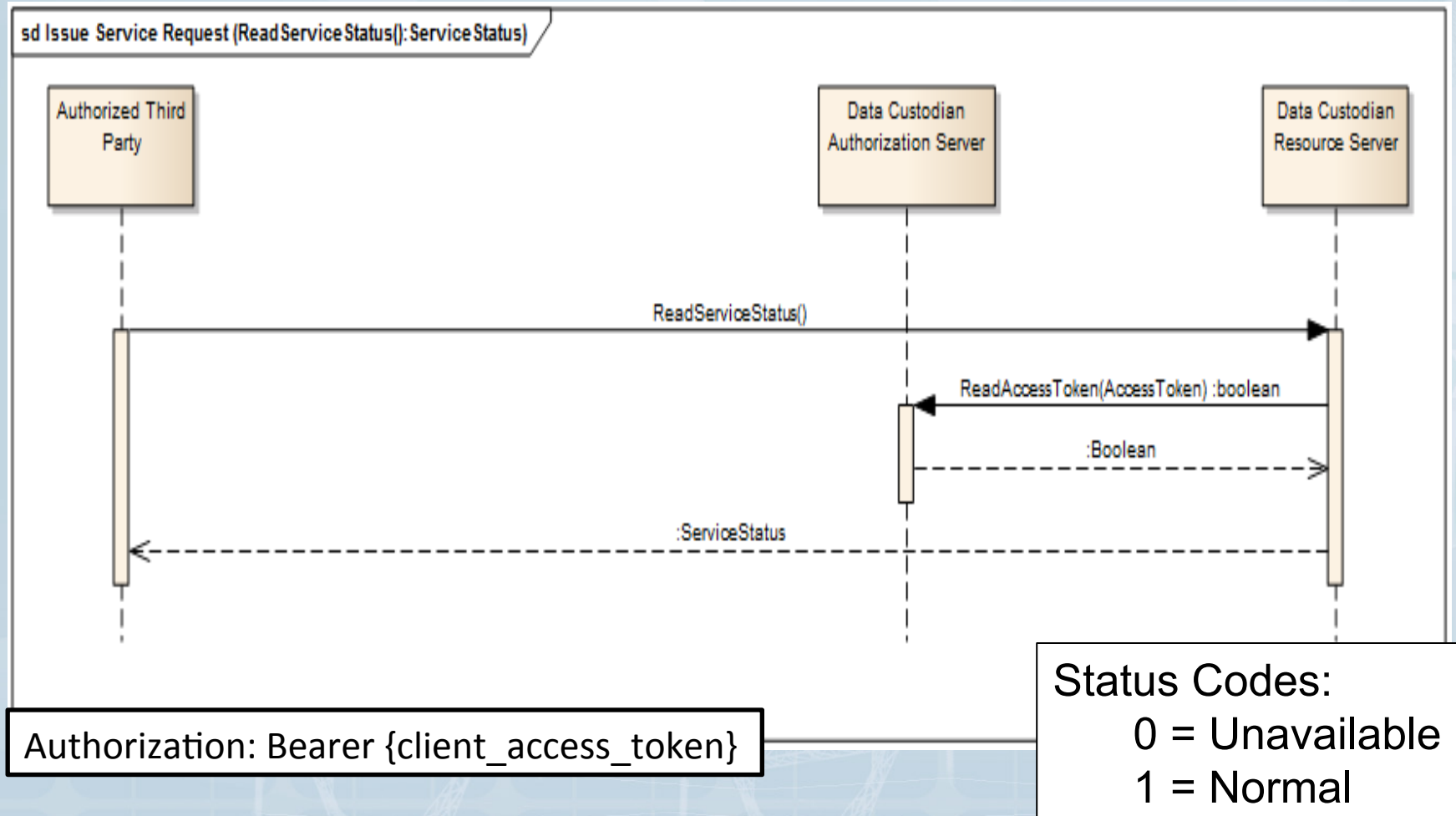
sd Authorized Third Party Establishes Relationship With Data Custodian



[Ref: Authorization.docx Section 2.2](#)



Green Button Connect My Data Status and Notification



[Ref: Authorization.docx Section 2.2.2](#)





Third Party Perspective

Registration of Third Party

- Manual and not yet standardized
- Each Data Custodian will have unique process
- Moving toward [OAuth2 Dynamic Client Registration](#)

Authorization of Data Access ←←

- Individual Retail Customer Accounts
 - single and multiple Usage Points
- Portfolio of Accounts
- Scope Parameters

Process of Data Exchange

- Notifications
- OAuth2 access_tokens
- Pull Scenarios



Authorization of Data Access

OAuth2 Authorization Patterns

- Using client_credential and authorization access codes
- OAuth2 scope negotiation

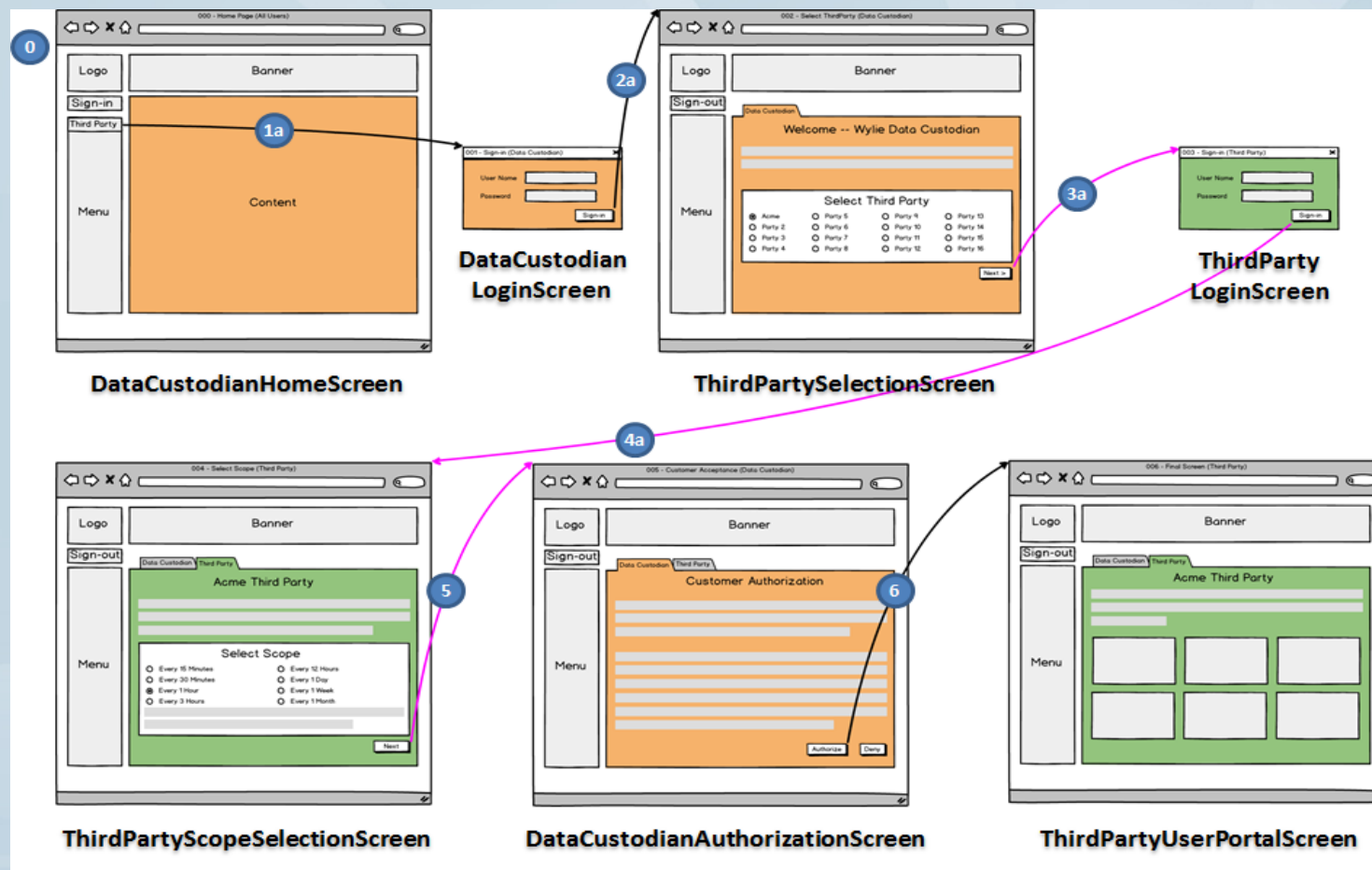
Authorization of Data Access

- Single Retail Customer/UsagePoint
- Message Patterns
- Scope Parameters

Scope parameter details

- Resource Terms
- Function Blocks supported
- Kinds of Data (Intervals, durations, ...)

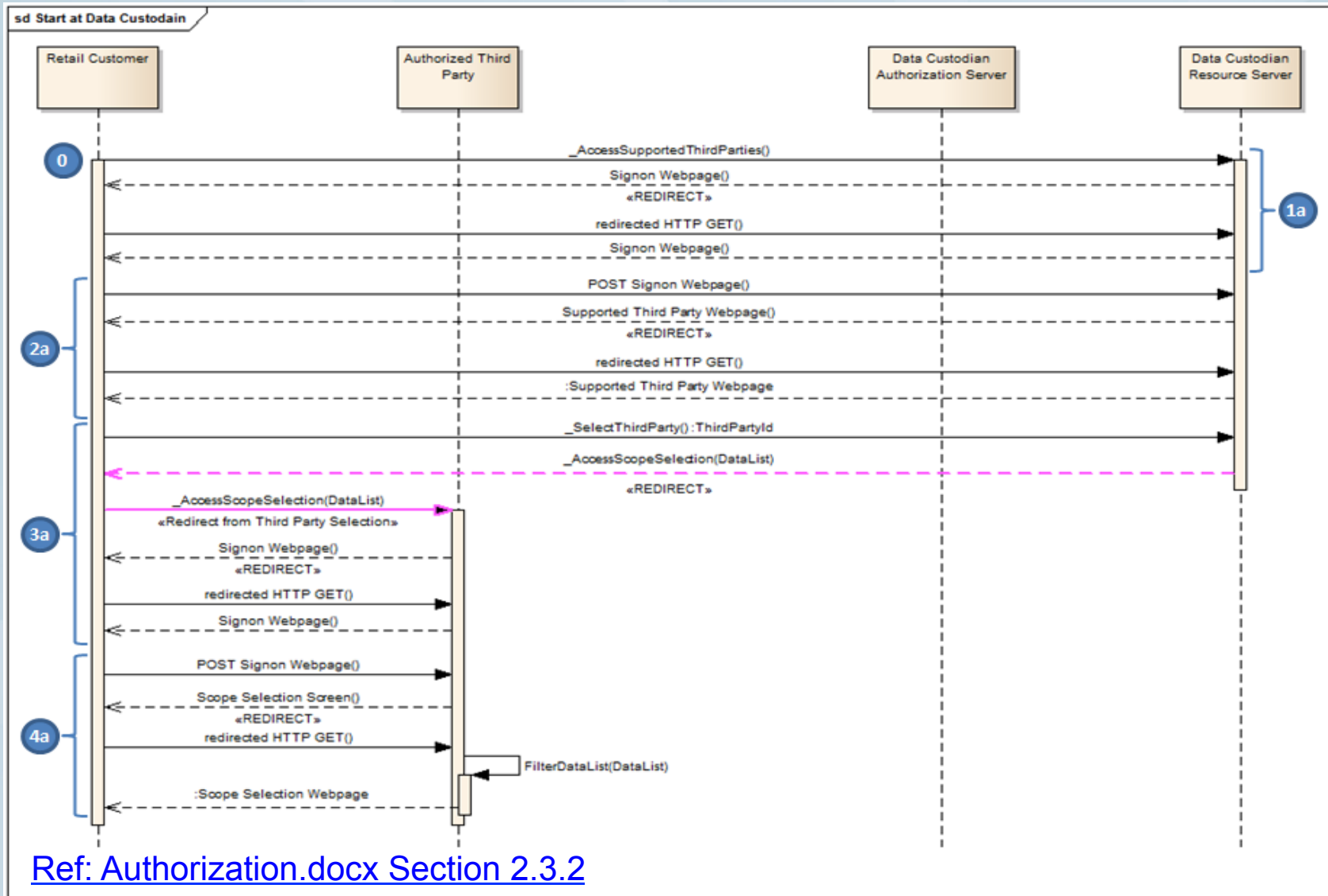
Green Button Connect My Data User Experience Guidelines



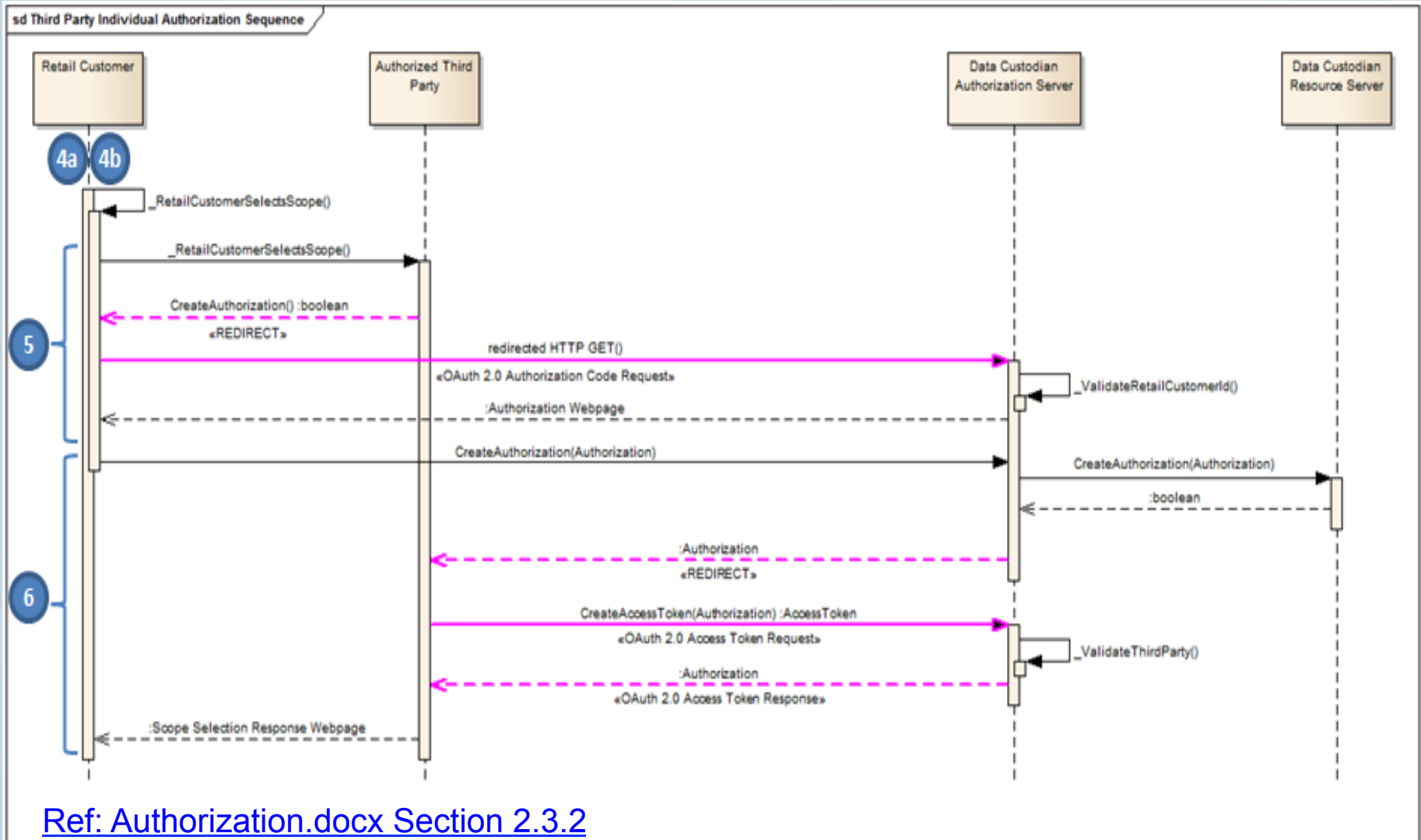
[Ref: Authorization.docx Section 2.3.2](#)



Green Button Connect My Data RetailCustomer Authorization



Green Button Connect My Data RetailCustomer Authorization





Access Tokens Obtained

access_token: allocated by datacustodian for individual account authorizations

refresh_token: allocated at the time of authorization and used to renew an access_token

Note that access tokens are entirely under the control of the Data Custodian which can determine the life cycle of the access tokens and refresh tokens. The Authorized Third Party is required to react to the changes in the access token and/or refresh token as they occur.

Use:

GET {ResourcePath}<resource>/{ResourceID} HTTP/1.1

Host: {ResourceServer}

Content-Type: application/atom+xml

Authorization: Bearer {AccessToken}

[Ref: Authorization.docx Section 1.2.2](#)

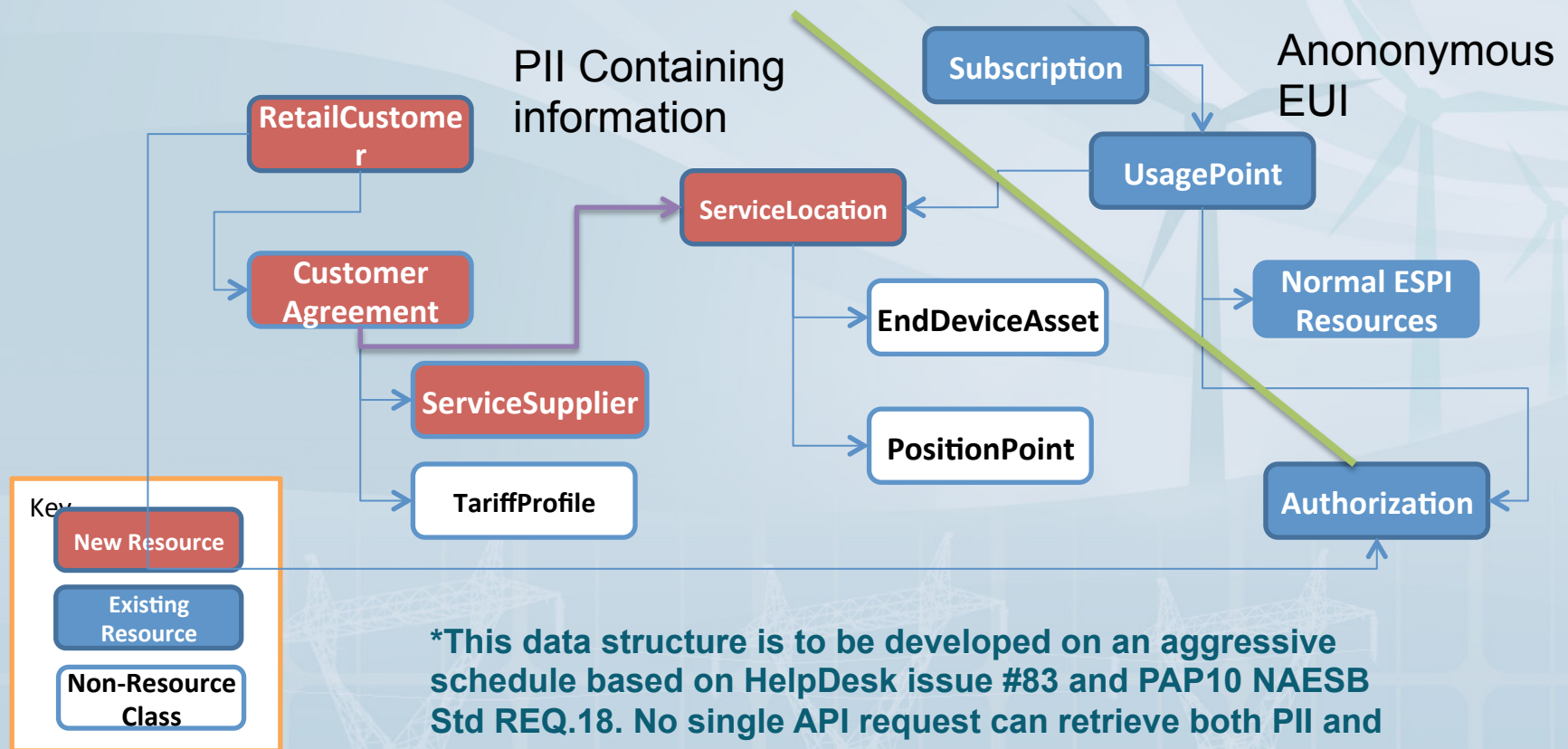




Portfolio and Multiple Accounts

Work is currently underway: [OpenADE](#)

- To extend the information model
- Insure PII integrity of ESPI



Ref: [OpenADE Meeting Notes: Slide 05](#)



Example Scope String

Scope = "FB=1_3_4_5_8_13_18_19_31_34_35_39;
IntervalDuration=900_3600;BlockDuration=Daily;
HistoryLength= 34128000;SubscriptionFrequency=Daily;
AccountCollection=5;BR=1;"

- Identifying which Function Blocks
- Looking at 15 min and 1 hour interval data blocked daily
- An initial history of 13 months, with daily feed
- When doing bulk (using bulkId = 1) there will be 5 usage points in the transfer

[Ref: Authorization.docx Section 2.6](#)

[Ref: GreenButtonCMDWorkshop: Slide 04](#)





Examples for Connect My Data

- Example Scope Strings (* subject to change)
 - SCE(*)
 - Scope = “FB=1_3_4_5_8_13_18_19_31_34_35_39;
↳ IntervalDuration=900_3600;BlockDuration=Daily; HistoryLength=34128000;SubscriptionFrequency=Daily;
↳ AccountCollection=5;BR=1;”
 - PG&E(*)
 - Scope = “FB=1_3_4_5_7_8_13_14_15_18_19_31_32_34_35_37_38_39_40;
↳ IntervalDuration=300_900_3600;BlockDuration=Daily_BillingPeriod_Weekly_Monthly; HistoryLength=63072000;SubscriptionFrequency=Daily;
↳ AccountCollection=5;BR=1;”
 - SDG&E(*)
 - Scope = “FB=1_3_4_5_8_13_14_18_19_31_34_35_39_40;
↳ IntervalDuration=300_900_3600;BlockDuration=Daily_BillingPeriod_Weekly_Monthly; HistoryLength=94608000;SubscriptionFrequency=Daily;
↳ AccountCollection=5;BR=1;”

[Ref: Authorization.docx Section 2.6](#)

[Ref: GreenButtonCMDWorkshop: Slide 04](#)





Scope Term Descriptions

Where:	
ResourceTerms	If a Bulk resource is specified via the “BR” term, the value of the {bulkID} is provided after the equals sign (“=”). There could be one or more terms in this list that express the granularity of notifications about resource changes. If the Subscription has more than one UsagePoint, the AccountCollection term can indicate the number of UsagePoints included
FBTerms	The function blocks supported
ValueTerms	These are parameterized terms
IntervalDuration	This is the minimum default length of an interval in seconds (e.g. 900 for 15 minutes, 3600 for one hour, ...)
BlockDuration	This is the length of a block that contains the intervals (based on enumeration of MacroPeriodKind in ESPI above as namedFrequency)
HistoryLength	This is the length of history buffer seconds
BulkAccountCollection	Used where the DC wants to provide for the reporting of multiple UsagePoints in a single Subscription. The number of UsagePoints is represented by the value in the assignment statement – e.g. 4 UsagePoints would be BulkAccountCollection=4.

[Ref: Authorization.docx Section 2.6](#)





Scope

[Ref: Summary Scope Table](#)

[Ref: Authorization.docx Section 2.6](#)

Term	Expansion
Scope	[FBTerms], [ValueTerms], [ResourceTerms];
FBTerms	"FB=", { [FBTerm], "_" }, FBTerm, ScopeDelimiter ;
FBTerm	"4" "5" "6" "7" "8" "9" "10" "11" "12" "15" "16" "17" "18" "19" "27" "28" "29" "31" "32" "33" "34" "35" "36" "37" "38" "39" "40" "41" "43"
ValueTerms	{ ("IntervalDuration=", namedOrNumber, { "_" , namedOrNumber }), ("BlockDuration=", namedOrNumber, { "_" , namedOrNumber }), ("HistoryLength=", nonNegativeNumber), ("SubscriptionFrequency=", nonNegativeNumber namedFrequency), ScopeDelimiter };
ResourceTerms	{ ("AccountCollection=", nonNegativeNumber) "BR=", brID), ScopeDelimiter }
ScopeDelimiter	","
namedFrequency	"billingPeriod" "daily" "monthly" "seasonal" "weekly"
namedOrNumber	nonNegativeNumber namedFrequency;
brID	Character, {Character}*;
nonNegativeNumber	digit, { digit };
Digit	0 "1" "2" "3" "4" "5" "6" "7" "8" "9" ;
Character	Digit "-" "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z" "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z" ;

[Ref: Green Button Test Datasheet](#)

[\(Ref: FunctionBLoCks by RESTful API\)](#)





Function Blocks for CMD

FunctionBlocks for Green Button		
Connect My Data	Description	Min
[FB_3] Core Green Button Connect My Data	Core Services	✓
[FB_13] Security and Privacy classes	HTTPS support	✓
[FB_14] Authorization and Authentication (OAuth)	OAuth	✓
[FB_19] Partial update data	IntervalBlocks without full data sets (Ups,MR, ...)	✓
[FB_31] Core Rest Services	Third Party Access to Subscription/Authorization	
[FB_32] Resource Level REST	Third Party Access to UsagePoints, MeterReading, ... and collections	
[FB_33] Management REST Interfaces	GET PUT POST DELETE individual resources ...	
[FB_34] SFTP for Bulk	SFTP delivery of Bulk for Bulk request	
[FB_35] REST for Bulk	Support the REST request for Bulk	
[FB_36] Third Party (Client) Dynamic Registration	Use Case 1	
[FB_37] Query Parameters		✓
[FB_38] On Demand Requests	Without Notification	
[FB_39] PUSH model	Notification followed by GET	✓
[FB_40] Offline Authorization	Allows manual Authorization without OAuth but produces Authorization resource XML	
[FB_42] Third Party Core REST Services		
[FB_43] Third Party Management REST Services		
[FB_44] Manage ApplicationInformation Resource	Allows PUT and DEL of ApplicationInformation	
[FB_41] Manage Authorization Resource	Allows PUT and DEL of Authorization	

[Ref: Green Button Test Datasheet](#)

[\(Ref: FunctionBlocks by RESTful API\)](#)





Third Party Perspective

Registration of Third Party

- Manual and not yet standardized
- Each Data Custodian will have unique process
- Moving toward [OAuth2 Dynamic Client Registration](#)

Authorization of Data Access

- Individual Retail Customer Accounts
 - single and multiple Usage Points
- Portfolio of Accounts
- Scope Parameters

Data Exchange ←←

- Atom encapsulation
- ESPI Data Representation
- RESTful APIs



Data Exchange

Atom (RFC 4287/5023)

- Atom Overview
- ESPI integration
- Time Stamps

ESPI ([espiDerived.xsd](#))

- ESPI Resources
- OAuth2 access_tokens
- Message Patterns

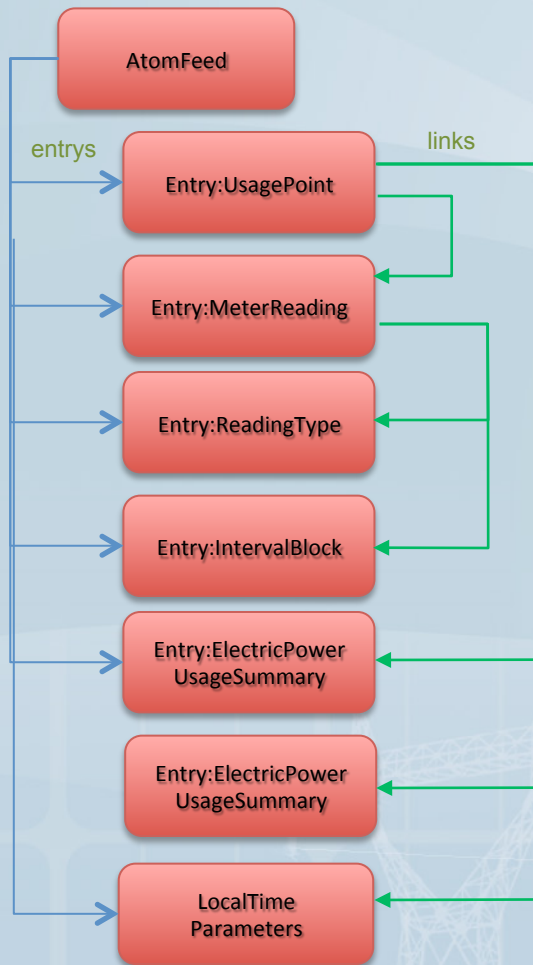
RESTful API

- Classes of API functions
- Determined by “Function Block”
- Focus today on Batch transfers

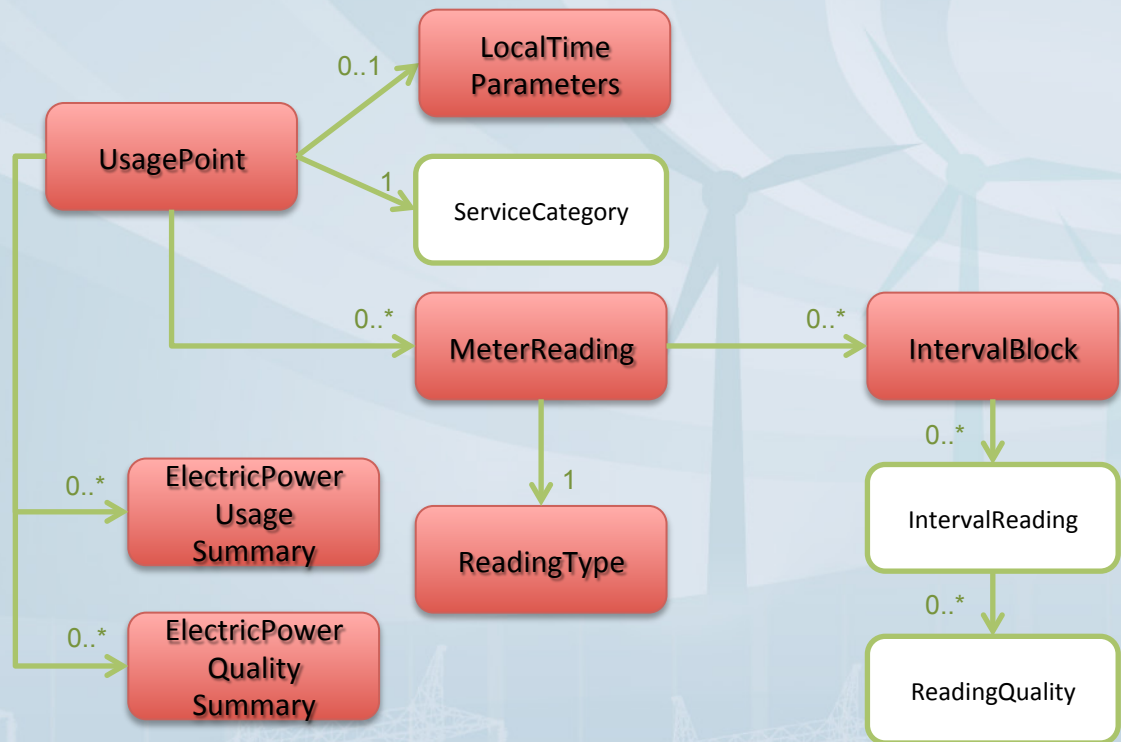


Composition and Atom Feed

Syntactic Model Atom Feed View



Information Model Profile View



Note: This information is multidimensional. Many different reading types, summaries, and readings possible. i.e. not "flat"





Notes about Time

- All Atom related times (publish-time, update-time) are of the form: 2012-10-24T00:00:00Z
- Time Localization of all times requires the LocalTimeParameters resource to be known.
- All interval and durations are in seconds in zulu time, also requiring LocalTimeParameters for interpretation
- PublishTime (at the Atom level) and interval start/durations should be consistent



Access Tokens Used

access_token: allocated by datacustodian for individual account authorizations

refresh_token: allocated at the time of authorization and used to renew an access_token

datacustodian_access_token: access token used by trusted administrative accounts

client_access_token: used by thirdparty applications to access bulk or multiple authorization subscriptions

upload_access_token: used by MeterDataManagement (backend) systems to upload/import data into a datacustodian

registration_access_token: reserved for future use in dynamic registration patterns

Note that access tokens are entirely under the control of the Data Custodian which can determine the life cycle of the access tokens and refresh tokens. The Authorized Third Party is required to react to the changes in the access token and/or refresh token as they occur.

[Ref: Authorization.docx Section 1.2.2](#)





Getting EUI from the DataCustodian

https://.../espi/1_1/resource/Batch/Subscription/1

Get a subscription (normally in response to a Notification)

https://.../espi/1_1/resource/Batch/ApplicationInformation/1

https://.../espi/1_1/resource/Batch/Authorization/1

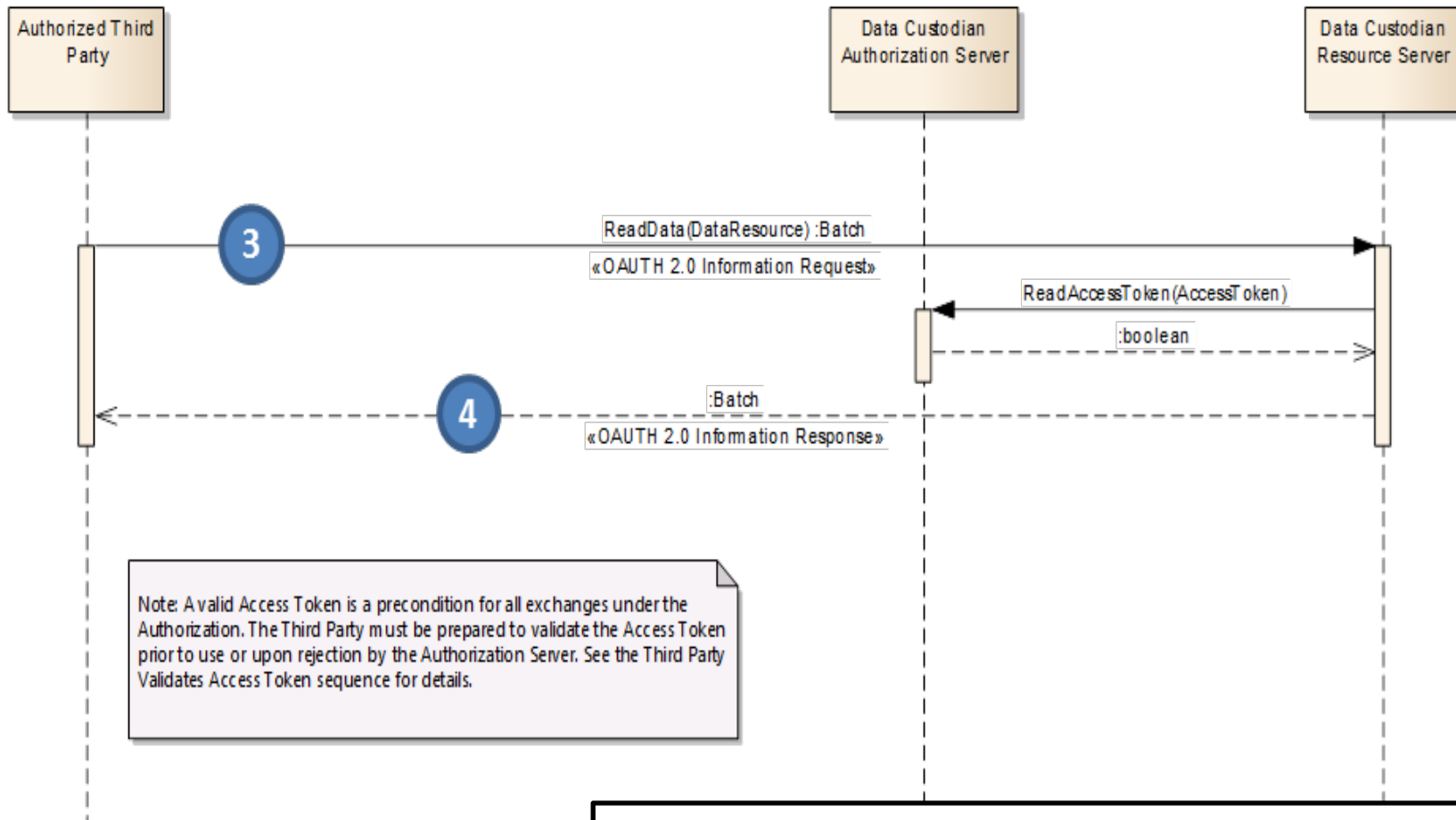
https://.../espi/1_1/resource/Batch/Bulk/1

https://.../espi/1_1/resource/Subscription/1/UsagePoint/1



Green Button CMD Data Flows

sd 11: Authorized Third Party Receives (Pulls) Requested EUI from Data Custodian - Asynchronous



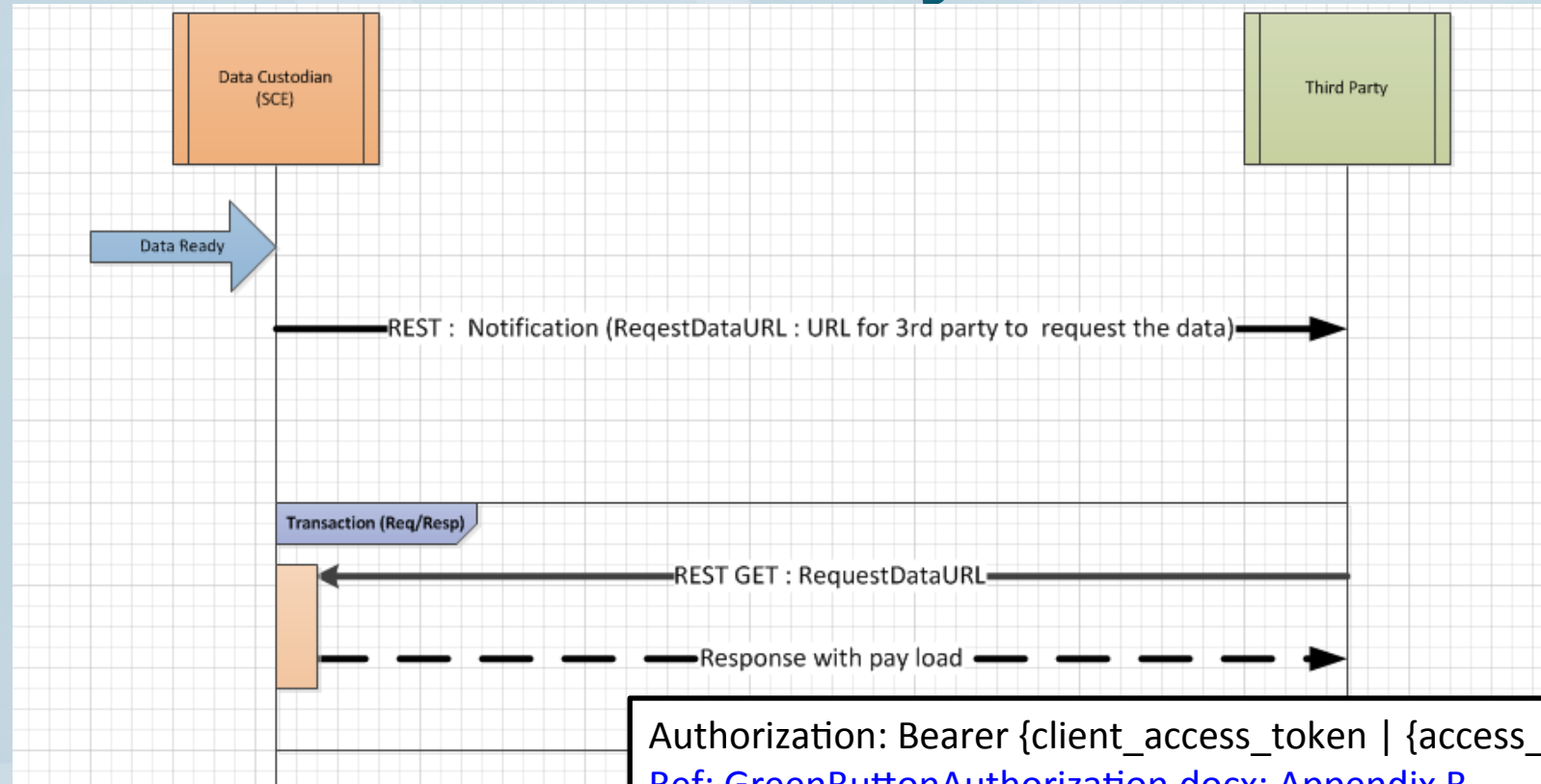
Authorization: Bearer {client_access_token | {access_token}
[Ref: GreenButtonAuthorization.docx: Appendix B](#)

[Ref: GreenButtonAuthorization.docx Section 2.2.4](#)





PUSH Model – Notify/Get



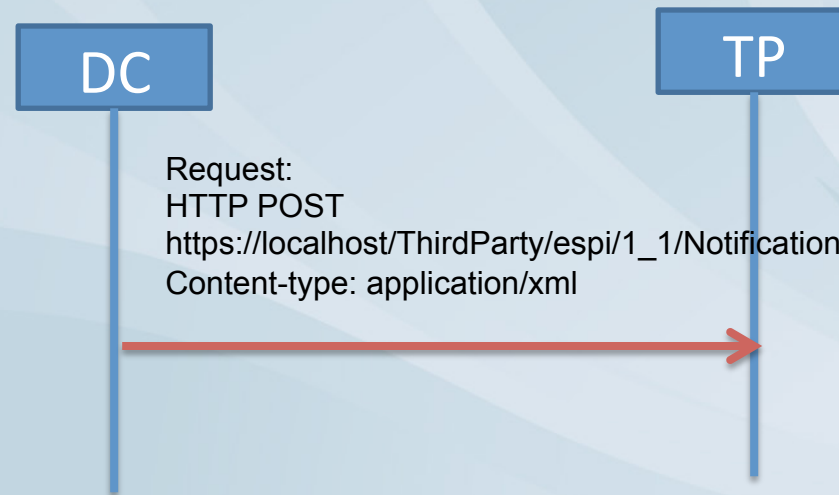
- When data is ready, data custodian will notify third party with Multiple URLs
- Third party will make call (GET) based on the URL(s) provided
- Data custodian will respond with the data as part of response to GET call

[Ref: GreenButtonCDMWorkshop: Slide 16](#)





Notification(*no access_token needed)



Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<BatchList xmlns="http://naesb.org/espi" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://naesb.org/espi espiDerived.xsd">
  <resources>https://localhost:8080/DataCustodian/espi/1_1/resource/Batch/Subscription/1
</resources>
  <resources>https://localhost:8080/DataCustodian/espi/1_1/resource/Authorization/1
</resources>
  <resources>https://localhost:8080/DataCustodian/espi/1_1/resource/Batch/Bulk/1
</resources>
</BatchList>
```

[Ref: GreenButtonCDMWorkshop: Slide 19](#)





Retrieve UsagePoints for a Subscription

```
<?xml version="1.0" encoding="UTF-8"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:espi="http://naesb.org/espi" xsi:schemaLocation="http://naesb.org/espi espiDerived.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <id>urn:uuid:1DA5F5DD-312B-4896-B910-098F59C73965</id>
  <title>Green Button Subscription Feed</title>
  <updated>1899-12-30T00:00:00Z</updated>
  <link rel="self" href="https://localhost/DataCustodian/espi/1_1/resource/Subscription/1/UsagePoint"/>
  <entry>
    <id>urn:uuid:C8C34B3A-D175-447B-BD00-176F60194DE0</id>
    <link rel="self" href="https://services.greenbuttondata.org/DataCustodian/espi/1_1/resource/Subscription/1/UsagePoint/1"/>
    <link rel="up" href="https://services.greenbuttondata.org/DataCustodian/espi/1_1/resource/Subscription/1/UsagePoint"/>
    <link rel="related" href="https://services.greenbuttondata.org/DataCustodian/espi/1_1/resource/Subscription/1/UsagePoint/1/MeterReading"/>
    <link rel="related" href="https://services.greenbuttondata.org/DataCustodian/espi/1_1/resource/Subscription/1/UsagePoint/1/ElectricPowerUsageSummary"/>
    <link rel="related" href="https://services.greenbuttondata.org/DataCustodian/espi/1_1/resource/LocalTimeParameters/01"/>
    <title>Green Button Sample Data File</title>
    <content>
      <UsagePoint xmlns="http://naesb.org/espi">
        <ServiceCategory>
          <kind>0</kind>
        </ServiceCategory>
        <ServiceDeliveryPoint>
          <name>sample tariff
          <tariffProfile>./Tariffs
        </ServiceDeliveryPoint>
      </UsagePoint>
    </content>
    <published>2013-09-19T04:00:00Z</published>
    <updated>2013-09-19T04:00:00Z</updated>
  </entry>
  <entry>
    <id>urn:uuid:C8C34B3A-D175-447B-BD00-176
    <link rel="self" href="https://services.greenbutton
    <link rel="up" href="https://services.greenbutton
    <link rel="related" href="https://services.greenb
    <link rel="related" href="https://services.greenb
    <link rel="related" href="https://services.greenb
    <title>Green Button Sample Data File</title>
    <content>
      <UsagePoint xmlns="http://naesb.org/
      <ServiceCategory>
        <kind>0</kind>
      </ServiceCategory>
      <ServiceDeliveryPoint>
        <name>sample tariff
        <tariffProfile>./Tariffs
      </ServiceDeliveryPoint>
    </UsagePoint>
    </content>
    <published>2013-09-19T04:00:00Z</published>
    <updated>2013-09-19T04:00:00Z</updated>
  </entry>
</feed>
```

DC

TP

HTTP GET
https://localhost/DataCustodian/espi/1_1/
resource/Subscription/{subscriptionID}/UsagePoint
Content-type: application/atom+xml

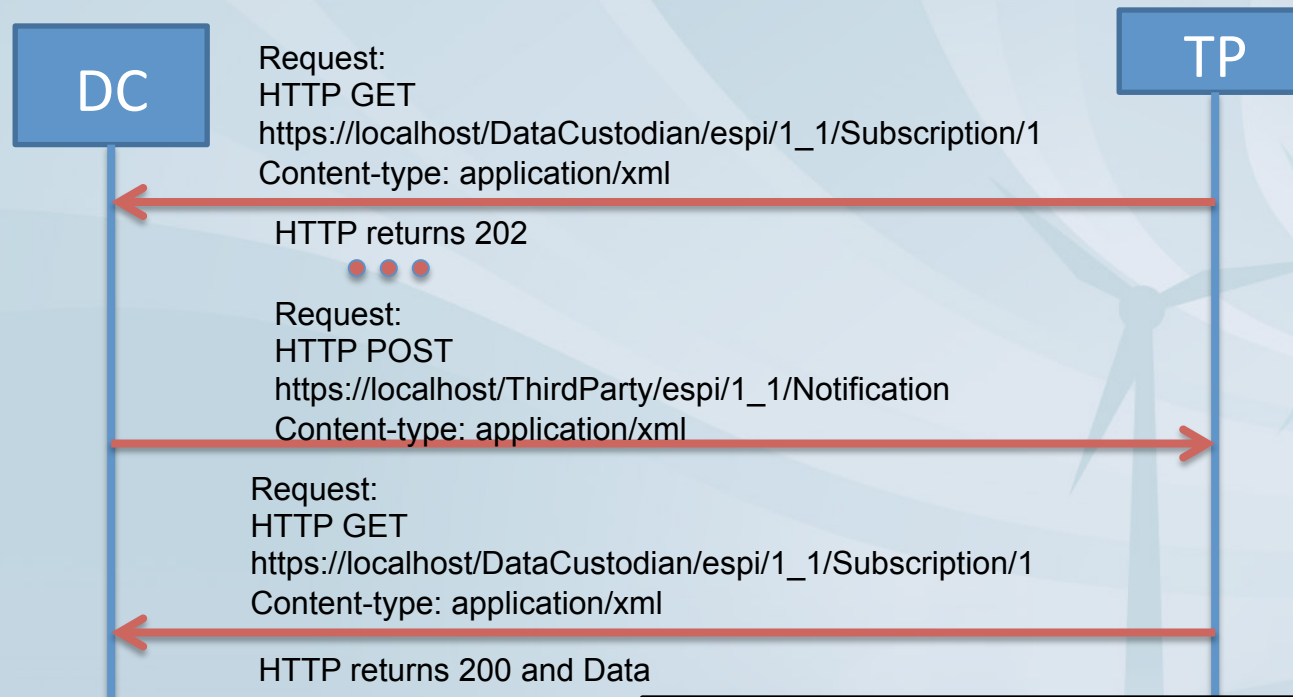
Authorization: Bearer {client_access_token | {access_token}}
[Ref: GreenButtonAuthorization.docx: Appendix B](#)

[Ref: GreenButtonCDMWorkshop: Slide 21](#)





Deferred Response when DC does not have data ready



Authorization: Bearer {client_access_token | {access_token}}
[Ref: GreenButtonAuthorization.docx: Appendix B](#)

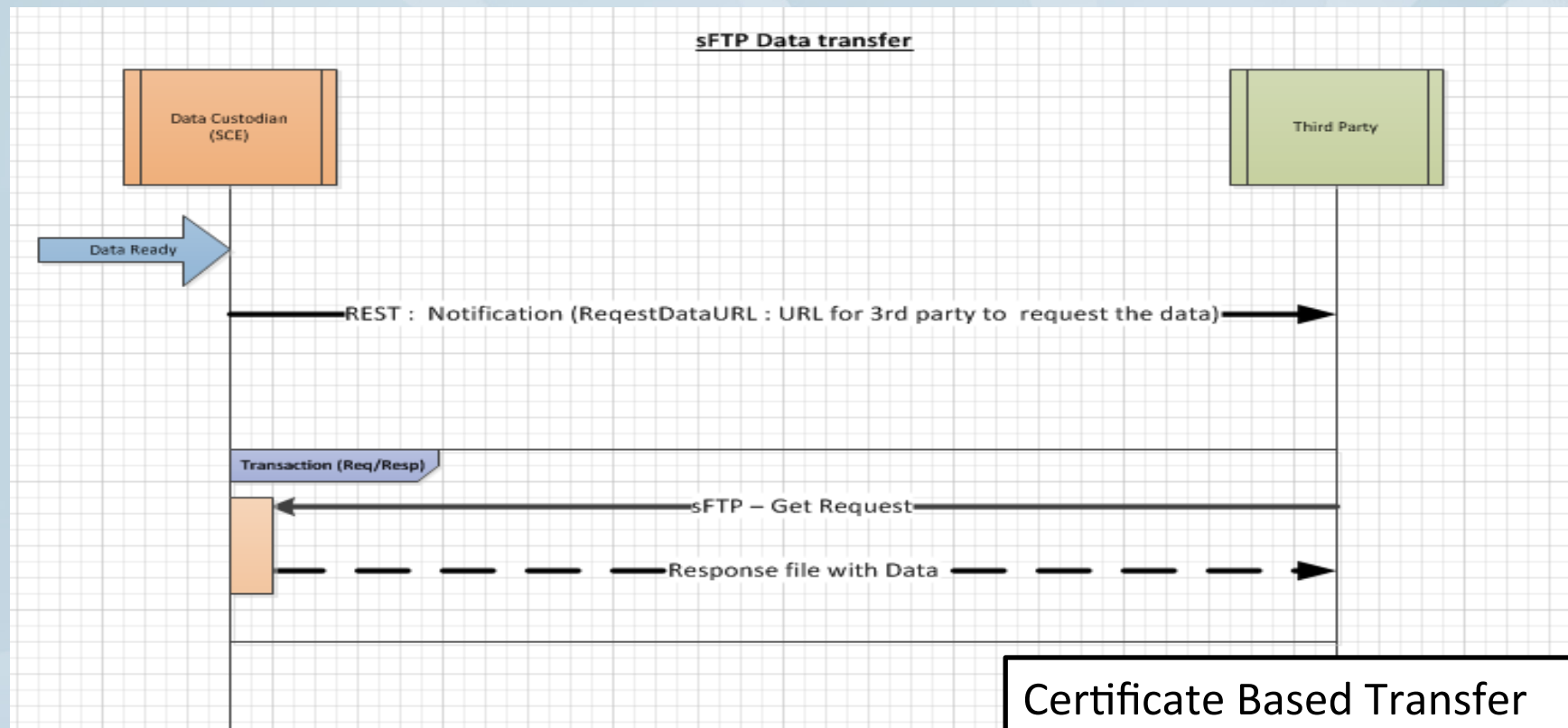
- ThirdParty makes asynchronous request
- DataCustodian does not have the data ready – returns HTTP 202
- Some time later (when ready no guarantees or time limit) DataCustodian sends Notification with original URL in BatchList
- DataCustodian provides the result

[Ref: GreenButtonCDMWorkshop: Slide 17](#)





SFTP Data Transfer



- When data is ready, data custodian will notify third party with Multiple URLs (1)
- Third party will make call (sFTP) based on the URL(s) provided
- Data custodian will respond with the data as part of response to sFTP get call

[Ref: GreenButtonCDMWorkshop: Slide 18](#)





SFTP for Bulk Transfer

- Pertinent to the SFTP discussion are the concepts that each Third Party has a defined relationship with the Data Custodian.
 - For automated exchange of information about his relationship there is a special Authorization obtained in Use Case #1
 - We anticipate that when the Data Custodian has data available, it sends an asynchronous Notification to the Third Party.
 - This Notification provides URIs of note that it is assumed the Third Party will want to retrieve.
- For the purposes of Bulk transfer, this URI will be:
 - sftp://hostname:port/DataCustodian/espi/1_1/resource/Batch/Bulk/{bulkId}
 - where {bulkId} is a unique identifier assigned by the Data Custodian and the balance of the URI is presented in the ApplicationInformation. The Third Party would then retrieve the bulk data by using an SFTP client with that URI.
- Used to Retrieve the data SFTP protocols
- Discussion –
 - After authorization of TP, they use Peney test, so what is benefit of access-token?
 - sftp user:pw, user=<tpname>, password=<tp client-credentials access-token>
- Summary
 - sftp://hostname:port/DataCustodian/espi/1_1/resource/Batch/Bulk/{bulkId}
 - sftp user:pw, user=<tpname>, password=<tp client-credentials access-token>

[Ref: Authorization.docx Section 2.3.2](#)

[Ref: GreenButtonCDMWorkshop: Slide 22](#)





Where is More Technical Information

- Base Standard
 - http://www.naesb.org/ESPI_Standards.asp (working version: [espiDerived.xsd](#))
- Developer Info
 - Download My Data - Content, Generation, and Samples
<https://collaborate.nist.gov/twiki-sggrid/bin/view/SmartGrid/GreenButtonSDK>
 - Open Source (DMD/CMD)
 - <http://energyos.github.io/OpenESPI-GreenButton-API-Documentation/>
 - <https://github.com/energyos/OpenESPI-Common-java>
 - <https://github.com/energyos/OpenESPI-DataCustodian-java>
 - <https://github.com/energyos/OpenESPI-ThirdParty-java>
 - SandBox (alpha)
 - DataCustodian: <https://services.greenbuttondata.org/DataCustodian>
 - ThirdParty: [ThirdParty https://services.greenbuttondata.org/ThirdParty](https://services.greenbuttondata.org/ThirdParty)
 - Sample Data: <https://services.greenbuttondata.org/sample-data.html>
 - API Sandbox: <http://energyos.github.io/OpenESPI-GreenButton-API-Documentation/API/>
- Testing and Certification
 - Data Validation Self-test tool - <http://www.greenbuttondata.org/greentest.aspx>
 - Green Button Test Plan -
<http://osgug.ucaiug.org/sqsystems/OpenADE/Shared%20Documents/Testing%20and%20Certification/GreenButtonTestPlan/GreenButtonTestPlan.docx>



Information for YOU!
Thank You



Definitions

- Customer
 - If residential it is individual (customer of record)
 - If commercial it is an entity
- Customer Account
 - Organization of the bill
 - A customer may have multiple customer accounts
- Service Account/Agreement
 - Corresponds Customer to a meter
- Customer Information Service Request (CISR)
 - A legal agreement that authorizes a California utility to provide customer information to a third party – paper or online
- Service Point/Location
 - Logical location representing the meter at the premises level
- ESPI
 - RetailCustomer
 - Owner of one or more UsagePoints (has right to authorize)
 - Subscription
 - Authorized access to a (sub) set of UsagePoints of a single RetailCustomer
 - Bulk
 - Authorized access to a collection of UsagePoints governed by individual authorizations to a third party





Open Source Green Button

<https://github.com/energyos>

The screenshot shows the GitHub profile page for the organization 'energyos'. The page features a header with navigation links (File, Edit, View, Favorites, Tools, Help) and a search bar. The main content area displays a list of repositories, including 'OpenESPI-DataCustodian-java', 'OpenESPI-Greenbutton-java', and 'OpenESPI-ThirdParty-java'. Each repository entry includes the repository name, the language (Java), the number of stars, and the number of forks. The left sidebar shows the organization's profile, including its name 'EnergyOS.org Master Admin energyos', a location 'Global', a website link 'http://www.energyos.org', and a join date 'Joined on Sep 27, 2011'. A green button icon is visible in the top right corner of the browser window.

Repositories Members

Find a repository... Search

All Public Private Sources Forks Mirrors

OpenESPI-DataCustodian-java
Last updated 14 hours ago
Java ★ 2 🍴 3

OpenESPI-Greenbutton-java
Last updated 20 hours ago
Java ★ 1 🍴 2

OpenESPI-ThirdParty-java
Java implementation of the ThirdParty for OpenESPI
Last updated 21 hours ago
Java ★ 1 🍴 2

EnergyOS.org Master Admin energyos
Global
<http://www.energyos.org>
Joined on Sep 27, 2011



Green Button Connect My Data Sandbox

<http://services.greenbuttondata.org/>



Green Button Sandbox

Help Login

Green Button Developer Sandbox

[Developer Guide»](#)

[Examples»](#)

[API Testbed»](#)

[DataCustodian»](#)

[ThirdParty»](#)

Welcome to the **EnergyOS Green Button Data Custodian**

EnergyOS has, with the support of Pivotal Labs, prepared a reference Green Button implementation. This implementation supports a full complement of Green Button facilities and, as an Open Source project, is freely available for download by any interested parties.

This sandbox web site gives Green Button developers a Data Custodian populated with numerous sample data files. This Green Button information is available to download and/or access directly through the EnergyOS RESTful API. Please feel free to test your Green Button skills by playing in our sandbox!

Complements of The Green Button for America Team and our Sponsors

[Learn more about Green Button »](#)

[Learn more about EnergyOS Open Source »](#)

Sponsors

Pivotal Labs

National Institute of Standards and Technology

EnerNex

PeoplePower

 Green Button for America

© EnergyOS.org 2013

www.energyos.org

NIST smart grid program

engineering laboratory





Green Button CMD Live API

Green Button API

[back to Sandbox](#)

The Green Button API provides access to RESTful services provided by Data Custodian and Third Party Applications. These APIs allow application developers full access, in Green Button Format, to Energy Usage Information for the purpose of developing new Green Button Applications.

Access to Green Button APIs require an OAuth2 Access Token. The following table contains a list of Access Tokens and the various Data Custodian Roles that may be tested using the assigned Access Token. Insert one of the following Access Tokens in the field labelled "Access Token:" at the top of the page prior to selecting the "Try It Out!" button:

Access Token	Role Being Tested
Bearer 688b026c-665f-4994-9139-6b21b13fbee	Data Custodian Admin
Bearer 809caf03-612e-4e89-94b1-6f86d83b1ef8	Meter Upload Admin
Bearer 75dd9c46-becf-48b5-9cb5-9c3233d718d0	Third Party Admin
Bearer d89bb056-0f02-4d47-9fd2-ec6a19ba8d0c	Third Party Registration Admin
Bearer 19f3087b-4717-4805-b9d5-98bf50212ed6	Role User (alan -- ID=1)
Bearer ed493189-4023-4196-81b3-154f5b72d377	Role User (charles -- ID=5)

Access Token:

DataCustodian

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

DataCustodian.ApplicationInformation

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

GET	/espi/1_1/resource/ApplicationInformation	Retrieve all Application Information Structure contents [FB_33]
POST	/espi/1_1/resource/ApplicationInformation	Add a new Application Information Structure [FB_33]
GET	/espi/1_1/resource/ApplicationInformation/{ApplicationInformationID}	Retrieve an Application Information Structure using it's ID [FB_3]
PUT	/espi/1_1/resource/ApplicationInformation/{ApplicationInformationID}	Update an existing Application Information Structure [FB_41]
DELETE	/espi/1_1/resource/ApplicationInformation/{ApplicationInformationID}	Delete an Application Information Structure [FB_41]

DataCustodian.Authorization

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)