Web Services Overview

Reader please note!!!

When it comes to eBusiness transactions and data transmissions, it is critical that trading partners know that they are identifying their various entities and products in these transactional messages (such as Buyer, Seller, ShipTo, BillTo, Carrier, etc.) correctly. In the Ag industry we have determined we will use a set of common identifiers instead of sending all the demographics for the entity locations and data information on the products. The Ag industry determined to house these common identifiers in a central industry database that we refer to as the Ag Industry Identification System (AGIIS).   AGIIS is a database and software that delivers a common set of identifier s and data elements to the Agricultural industry. AGIIS contains unique identifiers for entities, products and licenses that are used in eBusiness transactions such as identifying where a product was shipped in an electronic ship notice transaction.

One of the methods supported in AGIIS for accessing these unique identifiers is Web Services. Web Services allows companies to access AGIIS seamlessly through computer to computer interaction securely and in real time. There are several web services available within AGIIS that will allow companies to streamline their workflow processes; which can result in reduction or elimination of manual interaction with AGIIS. Below is a description of available web services and purpose of use.

In January 2011, the AGIIS Directory Oversight Committee (DOC) approved a project (aka PCR173) to rewrite AGIIS web services to conform to the AgGateway generic web service specification. This project provided continuity with the "old" web services from a functional standpoint and the improved interface architecture eliminated issues that had prevented some users from taking advantage of the services previously. The highlights of the "new" interface architecture include the following:

- o Conformance with the AgGateway Web Services Guideline 2010-11-28
- o Elimination of the AGIIS client side cookie
- o Use of WS-Security Username Token Profile in the Soap Header for user and password authentication
- o Use of independent XML schema definitions for all input and output data payloads

The remainder of this document applies only to the current web service architecture are services, i.e. the "post-PCR173" web services. For information on the "old" web service architecture and the services available therein, please refer to document https://www.agiis.org/Tutorials/WebServicesOverview.pdf.

Current AGIIS Web Service Architecture Overview

- All web services are exposed through the end point AGIISWebServices.svc
  - o Prod URL: https://www.agiis.org/Services/AGIISWebServices.svc
  - o Test URL: https://agiistest.covansysec.com/Services/AGIISWebServices.svc
- Two functions are exposed through this interface:
  - o HelloWorld – This method is meant only as a developer's tool. It does not require the AgGateway Generic Specification for input\output definition (e.g businessProcess, xmlPayload, etc.) but does require "the remainder of the

framework" including SSL transport layer security and user authentication via the WS-Security Username Token Profile in the Soap Header. When called successfully a string with the following content is returned:

- Hello **userid** You have successfully called the AGIIS web service HelloWorld.

o AGIISWebServices – This is the method that will be used for all "real" AGIIS web service calls and requires the entire AgGateway Generic Specification for input\output definition, SSL transport layer security, and user authentication via the WS-Security Username Token Profile in the Soap Header. The AgGateway specification is based on the premise of using a common XML schema for all web service calls.

o For the purposes of calling AGIIS web services, an understanding of the following 3 elements in the common XML schema is critical:

- inboundData.businessProcess
- inboundData.xmlPayload
- outboundData.xmlPayload

o inboundData.businessProcess

- Individual web service "business functions" are requested using enumerated values for the element **businessProcess**. The businessProcess values supported in AGIIS are the following:
- <xsd:enumeration value="EntitySearch"/>
- <xsd:enumeration value="EntityMaintenance"/>
- <xsd:enumeration value="ProductSearch"/>
- <xsd:enumeration value="ProductMaintenance"/>
- <xsd:enumeration value="ProductSubsetMaintenance"/>
- <xsd:enumeration value="HelloAGIIS"/>

o inboundData.xmlPayload

- The input data structure appropriate for any particular businessProcess is defined in a separate XML structure and populated in the xmlPayload element. In the AGIIS implementation all inboundData.xmlPayload definitions are provided in the standard AGIIS XML Schema (agiisschema.xsd). Subscribers that are already familiar with the EntityInformation and ProductInformation structures in the AGIIS XML Schema will have a "leg up" because the standard AGIIS XML structures are also used in the web services. The AGIIS XML Schema has elements for each of the businessProcess enumerations. They are the following:
  - EntityMaintenanceInboundXMLPayload
  - EntitySearchInboundXMLPayload
  - ProductSearchInboundXMLPayload
  - ProductMaintenanceInboundXMLPayload
  - ProductSubsetMaintenanceInboundXMLPayload
  - HelloAGIISInboundXMLPayload

o outboundData.xmlPayload

- The output data structure appropriate for any particular businessProcess is defined in a separate XML structure and populated in the xmlPayload element. In the AGIIS implementation all inboundData.xmlPayload definitions are provided in the standard AGIIS XML Schema (agiisschema.xsd). The AGIIS XML Schema has elements for each of the businessProcess enumerations. They are the following:
  - EntityMaintenanceOutboundXMLPayload
  - EntitySearchOutboundXMLPayload

- ProductSearchOutboundXMLPayload
- ProductMaintenanceOutboundXMLPayload
- ProductSubsetMaintenanceOutboundXMLPayload
- HelloAGIISOutboundXMLPayload

Helpful Hints by businessProcess

HelloAGIIS

The HelloAGIIS businessProcess is only intended as a programmer's development aid, i.e. to connect the plumbing without any AGIIS application knowledge. The framework of this service is the same as the standard functional services, but the input and output payloads are very simple. This framework includes three primary facets:

- Input and output as defined via the interface .wsdl (and .xsd used to generate that .wsdl file)
- Transport layer security via HTTPS
- User authentication via the WS-Security Username Token Profile value in the soap header. The userid and password provided are authenticated against the AGIIS user database.

This service echoes the input message back to the caller with the response:

"Welcome to AGIIS web services. Your input message said: " + *your input message*

EntitySearch

The EntitySearch businessProcess replaces the following services previously provided by AGIIS. Please note the attribute EntitySearchInboundXMLPayload@ActionRequest for each (more on that later)

Old Service - Method	EntitySearchInboundXMLPayload@ActionRequest
Search.asmx - IsAGIISSubscriber	Search By Entity\Subscriber Identifier
Search.asmx - EntitybyPropCode	Search By Entity Proprietary Code
Search.asmx - EntitybyIndustryID	Search By Entity Identifier
Search.asmx - EntitybyNamePostal	Search By Entity Attribute
Search.asmx - AdvancedEntitySearch	Search By Entity Attribute
Search.asmx - AdvancedEntitySearchByAddress	Search By Entity Attribute
Product.asmx - IsEntityLicensedToPurchaseThisProduct	Search By Entity\Product\License Identifier

All permutations of *EntitySearch* supply search criteria in the element EntitySearchInboundXMLPayload. The following notes pertain to populating that element:

Search By Entity|Subscriber Identifier

- EntitySearchInboundXMLPayload@ActionRequest="Search By Entity\Subscriber Identifier"

- At least one EntitySearchCriteria/RequiredEnitityIdentifierTypes must be designated, e.g. if the returned entity is required to have a GLN identifier, then populate as in the following example:
  - \<RequiredEnitityIdentifierTypes Agency="GLN">
  - \<RequiredEnitityIdentifierType>true\</RequiredEnitityIdentifierType>
  - \</RequiredEnitityIdentifierTypes>
  - Note that if this criteria is not important, include all Agency values in your search criteria population (GLN, EBID, NAPD/HP ID), i.e. as long as one of the "required" identifier types exists for the entity, it is included in the result set as long as all other search criteria specified is met.
- Populate the Subscriber Entity Identifier value search criteria as in the following example:
  - \<IndustryIdentifier Agency="EBID">
  - \<IndustryCode>9435074180000\</IndustryCode>
  - \</IndustryIdentifier>
- No other EntitySearchCriteria elements should be populated

Search By Entity Proprietary Code

- EntitySearchInboundXMLPayload@ActionRequest="Search By Entity Proprietary Code"
- At least one EntitySearchCriteria/RequiredEnitityIdentifierTypes must be designated, e.g. if the returned entity is required to have a GLN identifier, then populate as in the following example:
  - \<RequiredEnitityIdentifierTypes Agency="GLN">
  - \<RequiredEnitityIdentifierType>true\</RequiredEnitityIdentifierType>
  - \</RequiredEnitityIdentifierTypes>
  - Note that if this criteria is not important, include all Agency values in your search criteria population (GLN, EBID, NAPD/HP ID), i.e. as long as one of the "required" identifier types exists for the entity, it is included in the result set as long as all other search criteria specified is met.
- Populate the proprietary code search criteria as in the following example:
  - \<IndustryIdentifier Agency="EBID">
  - \<IndustryCode>\</IndustryCode>
  - \<ProprietaryCode>151623\</ProprietaryCode>
  - \</IndustryIdentifier>
- No other EntitySearchCriteria elements should be populated

Search By Entity Identifier

- EntitySearchInboundXMLPayload@ActionRequest="Search By Entity Identifier"
- At least one EntitySearchCriteria/RequiredEnitityIdentifierTypes must be designated, e.g. if the returned entity is required to have a GLN identifier, then populate as in the following example:
  - \<RequiredEnitityIdentifierTypes Agency="GLN">
  - \<RequiredEnitityIdentifierType>true\</RequiredEnitityIdentifierType>
  - \</RequiredEnitityIdentifierTypes>
  - Note that if this criteria is not important, include all Agency values in your search criteria population (GLN, EBID, NAPD/HP ID), i.e. as long as one of the "required" identifier types exists for the entity, it is included in the result set as long as all other search criteria specified is met.
- Populate the Identifier value search criteria as in the following example:

- o `<IndustryIdentifier Agency="EBID">`
- o   `<IndustryCode>9435074180000</IndustryCode>`
- o   `</IndustryIdentifier>`
- No other EntitySearchCriteria elements should be populated

Search By Entity Attribute

- EntitySearchInboundXMLPayload@ActionRequest="Search By Entity Attribute"
- This search is used for the majority of all "demographic" searches, e.g. by name, by address, by phone, etc.
- At least one EntitySearchCriteria/RequiredEnitityIdentifierTypes must be designated, e.g. if the returned entity is required to have a GLN identifier, then populate as in the following example:
  - o `<RequiredEnitityIdentifierTypes Agency="GLN">`
  - o   `<RequiredEnitityIdentifierType>true</RequiredEnitityIdentifierType>`
  - o `</RequiredEnitityIdentifierTypes>`
  - o Note that if this criteria is not important, include all Agency values in your search criteria population (GLN, EBID, NAPD/HP ID), i.e. as long as one of the "required" identifier types exists for the entity, it is included in the result set as long as all other search criteria specified is met.
- The search criteria is populated into the appropriate EntitySearchCriteria element.
- For performance reasons, one of the following "primary search criteria" **must** be populated[1]:
  - o GenericEnititySearchName/EntitySearchName
    - ▪ Note that a name search also requires one of the following geographic search criteria
      - Postal Code
      - Postal Code Range
      - City and State
      - Note that Address Line 1 can be optionally included with the above 3 geographic criteria for a more refined search
    - ▪ Wildcard search is supported (use the * for the wildcard) in the primary search criteria EntitySearchName value; maximum length = 128 characters
    - ▪ Entity search name has a 3 character minimum length requirement
    - ▪ City, if populated as search criteria, has a 3 character minimum length requirement
  - o GenericEnititySearchAddress/SearchAddressInformation/AddressLine1
    - ▪ Note that an address line 1 search also requires one of the following search criteria
      - Postal Code
      - Postal Code Range
      - City and State
    - ▪ Wildcard search is supported (use the * for the wildcard) in the primary search criteria AddressLine1 and City values; AddressLine1 maximum length = 128 characters; City maximum

---

[1] Adherence to the "primary search criteria" rules can be "prototyped" by using the Entity Search feature on the AGIIS web.

length = 30 characters; City, if populated as search criteria, has a 3 character minimum length requirement
- Address line 1 has a 3 character minimum length requirement
- GenericEnititySearchAddress/SearchAddressInformation/PostalCode or PostalCodeRange
  - Note that a postal code or postal code range search that specifies RequiredEntityIdentifierTypes "NAPD/HP ID" also requires one of the following name search criteria
    - Company Name
    - Last Name
  - Postal code range values must each be a minimum of 5 characters and a maximum of 9 characters
  - Single postal code searches must be a minimum of 3 and a maximum of 9 characters
- TelephoneNumber (requires no other primary criteria; must be 10 digits)

- Note that as long as the previous "primary search criteria" requirement is satisfied, "secondary criteria[2]" can be also be used for searching including the following:
  - PreferredName
  - DBAName
  - IndividualName
  - EntityType
    - As of PCR185, 07/01/2013, this field is no longer supported
  - EntityStatus
  - GenericEntitySearchAddress components other than AddressLine1
  - LastModifiedDateTime
  - Latitude
  - Longitude
  - SPLC
  - Entity Classifications
- The following elements are reserved for future use and should not be used:
  - PhysicalAddress (to search on address components, populate the GenericEntitySearchAddress element)
  - MailingAddress (to search on address components, populate the GenericEntitySearchAddress element)
- IndustryIdentifier should not be populated

Search By Entity|Product|License Identifier

- EntitySearchInboundXMLPayload@ActionRequest="Search By Entity\Product\License Identifier"
- At least one EntitySearchCriteria/RequiredEntityIdentifierTypes must be designated, e.g. if the returned entity is required to have a GLN identifier, then populate as in the following example:
  - <RequiredEnitityIdentifierTypes Agency="GLN">
  - <RequiredEnitityIdentifierType>true</RequiredEnitityIdentifierType>
  - </RequiredEnitityIdentifierTypes>
  - Note that if this criteria is not important, include all Agency values in your search criteria population (GLN, EBID, NAPD/HP ID), i.e. as long as

---

[2] There is no support for wildcard searches on secondary criteria. Matches are limited to an exact match of the secondary criteria specified.

one of the "required" identifier types exists for the entity, it is included in the result set as long as all other search criteria specified is met.

- Populate the Entity Identifier value search criteria as in the following example:
  - <IndustryIdentifier Agency="EBID">
  - <IndustryCode>9435074180000</IndustryCode>
  - </IndustryIdentifier>
- No other EntitySearchCriteria elements should be populated
- One of the following "product" search criteria should be populated:
  - ProductOwnerIdentifier *and* ProductIdentifier
  - ProductLicensingSearchCriteria/GTIN
  - ProductLicensingSearchCriteria/ReportingIdentifier

EntityMaintenance

"Entity Maintenance" is modeled after the GLN Bulk transaction and uses the same XML EntityInquiry structure. See the GLN Bulk XML usage document available on the web for further information on element population. Note that only a single transaction is permitted per web service call. Currently, the following GLN Bulk "transaction types" are supported via the web service:

- EntityInquiry@ActionRequest="Add EBID"
  - This transaction, when successful[3], adds a new entity to the AGIIS entity database and the requestor's EBID subset. Note that all required fields must be supplied in the request, no defaults are provided. Note that in addition to the requested EBID Identifier, an AGIIS GLN will also be associated with the new entity. Note that "Add EBID" transactions with valid data but which require review by AGIIS Member Services according to AGIIS business rules will be queued for manual review. Such review will result in their acceptance or rejection based on directory policy regarding duplicates, etc.
  - EntityInformation.IndustryIdentifier
    - EntityInformation.IndustryIdentifier@Agency="EBID"
    - EntityInformation.IndustryIdentifier.IndustryCode
      - This is a required element in the schema but should not be valued in the request XML as AGIIS will assign a new EBID in a successful transaction. Use <IndustryCode></IndustryCode>.
    - EntityInformation.IndustryIdentifier.ProprietaryCode
      - The requestor should supply a proprietary code for the EBID Subset entry or use the constant "USE_ID" in order to have AGIIS populate the proprietary code with the newly assigned EBID value.
    - EntityInformation.IndustryIdentifier.ProprietaryCodes
      - This repeating element can be used instead of .ProprietaryCode to set multiple proprietary code values for the entity subset. Up to 99 value can be set per entity subset. The constant "USE_ID" can be used to direct AGIIS to use the entity identifier value as one of the proprietary code values.

---

[3] Successful add entity transactions require that all business rule edits are passed, the "new entity" is not a duplicate of an existing entity, and the request does not require manual review by AGIIS Member Services.

- EntityInquiry@ActionRequest="Add GLN"
  - This transaction, when successful, adds a new entity to the AGIIS entity database and the requestor's GLN subset. Note that all required fields must be supplied in the request, no defaults are provided. Note that "Add GLN" transactions with valid data but which require review by AGIIS Member Services according to AGIIS business rules will be queued for manual review. Such review will result in their acceptance or rejection based on directory policy regarding duplicates, etc.
  - EntityInformation.IndustryIdentifier
    - EntityInformation.IndustryIdentifier@Agency="GLN"
    - EntityInformation.IndustryIdentifier.IndustryCode
      - In order to have AGIIS assign a GLN from the pool of available "AGIIS GLN's", use the constant AGIIS_GLN.
      - If the GLN of the entity to be added to AGIIS has already been established by an outside agency (e.g. this is an owned location of a subscriber that assigns their own GLN's), the request should be populated with the pre-existing GLN value.
    - EntityInformation.IndustryIdentifier.ProprietaryCode
      - The requestor should supply a proprietary code for the GLN Subset entry or use the constant "USE_ID" in order to have AGIIS populate the proprietary code with the newly assigned GLN value.
    - EntityInformation.IndustryIdentifier.ProprietaryCodes
      - This repeating element can be used instead of .ProprietaryCode to set multiple proprietary code values for the entity subset. Up to 99 value can be set per entity subset. The constant "USE_ID" can be used to direct AGIIS to use the entity identifier value as one of the proprietary code values.
- EntityInquiry@ActionRequest="Update"
  - This transaction, when successful, updates the existing AGIIS entity with those data elements supplied in the request. The requestor specifies the identifier (GLN or EBID) of the "target entity" to be updated. All elements that are present in the request EntityInformation XML will be applied to the updated entity. Therefore, the request should either call the AGIIS "Search By Entity Identifier" web service immediately before the update (in order to retrieve the current AGIIS values for the fields not present in the requestor's database) OR should omit the elements in the update request that are not present in the requestor's database. Note that the entity must be in the requestor's entity subset in order to submit an update transaction. Note that "Update" transactions with valid data but which require review by AGIIS Member Services according to AGIIS business rules will be queued for manual review. Such review will result in their acceptance or rejection based on directory policy regarding duplicates, etc.
    - EntityInformation.IndustryIdentifier
      - This element must be populated with the unique identifier of the "target entity" to be updated.
      - EntityInformation.IndustryIdentifier@Agency
        - The Agency attribute must match the IndustryCode, e.g. when the IndustryCode value is a GLN,

EntityInformation.IndustryIdentifier@Agency="GLN"
- o The entity must be in the requesting subscriber's "Agency" entity subset or the transaction will be rejected, e.g. when Agency="GLN", the entity must be in the GLN Subset.
- EntityInformation.IndustryIdentifier.IndustryCode
  - o The IndustryCode uniquely identifies the "target entity" to be updated. It should contain a valid AGIIS entity identifier, e.g. an EBID or a GLN.
- EntityInformation.IndustryIdentifier.ProprietaryCode
  - o This element should be omitted and if not omitted is ignored.
  - ▪ EntityInformation.EntityStatus
    - The EntityStatus is not allowed to be modified in the update transaction. If supplied, it must match the current value in AGIIS. Only entities with EntityStatus="In Business" are allowed to be updated.
  - ▪ EntityInformation.EntityType
    - The EntityType is not allowed to be modified in the update transaction. If supplied, it must match the current value in AGIIS. As of PCR185, 07/01/2013, this field is no longer supported.
  - ▪ EntityInformation@ActiveFlag
    - The EntityInformation@ActiveFlag is not allowed to be modified in the update transaction. If supplied, it must match the current value in AGIIS. Only entities with EntityInformation@ActiveFlag="Active" are allowed to be updated.
  - ▪ EntityInformation.EntityClassifications
    - The EntityClassification(s) are not allowed to be modified in the update transaction. If supplied, it must match the current value in AGIIS.
- EntityInquiry@ActionRequest="Add to Subset"
  - o This transaction, when successful, adds a new entry in the requestor's entity subset or updates the proprietary code of an existing entity subset entry. The requestor specifies the identifier (GLN or EBID) of the "target entity" for which the subset entry is to be added\updated. The "target subset" is determined by the identifier type in the request.
    - ▪ EntityInformation.IndustryIdentifier
      - This element must be populated with the unique identifier of the "target entity" for the subset to be modified.
      - EntityInformation.IndustryIdentifier@Agency
        - o The Agency attribute must match the IndustryCode, e.g. when the IndustryCode value is a GLN, IndustryIdentifier@Agency="GLN"
        - o The Agency will determine the entity subset to be modified, i.e. when Agency="GLN", the GLN subset will be modified.
      - EntityInformation.IndustryIdentifier.IndustryCode
        - o The IndustryCode uniquely identifies the "target entity" for the subset to be modified. It should

contain a valid AGIIS entity identifier, e.g. an EBID or a GLN.

- EntityInformation.IndustryIdentifier.ProprietaryCode
  - This element should contain the "new value" of the subset entry. The constant "USE_ID" can be used to direct AGIIS to use the entity identifier value as the proprietary code value. This can be particularly useful when requesting EBID enumeration.
- EntityInformation.IndustryIdentifier.ProprietaryCodes
  - This repeating element can be used instead of .ProprietaryCode to set multiple proprietary code values for the entity subset. Up to 99 value can be set per entity subset. The constant "USE_ID" can be used to direct AGIIS to use the entity identifier value as one of the proprietary code values.

- Requesting EBID Enumeration
  - An active entity that does not currently have an EBID can be enumerated with a "GLN based EBID" using the "Add to Subset" web service. This is accomplished by populating 2 iterations of EntityInformation.IndustryIdentifier. The first must specify the entity's GLN (using Agency="GLN"). The second must specify Agency="EBID" and the constant value "Add EBID" must be populated in the IndustryCode. The proprietary code should reflect the unique identifier from the caller's legacy system or whatever unique value the caller desires. In order to have AGIIS populate the proprietary code with the AGIIS EBID identifier value, use the constant "USE_ID" for the proprietary code value.

- EntityInquiry@ActionRequest="Delete from Subset"
  - This transaction, when successful, removes an entry from the requestor's entity subset. The requestor specifies the identifier (GLN or EBID) of the "target entity" for which the subset entry is to be removed. The "target subset" is determined by the identifier type in the request.
    - EntityInformation.IndustryIdentifier
      - This element must be populated with the unique identifier of the "target entity" for the subset to be deleted.
      - EntityInformation.IndustryIdentifier@Agency
        - The Agency attribute must match the IndustryCode, e.g. when the IndustryCode value is a GLN, IndustryIdentifier@Agency="GLN"
        - The Agency will determine the entity subset to be removed, i.e. when Agency="GLN", the GLN subset will be removed.
      - EntityInformation.IndustryIdentifier.IndustryCode
        - The IndustryCode uniquely identifies the "target entity" for the subset to be removed. It should contain a valid AGIIS entity identifier, e.g. an EBID or a GLN.

- EntityInformation.IndustryIdentifier.ProprietaryCode
    - This element should be omitted and if not omitted is ignored.
- EntityInquiry@ActionRequest="Report Duplicates"
    - This transaction, when successful, submits a set of active entities, suspected to be duplicates, to AGIIS Member Services for manual review. When Member Services has completed their review, the result of the review will be communicated by email to the user who has been designated as the "primary contact" for the subscriber who submitted the request. The "set of duplicate entities" must be identified by GLN using the standard "IndustryIdentifer" element.
        - EntityInquiry.EntitiyInformation
            - This element is not used in this transaction
        - EntityInquiry.DuplicateEntities.DuplicateIdentifier.IndustryIdentifier
            - An array of up to 20 GLN's are submitted for review using the IndustryIdentifer.IndustryCode element.

ProductSearch

All permutations of *ProductSearch* supply search criteria in the element ProductSearchInboundXMLPayload. The format of the output data can be requested in two different formats by specifying the PreferredProductOutputView (Flattened Product GTIN or Traditional Hierarchy which is the default if not specified). The following notes pertain to populating the ProductSearchInboundXMLPayload element:

Search By Product Identifier

- ProductSearchInboundXMLPayload@ActionRequest="Search By Product Identifier"
- Searching by GTIN requires only the GTIN element
    - ProductSearchCriteria/PackageConfigurationSearchInformation/GTIN. The 14 digit GTIN uniquely identifies the package configuration for the search. The GTIN can be supplied "with or without hyphens", e.g. 0-003583253256-6 or 00035832532566.
- Searching by Product Identifier or Reporting Identifier require both that value and an identifier of the company that owns the product.
    - The product owning company is identified by ProductSearchCriteria/CompanySearchInformation/PartnerIdentifier
    - The Product Identifier or Reporting Identifier is identified by one of the following:
        - Product Identifier
            - ProductSearchCriteria/ProductSearchInformation/ProductIdentification
            - Maximum length = 30 characters
        - Reporting Identifier
            - ProductSearchCriteria/PackageConfigurationSearchInformation/ReportingIdentifier
            - Maximum length = 30 characters
- No other ProductSearchCriteria elements should be populated

Search By Product Attribute

- ProductSearchInboundXMLPayload@ActionRequest="Search By Product Attribute"
- Subscriber Name and Product Name are required and at least 3 characters of each must be supplied. These values must be supplied in the following elements:
    - Subscriber Name can be supplied by either supplying a name or subscriber identifier
        - ProductSearchCriteria/CompanySearchInformation/PartnerName
            - Wildcard search is supported (use the * for the wildcard); maximum length = 128 characters, Subscriber Name, if populated as search criteria, has a 3 character minimum length requirement.
        - ProductSearchCriteria/CompanySearchInformation/PartnerIdentifier
    - Product Name is supplied in
        - ProductSearchCriteria/ProductSearchInformation/ProductIdentification/ProductName
        - Wildcard search is supported (use the * for the wildcard); maximum length = 128 characters, Product Name, if populated

as search criteria, has a 3 character minimum length requirement.

- The following criteria are optional:
  - Brand Name
    - ProductSearchCriteria/ProductSearchInformation/BrandName
    - Wildcard search is supported (use the * for the wildcard); maximum length = 50 characters
  - Variety
    - ProductSearchCriteria/ProductSearchInformation/ProductVariety
    - Wildcard search is supported (use the * for the wildcard); maximum length = 128 characters
  - Maturity Days Minimum
    - ProductSearchCriteria/ProductSearchInformation/ProductMaturityDaysMinimum; Maturity Days Minimum, if populated as search criteria, must be a numeric value between 1 and 999.
  - Maturity Days Maximum
    - ProductSearchCriteria/ProductSearchInformation/ProductMaturityDaysMaximum; Maturity Days Maximum, if populated as search criteria, must be a numeric value between 1 and 999
  - Seed Size
    - ProductSearchCriteria/PackageSizeSearchInformation/SeedSize
    - Wildcard search is supported (use the * for the wildcard); maximum length = 128 characters
- No other ProductSearchCriteria elements should be populated

Search By Product Proprietary Code

- ProductSearchInboundXMLPayload@ActionRequest="Search By Product Proprietary Code"
- Proprietary Code is required
  - ProductSearchCriteria/PackageConfigurationSearchInformation/ProprietaryCode
  - Maximum length = 30 characters
- Note that the subscriber for which the product subset search pertains is derived from the userid credential. The userid is associated with a subscriber, and that subscriber is used in the product subset search by proprietary code.
- No other ProductSearchCriteria elements should be populated

ProductMaintenance

"Product Maintenance" is modeled after XML batch product imports and uses the same XML structure. The input can be specified using either the Flattened Product GTIN (ProductGTINInformation element) or Traditional Hierarchy (CompanyInformation element) XML structures. Output is provided using the same Flat or Traditional format specified in the inbound payload. Add, Delete, Replace, and AddOrReplace transactions are supported. See the Product Import XML usage document available on the web for further information on element population. Note that only a single transaction is permitted per web service call.

ProductSubsetMaintenance

"Product Subset Maintenance" is modeled after the XML batch product subset import and uses the same XML structure. The input is specified using the GTINCrossReference XML structure. The "target" product package configuration GTIN must be specified. Add, Delete, Replace, and AddOrReplace transactions are supported. See the Product Subset Import XML usage document available on the web for further information on element population. Note that only a single transaction is permitted per web service call.

Converting Old Web Service Calls to the New, Post PCR173 Web Service Calls

The following document, **https://www.agiis.org/Tutorials/PCR173WebServiceXref.pdf**, provides a cross reference of old services and methods to their new counterparts. Additionally, sample web service client code is provided (Microsoft .Net only) which demonstrates calling the new web services. The following link downloads this sample code **https://www.agiis.org/Help/Samples/PCR173WebServices.zip**. Examples are provided "as is". In addition to the sample code the zip file contains a number of sample .xml instance file that demonstrate common inboundData.xmlPayload examples, e.g. searching for an entity by name and address, searching for an entity by identifier, etc.

Example Request and Response SOAP Envelopes

The following documents (generated using SoapUI) provide examples of the request and response SOAP envelopes**:**
https://www.agiis.org/Help/Samples/SoapUISampleRequestSoapEnvelope.xml
https://www.agiis.org/Help/Samples/SoapUISampleResponseSoapEnvelope.xml