Affordable Care Act (ACA) Information Returns (AIR)

Working Group Meeting

November 15, 2016

Affordable Care Act Information Returns (AIR) Program

Today's Topics

- Adding a Software Developer Package

- Changes to the AIR Submission Composition and Reference Guide

- Demonstration of Manifest and Crosswalk

- Demonstration of Form 1095-C Schema and Crosswalk

- Q & A

  - October Questions and Response Appendix

- General AIR Mailbox and e-Help Desk Information

Adding a Software Developer Package

Software Developer Actions for PY 2017

- Software Developers, who passed AATS for Tax Year 2015, do not have to retest for Tax Year 2016; however, you do need to get your TY 2016 Software IDs.

- Software Package information must be updated annually online through the *ACA Application for TCC.*

- New Software IDs will be assigned for each tax year. To update your application, the Responsible Official should go to the Application Details page and click the "Add" button under the Software Developer Package List.

Note: New participants will need to comply with software test requirements for Tax Year 2016. Also, if you are adding a form that you did not test in 2015, you will have to test the new form.
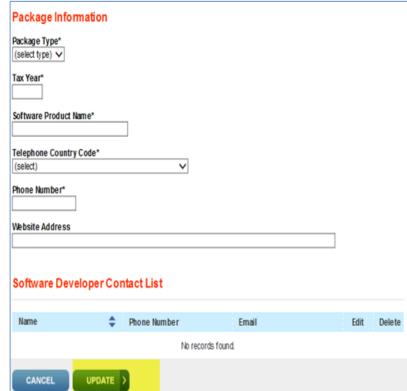
Adding a Software Developer Package

Select Application Details from the menu bar and scroll down to the Software Developer Package List and click "Add"

Complete the Pop Box with the information for your TY 2016 Software Package. If you have more than one package, you will need to repeat this step for each package.

Changes to the AIR Submission Composition and Reference Guide

The AIR Composition and Reference Guide

The AIR Composition and Reference Guide is now Publication 5258

- This Publication can be found on the AIR Program Page

Revised Error and Business Rule Library

- Appendix A, AIR TY2015 Error Code Library, contains the error codes and descriptions that may be returned to the transmitter

- The AIR TY2016 Business Rules can be found on the following irs.gov web site: AIR TY2016 Business Rules

Demonstration of Manifest and Crosswalk

Sample Manifest Schema

Creating the Manifest files

- **UI Channel**

  Use the ISS-UI Schemas to create the "ACA Business Header" and "Request Manifest Details"

  - **ISS-UI – Manifest Header**
    - IRS-ACAUserInterfaceHeaderMessage.xsd

- **A2A Channel**

  Use the ISS-A2A Schemas and WSDL's to create the "ACA Business Header", "Request Manifest Details" and the security elements.

  - **ISS- A2A – Manifest Header**
    - ACABusinessHeader XML
      - IRS-Form1094-1095BCTransmitterReqMessage.xsd
  - IRS-ACASubmitService SOAP Request Message (WSSESecurity and ACASecurity XML)
    - ACAGetTransmitterBulkRequestService.**wsdl**

Sample Manifest Schema

BusinessHeader

<?xml version="1.0" encoding="UTF-8"?>

<p3:ACAUIBusinessHeader xmlns:irs="urn:us:gov:treasury:irs:common"

xmlns="urn:us:gov:treasury:irs:ext:aca:air:ty16" xmlns:p2="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"

xmlns:p1="urn:us:gov:treasury:irs:msg:acabusinessheader"

xsi:schemaLocation="urn:us:gov:treasury:irs:msg:acauibusinessheader IRS-ACAUserInterfaceHeaderMessage.xsd "

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:p3="urn:us:gov:treasury:irs:msg:acauibusinessheader">

Sample Manifest Schema

BusinessHeader- continued

```
<p1:ACABusinessHeader>
<UniqueTransmissionId>Ae2320b33-f329-31d4-a716-
394655440000:SYS12:XXXXX::T</UniqueTransmissionId>
<irs:Timestamp>2017-01-26T21:18:58Z</irs:Timestamp>
</p1:ACABusinessHeader>
<ACATransmitterManifestReqDtl>
<PaymentYr>2016</PaymentYr>
<PriorYearDataInd>0</PriorYearDataInd>
<irs:EIN>XXXXXXXX</irs:EIN>
<TransmissionTypeCd>O</TransmissionTypeCd>
<TestFileCd>P</TestFileCd>
```

Sample Manifest Schema

TransmitterNameGrp

<TransmitterNameGrp>

<BusinessNameLine1Txt>ANY BUSINESS NAME</BusinessNameLine1Txt>

</TransmitterNameGrp>

<CompanyInformationGrp>

<CompanyNm>ANY COMPANY NAME</CompanyNm>

Sample Manifest Schema

MailingAddressGrp

```
<MailingAddressGrp>
<USAddressGrp>
<AddressLine1Txt>Any Address</AddressLine1Txt>
<irs:CityNm>Any City</irs:CityNm>
<USStateCd>XX</USStateCd>
<irs:USZIPCd>XXXXX</irs:USZIPCd>
</USAddressGrp>
</MailingAddressGrp>
<ContactNameGrp>
<PersonFirstNm>XXXXX</PersonFirstNm>
<PersonMiddleNm>X</PersonMiddleNm>
<PersonLastNm>XXXXX</PersonLastNm>
</ContactNameGrp>
<ContactPhoneNum>XXXXXXXXX</ContactPhoneNum>
</CompanyInformationGrp>
```

Sample Manifest Schema

VendorInformationGRP

<VendorInformationGrp>

<VendorCd>V</VendorCd>

<ContactNameGrp>

<PersonFirstNm>XXXXX</PersonFirstNm>

<PersonLastNm>XXXXX</PersonLastNm>

</ContactNameGrp>

<ContactPhoneNum>XXXXXXXXX</ContactPhoneNum>

</VendorInformationGrp>

<TotalPayeeRecordCnt>2</TotalPayeeRecordCnt>

<TotalPayerRecordCnt>1</TotalPayerRecordCnt>

<SoftwareId>16A00000XX</SoftwareId>

<FormTypeCd>1094/1095C</FormTypeCd>

<irs:BinaryFormatCd>application/xml</irs:BinaryFormatCd>

<irs:ChecksumAugmentationNum>85AD808789834645C3115E39A886B1A8</irs:ChecksumAugmentationNum>

<irs:AttachmentByteSizeNum>6903</irs:AttachmentByteSizeNum>

<DocumentSystemFileNm>1094C_Request_XXXXX_20170126T195002000Z.xml</DocumentSystemFileNm>

</ACATransmitterManifestReqDtl>

</p3:ACAUIBusinessHeader>

Sample Manifest Schema

Signing the Header Elements (A2A only)

1. Calculate the digest of the <SignedInfo> element, sign that digest and put the signature value in a <SignatureValue> element

2. If keying information is to be included, place it in a <KeyInfo> element. In the Section 5.3.1.1 (5) sample, the keying information contains the X.509 certificate for the transmitter, which would include the public key needed for signature verification.

3. Place the <SignedInfo>, <SignatureValue>, and <KeyInfo> elements into a <Signature> element. The <Signature> element comprises the XML signature.

Note: See Publication 5258, Section 5.3.1.1 for additional information for creating the XML Signature

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
UniqueTransmissionId	See the AIR Submission Composition and Reference Guide Section 3.4.2 for the UTID composition	Required 1/1	Unique identifier assigned by the transmitter to each transmission Always 51 bytes	\<UniqueTransmissionId\>Ae232 0b33-f329-31d4-a716-394655440000:SYS12:XXXXX:: T\</UniqueTransmissionId\>
Timestamp	TimestampType Base type for a date and time stamp – Timezone portion is required and fractional seconds are prohibited	Required 1/1	The date and time (based on the local time zone of the transmitter) that the SOAP Message was created	\<irs:Timestamp\>2017-01-26T21:18:58Z\</irs:Timestamp\>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
PaymentYr	YearType string pattern "[1-9][0-9]{3}"	Required 1/1	This is the tax year for which the information return is being reported	<:PaymentYr>2016</PaymentYr>
PriorYearDataInd	DigitBooleanType enumerated string value="0" value="1"	Required 1/1	"0" for current filing year "1" for prior filing year	<PriorYearDataInd>0</PriorYearDataInd>
EIN	EINType string pattern "[0-9]{9}"	Required 1/1	Employer Identification Number belonging to the transmitter	<EIN>XXXXXXXXX</EIN>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
TransmissionTypeCd	TransmissionTypeCdType enumerated string value="O" value="C" value="R"	Required 1/1	Type of records: "O" – Original "C" – Corrections "R" - Replacement	\<TransmissionTypeCd>**O**\</TransmissionTypeCd>
TestFileCd	TestFileCdType string pattern "([TP])?"	Required 1/1	"T" – AATS "P" – Production	\<TestFileCd>**P**\</TestFileCd>
OriginalReceiptId	String80Type string maxLength value="80"	Optional 0/1	Reference to the Receipt ID of the transmission for which replacement transmissions are submitted - Do not include XML tags on transmissions with original records	(This is not in the sample XML – use for Replacement transmission) \<OriginalReceiptId>**1095C-16-000XXXXXX**\</OriginalReceiptId>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
TransmitterForeignEntityInd	DigitBooleanType enumerated string value="0" value="1"	Optional 0/1	"1" identifies the transmitter as a foreign entity, Do not include XML tags if transmitter is not a foreign entity	This is not in the sample XML - <TransmitterForeignEntityInd>1</TransmitterForeignEntityInd>
TransmitterNameGrp	BusinessNameType Complex Element	Required 1/1	The transmitter's name	<TransmitterNameGrp> <BusinessNameLine1Txt>**ANY BUSINESS NAME**</BusinessNameLine1Txt> </TransmitterNameGrp>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example

CompanyInformationGrp	CompanyInformationGrpType
	
	Complex Element
		Required
		
		1/1
			The transmitter address, name of the company to contact, and phone number to call	\<CompanyInformationGrp\>
				  \<CompanyNm\>**ANY COMPANY NAME**\</CompanyNm\>
				  \<MailingAddressGrp\>
				   \<USAddressGrp\>
				    \<AddressLine1Txt\>**Any Address**\</AddressLine1Txt\>
				    \<CityNm\>**Any City**\</CityNm\>
				    \<USStateCd\>**XX**\</USStateCd\>
				
				\<USZIPCd\>**XXXXX**\</USZIPCd\>
				   \</USAddressGrp\>
				  \</MailingAddressGrp\>
				  \<ContactNameGrp\>
				   \<PersonFirstNm\> **XXXXX**\</PersonFirstNm\>
				   \<PersonLastNm\> **XXXXX**\</PersonLastNm\>
				  \</ContactNameGrp\>
				  \<ContactPhoneNum\>**XXXXXXXXXX** \</ContactPhoneNum\>
				\</CompanyInformationGrp\>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example

VendorInformationGrp	VendorInformationGrp Type
	
	Complex Element
		Required
		
		1/1
			Identifies if software was developed by a vendor or in-house as well as contact information regarding the software
			
			"V" – Provided by a Vendor
			"I" – Developed In-house
				<VendorInformationGrp>
				  <VendorCd>**V**</VendorCd>
				  <ContactNameGrp>
				
				<PersonFirstNm>**XXXXX**</PersonFirstNm>
				
				<PersonLastNm>**XXXXX**</PersonLastNm>
				  </ContactNameGrp>
				
				<ContactPhoneNum>**XXXXXXXXX**</ContactPhoneNum>
				</VendorInformationGrp>

TotalPayeeRecordCnt	TotalNumberNonNegativeType nonNegativeInteger	Required 1/1	The total number of Forms 1095 included in the entire transmission	<TotalPayeeRecordCnt>**2**</TotalPayeeRecordCnt>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
TotalPayerRecordCnt	TotalNumberType positiveInteger	Required 1/1	The total number of Forms 1094 included in the entire transmission	\<TotalPayerRecordCnt\>**1**\</TotalPayerRecordCnt\>
SoftwareId	SoftwareIdNumberType "StringWithNoHashCharacterType" maxLength value="10"	Required 1/1	The Software ID assigned to the software when the software was registered with the IRS., found in the AIR TCC application. Transmitters/Issuers using a 3rd party vendor need to get the SoftwareId from their vendor.	\<SoftwareId\>**16A0000XXX**\</SoftwareId\>
FormTypeCd	FormNameType enumerated string value="1094/1095B" value="1094/1095C"	Required 1/1	Will contain "1094/1095B" if Forms 1094/1095-B are included in the transmission or "1094/1095C" if Forms 1094/1095-C are included in the transmission	\<FormTypeCd\>**1094/1095C**\</FormTypeCd\>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
BinaryFormatCd	BinaryFormatCodeType enumerated string value="application/xml"	Required 1/1	Will contain "application/xml"	<BinaryFormatCd>**application/xml**</BinaryFormatCd>
ChecksumAugmentationNum	MD5Type string pattern "[0-9A-Fa-f]{32}"	Required 1/1	The MD5 Checksum computed on the Form Data File attached to the transmission Reference: Publication 5258 – Section 3.4.3	<ChecksumAugmentationNum>**85AD808789834645C3115E39A886B1A8**</ChecksumAugmentationNum>

Manifest Crosswalk

Manifest Element Name	eFile Type and eFileType Definition	Required or Optional? minOccurs/ maxOccurs	Description	Example
AttachmentByteSizeNum	Integer integer	Required 1/1	The size in bytes of the Form Data File attached to the transmission Reference: Publication 5258 – Section 5.2.1	\<AttachmentByteSizeNum>**6903**\</AttachmentByteSizeNum>
DocumentSystemFileNm	AIRDocumentSystemFileNmType String value="1094[B|C]_Request_[B-DF-HJ-NP-TV-Z]{2}[B-DF-HJ-NP-TV-Z0-9]{3}_[1-9][0-9]{3}(0[1-9]|1[0-2])(0[1-9]|[1-2][0-9]|3[0-1])T(0[0-9]|1[0-9]|2[0-3])(0[0-9]|[1-5][0-9])(0[0-9]|[1-5][0-9])[0-9]{3}Z\.xml"	Required 1/1	The name of the Form Data File attached to the transmission Reference: Publication 5258– Section 3.3	\<DocumentSystemFileNm>**1094C_Request_XXXXX_20170126T195002000Z.xml**\</DocumentSystemFileNm>

Demonstration of Form 1095-C Schema

Sample Form 1095-C Schema

CorrectedRecordInfoGrp

```xml
<p1:Form1095CUpstreamDetail recordType="String" lineNum="0">
    <p1:RecordId>1</p1:RecordId>
    <p1:CorrectedInd>1</p1:CorrectedInd>
    <p1:CorrectedRecordInfoGrp>
     <p1:CorrectedUniqueRecordId>1095C-16-00000001|1|1</p1:CorrectedUniqueRecordId>
     <p1:CorrectedRecordPayeeName>
       <p1:PersonFirstNm>First</p1:PersonFirstNm>
       <p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
       <p1:PersonLastNm>Last</p1:PersonLastNm>
       <p1:SuffixNm>Suffix</p1:SuffixNm>
     </p1:CorrectedRecordPayeeName>
     <p1:CorrectedRecordPayeeTIN>111111111</p1:CorrectedRecordPayeeTIN>
    </p1:CorrectedRecordInfoGrp>
```

CorrectedRecordInfoGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
1095CUpstreamDetail recordType lineNum	1095CUpstreamDetailType string integer	Form 1095-C Upstream Detail	1 1 1	1 1 1	Required	</p1:Form1095CUpstreamDetail lineNum="0" recordType="">>
RecordId	RecordIdType	string maxLength value="10" pattern value="[1-9][0-9]*"	1	1	Required	<p1:RecordId>1</p1:RecordId>
CorrectedInd	DigitBooleanType	string enumerations: "0" or "1"	1	1	Required	<p1:CorrectedInd>1</p1:CorrectedInd>
CorrectedRecordInfoGrp	CorrectedRecordInfoGrpType	complexType	0	1	Optional	<p1:CorrectedRecordInfoGrp>
CorrectedUniqueRecordId	UniqueRecordIdType	string pattern value="[^\s].[^\s]{1,80}|[1-9]{1}[0-9]{0,15}|[1-9]{1}[0-9]{0,15}"	1	1	Required	<p1:CorrectedUniqueRecordId>1095C-16-00000001|1|1</p1:CorrectedUniqueRecordId>
CorrectedRecordPayeeName	OtherCompletePersonNameType	complexType	0	1	Optional	<p1:CorrectedRecordPayeeName>
PersonFirstNm	PersonFirstNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonFirstNm>First</p1:PersonFirstNm>
PersonMiddleNm	PersonMiddleNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
PersonLastNm	PersonLastNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonLastNm>Last</p1:PersonLastNm>
SuffixNm	SuffixNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:SuffixNm>Suffix</p1:SuffixNm>
CorrectedRecordPayeeTIN	SSNType	string pattern [0-9]{9}	0	1	Optional	<p1:CorrectedRecordPayeeTIN>111111111</p1:CorrectedRecordPayeeTIN>
CorrectedRecordInfoGrp	CorrectedRecordInfoGrpType	complexType	0	1	Optional	</p1:CorrectedRecordInfoGrp>

Sample Form 1095-C Schema

EmployeeInfoGrp

```xml
<p1:TaxYr>2016</p1:TaxYr>
    <p1:EmployeeInfoGrp>
      <p1:OtherCompletePersonName>
        <p1:PersonFirstNm>First</p1:PersonFirstNm>
        <p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
        <p1:PersonLastNm>Last</p1:PersonLastNm>
        <p1:SuffixNm>Suffix</p1:SuffixNm>
      </p1:OtherCompletePersonName>
      <p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>
      <p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>
      <p2:SSN>111111111</p2:SSN>
```

Sample Form 1095-C Schema

EmployeeInfoGrp-continued

```
<p1:MailingAddressGrp>
    <p1:USAddressGrp>
      <p1:AddressLine1Txt>Address Line</p1:AddressLine1Txt>
      <p1:AddressLine2Txt>Address Line 2</p1:AddressLine2Txt>
      <p2:CityNm>City</p2:CityNm>
      <p1:USStateCd>AL</p1:USStateCd>
      <p2:USZIPCd>22222</p2:USZIPCd>
      <p2:USZIPExtensionCd>2222</p2:USZIPExtensionCd>
    </p1:USAddressGrp>
  </p1:MailingAddressGrp>
 </p1:EmployeeInfoGrp>
<p1:ALEContactPhoneNum>1111111111</p1:ALEContactPhoneNum>
<p1:StartMonthNumberCd>04</p1:StartMonthNumberCd>
```

EmployeeInfoGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
TaxYr	YearType	string pattern value="[1-9][0-9]{3}"	1	1	Required	<p1:TaxYr>2016</p1:TaxYr>
EmployeeInfoGrp	EmployeeInformationGrpType	complexType	1	1	Required	<p1:EmployeeInfoGrp>
OtherCompletePersonName	OtherCompletePersonNameType	complexType	1	1	Required	<p1:OtherCompletePersonName>
PersonFirstNm	PersonFirstNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonFirstNm>First</p1:PersonFirstNm>
PersonMiddleNm	PersonMiddleNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
PersonLastNm	PersonLastNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonLastNm>Last</p1:PersonLastNm>
SuffixNm	SuffixNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:SuffixNm>Suffix</p1:SuffixNm>
PersonNameControlTxt	PersonNameControlType	string maxlength value = "4" [A-Z][A-Z\-]{0,3}	0	1	Optional	<p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>

TINRequestTypeCd	TINRequestTypeCodeType	string
		enumerations:
		  INDIVIDUAL_TIN
		  BUSINESS_TIN
		  UNKNOWN
			0	1	Optional	<p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>

SSN	SSNType	string pattern [0-9]{9}	0	1	Optional	<p2:SSN>111111111</p2:SSN>

EmployeeInfoGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
MailingAddressGrp	BusinessAddressGrpType	complexType	0	1	Optional	<p1:MailingAddressGrp>
USAddressGrp	USAddressGrpType	complexType	1	1	Required	<p1:USAddressGrp>
AddressLine1Txt	StreetAddressType	string maxLength="35" pattern [A-Za-z0-9](?[A-Za-z0-9\-/])*	1	1	Required	<p1:AddressLine1Txt>Address Line</p1:AddressLine1Txt>
AddressLine2Txt	StreetAddressType	string maxLength="35" pattern [A-Za-z0-9](?[A-Za-z0-	0	1	Optional	<p1:AddressLine2Txt>Address Line 2</p1:AddressLine2Txt>
CityNm	CityType	string maxLength="22" pattern ([A-Za-z] ?)*[A-Za-z]	1	1	Required	p2:CityNm>City</p2:CityNm>
USStateCd	StateType	string 2 character code	1	1	Required	<p1:USStateCd>AL</p1:USStateCd>
USZIPCd	USZIPCdType	string pattern [0-9]{5}	1	1	Required	<p2:USZIPCd>22222</p2:USZIPCd>
USZIPExtensionCd	USZIPExtensionCdType	string pattern [0-9]{4}	0	1	Optional	<p2:USZIPExtensionCd>2222</p2:USZIPExtensionCd>
USAddressGrp	USAddressGrpType	complexType	1	1	Required	</p1:USAddressGrp>
MailingAddressGrp	BusinessAddressGrpType	complexType	0	1	Optional	</p1:MailingAddressGrp>
EmployeeInfoGrp	EmployeeInformationGrpType	complexType	1	1	Required	</p1:EmployeeInfoGrp>
ALEContactPhoneNum	ContactPhoneNumberType	string minLength="10" Length="15" pattern ([0-9])*	0	1	Optional	<p1:ALEContactPhoneNum>1111111111</p1:ALEContactPhoneNum>
StartMonthNumberCd	PlanStartMonthType	string maxLength="2" pattern (0[0-9]|1[0-2])	0	1	Optional	<p1:StartMonthNumberCd>01</p1:StartMonthNumberCd>

Sample Form 1095-C Schema

EmployeeOfferAndCoverageGrp

```
<p1:EmployeeOfferAndCoverageGrp>
    <p1:AnnualOfferOfCoverageCd>1D</p1:AnnualOfferOfCoverageCd>
    <p1:MonthlyShareOfLowestCostMonthlyPremGrp>
      <p1:JanuaryAmt>120.00</p1:JanuaryAmt>
      <p1:FebruaryAmt>120.00</p1:FebruaryAmt>
      <p1:MarchAmt>120.00</p1:MarchAmt>
      <p1:AprilAmt>150.00</p1:AprilAmt>
      <p1:MayAmt>150.00</p1:MayAmt>
      <p1:JuneAmt>150.00</p1:JuneAmt>
      <p1:JulyAmt>150.00</p1:JulyAmt>
      <p1:AugustAmt>150.00</p1:AugustAmt>
      <p1:SeptemberAmt>150.00</p1:SeptemberAmt>
      <p1:OctoberAmt>150.00</p1:OctoberAmt>
      <p1:NovemberAmt>150.00</p1:NovemberAmt>
      <p1:DecemberAmt>150.00</p1:DecemberAmt>
    </p1:MonthlyShareOfLowestCostMonthlyPremGrp>
    <p1:AnnualSafeHarborCd>2C</p1:AnnualSafeHarborCd>
<p1:EmployeeOfferAndCoverageGrp>
```

EmployeeOfferAndCoverageGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
EmployeeOfferAndCoverageGrp	EmployeeOfferAndCoverageGrpType	complexType	0	1	Optional	<p1:EmployeeOfferAndCoverageGrp>
AnnualOfferOfCoverageCd	OfferCoverageType	string maxLength="2" enumerations 1A-1K or 1a-1k	0	1	Optional	<p1:AnnualOfferOfCoverageCd>1E</p1:AnnualOfferOfCoverageCd>
MonthlyShareOfLowestCostMonthlyPremGrp	AmountByMonthDetailType	complexType	0	1	Optional	<p1:MonthlyShareOfLowestCostMonthlyPremGrp>
JanuaryAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:JanuaryAmt>120.00</p1:JanuaryAmt>
FebruaryAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:FebruaryAmt>120.00</p1:FebruaryAmt>
MarchAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:MarchAmt>150.00</p1:MarchAmt>
AprilAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:AprilAmt>150.00</p1:AprilAmt>
MayAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:MayAmt>150.00</p1:MayAmt>

EmployeeOfferAndCoverageGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
JuneAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:JuneAmt>150.00</p1:JuneAmt>
JulyAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:JulyAmt>150.00</p1:JulyAmt>
AugustAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:AugustAmt>150.00</p1:AugustAmt>
SeptemberAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:SeptemberAmt>150.00</p1:SeptemberAmt>
OctoberAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:OctoberAmt>150.00</p1:OctoberAmt>
NovemberAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:NovemberAmt>150.00</p1:NovemberAmt>
DecemberAmt	AmountType	string maxLength value="19"/> pattern value="([0-9]+\.[0-9][0-9]?)"	0	1	Optional	<p1:DecemberAmt>150.00</p1:DecemberAmt>
MonthlyShareOfLowestCost MonthlyPremGrp	AmountByMonthDetailType	complexType	0	1	Optional	</p1:MonthlyShareOfLowestCostMonthlyPremGrp>
AnnualSafeHarborCd	SafeHarborCdType	"string" maxLength="2" enumerations added 2A-2H, 2a-2h	0	1	Optional	<p1:AnnualSafeHarborCd>2C</p1:AnnualSafeHarborCd>
EmployeeOfferAndCoverage Grp	EmployeeOfferAndCoverage GrpType	complexType	0	1	Optional	</p1:EmployeeOfferAndCoverageGrp>

Sample Form 1095-C Schema

CoveredIndividualGrp (1st covered individual)

```
<p1:CoveredIndividualInd>1</p1:CoveredIndividualInd>
    <p1:CoveredIndividualGrp>
     <p1:CoveredIndividualName>
       <p1:PersonFirstNm>First</p1:PersonFirstNm>
       <p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
       <p1:PersonLastNm>Lasting</p1:PersonLastNm>
       <p1:SuffixNm>Suffix</p1:SuffixNm>
     </p1:CoveredIndividualName>
     <p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>
     <p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>
     <p2:SSN>222222222</p2:SSN>
     <p1:CoveredIndividualAnnualInd>1</p1:CoveredIndividualAnnualInd>
    <p1:CoveredIndividualGrp>
```

First CoveredIndividualGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
CoveredIndividualInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:CoveredIndividualInd>1</p1:CoveredIndividualInd>
CoveredIndividualGrp	EmployerCoveredIndividualType	complexType	0	99	Optional	<p1:CoveredIndividualGrp>
CoveredIndividualName	OtherCompletePersonNameType	complexType	0	1	Optional	<p1:CoveredIndividualName>
PersonFirstNm	PersonFirstNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonFirstNm>First</p1:PersonFirstNm>
PersonMiddleNm	PersonMiddleNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
PersonLastNm	PersonLastNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonLastNm>Lasting</p1:PersonLastNm>
SuffixNm	SuffixNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:SuffixNm>Suffix</p1:SuffixNm>
PersonNameControlTxt	PersonNameControlType	string maxlength value = "4" [A-Z][A-Z\-]{0,3}	0	1	Optional	<p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>
TINRequestTypeCd	TINRequestTypeCodeType	string enumerations: INDIVIDUAL_TIN BUSINESS_TIN UNKNOWN	0	1	Optional	<p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>
SSN	SSNType	string pattern [0-9]{9}	0	1	Optional	<p2:SSN>222222222</p2:SSN>
CoveredIndividualAnnualInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:CoveredIndividualAnnualInd>1</p1:CoveredIndividualAnnualInd>
CoveredIndividualGrp	EmployerCoveredIndividualType	complexType	0	99	Optional	<p1:CoveredIndividualGrp>

Sample Form 1095-C Schema

CoveredIndividualGrp (2nd covered individual)

```
<p1:CoveredIndividualGrp>
    <p1:CoveredIndividualName>
      <p1:PersonFirstNm>Second</p1:PersonFirstNm>
      <p1:PersonMiddleNm>Mid</p1:PersonMiddleNm>
      <p1:PersonLastNm>Lasting</p1:PersonLastNm>
      <p1:SuffixNm>Suffix</p1:SuffixNm>
    </p1:CoveredIndividualName>
    <p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>
    <p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>
    <p1:BirthDt>1980-03-17</p1:BirthDt>
    <p1:CoveredIndividualMonthlyIndGrp>
     <p1:AprilInd>1</p1:AprilInd>
     <p1:MayInd>1</p1:MayInd>
     <p1:JuneInd>1</p1:JuneInd>
     <p1:JulyInd>1</p1:JulyInd>
     <p1:AugustInd>1</p1:AugustInd>
     <p1:SeptemberInd>1</p1:SeptemberInd>
     <p1:OctoberInd>1</p1:OctoberInd>
     <p1:NovemberInd>1</p1:NovemberInd>
     <p1:DecemberInd>1</p1:DecemberInd>
    </p1:CoveredIndividualMonthlyIndGrp>
   </p1:CoveredIndividualGrp>
  </p1:Form1095CUpstreamDetail>
```

Second CoveredIndividualGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
CoveredIndividualInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:CoveredIndividualInd>1</p1:CoveredIndividualInd>
CoveredIndividualGrp	EmployerCoveredIndividualType	complexType	0	99	Optional	<p1:CoveredIndividualGrp>
CoveredIndividualName	OtherCompletePersonNameType	complexType	0	1	Optional	<p1:CoveredIndividualName>
PersonFirstNm	PersonFirstNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonFirstNm>First</p1:PersonFirstNm>
PersonMiddleNm	PersonMiddleNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:PersonMiddleNm>Middle</p1:PersonMiddleNm>
PersonLastNm	PersonLastNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	1	1	Required	<p1:PersonLastNm>Lasting</p1:PersonLastNm>
SuffixNm	SuffixNameType	string maxLength="20" ([A-Za-z\-] ?)*[A-Za-z\-]	0	1	Optional	<p1:SuffixNm>Suffix</p1:SuffixNm>
PersonNameControlTxt	PersonNameControlType	string maxlength value = "4" [A-Z][A-Z\-]{0,3}	0	1	Optional	<p1:PersonNameControlTxt>LAST</p1:PersonNameControlTxt>
TINRequestTypeCd	TINRequestTypeCodeType	string enumerations: INDIVIDUAL_TIN BUSINESS_TIN UNKNOWN	0	1	Optional	<p2:TINRequestTypeCd>INDIVIDUAL_TIN</p2:TINRequestTypeCd>
SSN	SSNType	string pattern [0-9]{9}	0	1	Optional	<p2:SSN>222222222</p2:SSN>
CoveredIndividualAnnualInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:CoveredIndividualAnnualInd>1</p1:CoveredIndividualAnnualInd>
CoveredIndividualGrp	EmployerCoveredIndividualType	complexType	0	99	Optional	</p1:CoveredIndividualGrp>

Second CoveredIndividualGrp Crosswalk

Element Name	eFile Type	eFile Type Definition	min Occurs	max Occurs	Required or Optional	Example
CoveredIndividualMonthlyIndGrp	MonthIndGrpType	complexType	0	1	Optional	<p1:CoveredIndividualMonthlyIndGrp>
AprilInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:AprilInd>1</p1:AprilInd>
MayInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:MayInd>1</p1:MayInd>
JuneInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:JuneInd>1</p1:JuneInd>
JulyInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:JulyInd>1</p1:JulyInd>
AugustInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:AugustInd>1</p1:AugustInd>
SeptemberInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:SeptemberInd>1</p1:SeptemberInd>
OctoberInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:OctoberInd>1</p1:OctoberInd>
NovemberInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:NovemberInd>1</p1:NovemberInd>
DecemberInd	DigitBooleanType	string "0" or" 1"	0	1	Optional	<p1:DecemberInd>1</p1:DecemberInd>
CoveredIndividualMonthlyIndGrp	MonthIndGrpType	complexType	0	1	Optional	<p1:CoveredIndividualMonthlyIndGrp>
CoveredIndividualGrp	EmployerCoveredIndividualType	complexType	0	99	Optional	</p1:CoveredIndividualGrp>
1095CUpstreamDetail	1095CUpstreamDetailType	Form 1095-C Upstream Details	1	1	Required	</p1:Form1095CUpstreamDetail>

October Webinar Questions and Responses

October Webinar Questions and Responses

Slides 42 through 46 provide answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

October Webinar Questions and Responses

ID	Question	Answer
1	I am a Software Developer that was approved and 2015 software ID is in production. I updated my profile and received a software ID for 2016. This is currently in Test. Are software developers required to pass scenario testing for TY 2016 if they have already passed TY 2015 scenario testing and are in production mode?	No, Software Developers, who passed AATS for Tax Year 2015, will not have to retest for Tax Year 2016. Your 2016 software products will be moved to production automatically.
2	After getting approved to submit Production Files, can a Software Developer (In-House package) submit the files to IRS under his own TCC? Or Should only a Transmitter TCC be used to submit the files in Production System?	The Software Developer TCC is only used for Testing. If you also have the Issuer or Transmitter TCC, you can submit you own files.
3	We are moving to A2A; however, we have some questions about how to request the certificate. Could we have a technical call with your team to assist us as we move to A2A?	Review Publication 5258, Section 5. If you still have questions or need to troubleshoot your files, contact the AIR Mailbox.
4	Does the empty tags resulting in rejection apply to only AATS...or does it also apply to production?	The AIR schemas will not allow empty or null values within schema tags for AATS or Production for TY2016.
5	I'm a transmitter do I need to re-re-register for TY2016?	No, as long as your e-Services account is still active, you do not need to register again for TY2016.
6	If a submission on the transmission is rejected, do I have to resend the submission in replacement transmission? or can I resend the record instead of submission in a correction transmission?	When a status check reveals the transmission was rejected. it must be replaced not corrected. Please see Publication 5165 Section 7.2 for the replacement process.

This slide provides answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

October Webinar Questions and Responses

ID	Question	Answer
7	If someone needs to make a correction to a 2015 submission in 2017, will they need to use the 2015 Schema for that submission or the 2016 (latest) schema?	For Tax Year 2015, Use the schemas and business rules that are in effect for Tax Year 2015, Do not use the Tax Year 2016 schemas or rules.
8	Is annual testing required for Transmitters?	No, annual testing is only for software developers. See Publication 5164 Section 2.1 for more information.
9	Since your return error file is more specific for TY2016, it still has to be limited to 100 MB. If there are multiple error files we will have to match them by receipt id? Is that correct?	Only one Error Data File will ever be created per Transmission. However, starting with Tax Year 2016 the Error Data File size has been increased to 200 MB.
10	Please clarify "tags".	A tag is a markup construct that begins with < and ends with >. Tags form the foundation of XML, they define the scope of an element in the XML.
11	Is the max total size SOAP transmitted to the AIR system before compression 100MB, or can the compressed Form data be up to 100MB?	The Form Data File (UI) or Message Transmission Optimization Mechanism (MTOM) (A2A) attachment may not exceed 100 MB of uncompressed native XML.
12	Is there going to be another update to the AIR Submission Composition and Reference Guide Version? Or is this guide the one that will be used for 2016?	An early-look of the AIR Submission Composition and Reference Guide Version 1.0 October 2016 (Now Publication 5258) has already posted to the AIR Program Page.

This slide provides answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

October Webinar Questions and Responses

ID	Question	Answer
13	I have a Form 4419 (FIRE) application on record from 2015, can I use this for AIR? How do I get an e-Services account?	The FIRE TCC application cannot be used for the AIR System. You will need to complete the ACA Application for Transmitter Control Code. More information on how to apply can be found in Section 3 of the Publication 5164 (Processing Year 2017).
14	I used a third-party to submit our information last year and was wondering if the instructions for signing up for AIR program different from last year? If so, where can I get the most updated checklist to start that process? Thanks.	Information on e-Services registration and the Application for TCC can be found in Section 3 of the Publication 5164 (Processing Year 2017) and the AIR Program landing page "Program Overview"
15	For AIRTN500 errors the specific individual the error was received for will be noted. Is that for all cases or only if a specific data element is provided in the XML with the transmission?	The Xpath will indicate which element in the Record contains the error. This will allow you to identify which specific individual contains the error. Please see Figure 4-7 of the Publication 5258, AIR Submission Composition and Reference Guide.
16	What are the changes in WSDL for TY2016 compared to TY2015 wsdl and is there any change to Manifest File and Bulk data File schema?	Please refer to the DIFF File located in the Schema package on the AIR Program page for changes in the Schema. The Endpoints have been updated in the WSDL.
17	Is there an example of the expanded TIN validation failures for a specific dependent? How is the dependent referenced in the error message?	The Xpath will indicate which element in the Record contains the error. This will allow you to identify which specific individual contains the error. Please see Figure 4-7 of the Publication 5258, AIR Submission Composition and Reference Guide.
18	When will the new wsdl files be released	The Filing Season 2017/Tax Year 2016 (FS17/TY16) WSDL packages are currently available.

This slide provides answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

October Webinar Questions and Responses

ID	Question	Answer
19	How do we register to get quick alerts	If you are not subscribed to AIR QuickAlerts, click the link here to subscribe. Choose the Affordable Care Act Information Return (AIR) category.
20	Are the A2A endpoints for production going to be the same for 2016 transmissions or are there going to be 2016 production endpoints?	The Filing Season 2017 WSDL Package contains the appropriate WSDL package and associated ends points for AATS and Production.
21	When is production planned to be available for Tax Year 2016 submissions?	The opening date for Filing Season 2017 Production is planned for January 2017, however an exact date has not been published. A QuickAlerts announcing the opening date will be issued in the future.
22	Are TIN errors only identified in the testing environment or will they also be identified in production?	TIN errors are not identified during AATS. AATS only checks against the mock TIN and Name Controls listed in Publication 5164.
23	Previously it was stated that blank tags although optional would be required if they are referenced in a business rule, are you now stating that no blank elements will be allowed regardless of whether they are used in the business rule?	Please do not include blank tags for TY2016. It will cause schema validation errors. This is a change from TY2015.
24	In the Sept Webinar questions and responses you say the Business Name Control will be required Where does the value come from? Is that supplied by the IRS when the TCC is assigned? Or, is this something we create ourselves?	Business name control is not required. We are requesting that name controls be included for the TIN Validation process. They are still optional elements and they will not generate errors if they are missing. Instructions for creating name controls are referenced in Publication 5165 Appendix 10. "For details on Name Control for Businesses, see Section 3.11 in Publication 4163." "For details on Name Control for Individuals, see Exhibit 5 in Publication 4164"

This slide provides answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

October Webinar Questions and Responses

ID	Question	Answer
25	Will we be required to use Name Control Tags in the xml we submit tot he IRS for TY 2016?	Name controls are not required. We requested that name control elements, (with a value), be voluntarily included in the xml for TIN Validation purposes. If they are not included in the submission, no error will be set.
26	If this question is not addressed in the presentation. Section 6.2, Page 46 of p5165 states: "Also new for TY2016, if the same business rule is violated too many times the entire submission will be rejected and a generic threshold error will be returned in the Error Data File." Please elaborate on thresholds. Is this threshold of errors a percentage of the overall file size, a set number of errors, etc. ?	Threshold levels vary based on the business rule and the criticality of the information enforced by the business rule. Since the thresholds vary by business rule, it is impossible to provide a single threshold value. For this reason, the irs does not share the values externally.
27	What is the threshold for percentage of errors per file for a file to be rejected?	The thresholds vary for each business rule.
28	the rule that does not allow empty tags etc. Is this a rule for 2015 corrections? Or just 2016 and forward?	TY2015 corrections may have empty tags. Do not include empty tags for TY2016.
29	Will the XPathContent in the Error Data File include namespace prefixes or will the local name only be shown in the paths?	Please see the Error Data File examples in Publication 5165, Section 6.2.
30	When submitting a correction for 2015, should the software id on the manifest file be the 2015 software id?	Yes, use the 2015 software id.
31	Having successfully uploaded our1094-C & 1095-C forms for TY2015, will we have to test again?	Software Developers, who passed AATS for Tax Year 2015, will not have to retest for Tax Year 2016. However Software Developers need a new Software ID for 2016, each tax year, and each ACA Information Return Type they support. To update your application, the Responsible Official should go to the Application Details page and click the "Add" button under the Software Developer Package List.
32	Is the removal of all blank tags/elements only for AATS testing or is it to also occur in the production environment as well?	Do not include blank tags in both TY2016 AATS and Production.

This slide provides answers to common questions posed by software developers at the October AIR webinar. The answers provided are based upon our interpretation of the question and are valid as of November 15, 2016. Future changes in technical requirements may change an answer. None of the questions and answers are intended to address any legal or policy issues related to the ACA information return requirements.

General Information

General AIR Information

The purpose of the AIR Mailbox is to provide technical assistance with the transmission of data to the IRS related to:

- Publication 5164
- Publication 5165
- Publication 5258
- Automated Enrollment Guide
- WSDLs
- Schema
- Transmission File review
- Transmission errors

Note: The AIR Mailbox does not support Tax Law, refer to IRS.gov/aca [Affordable Care Act (ACA) Tax Provisions](#)

General AIR Information

The Help Desk provides customer support for the following:

- Registration and confirmation process for e-services for Responsible Officials and Contacts
- ACA Application for a Transmitter Control Code (TCC) process
  - Navigation of the ACA Application for a TCC
  - Modification / Updates of the ACA Application TCC
  - Production / Test Status
- Assurance Testing for Software Developers and Transmitters
  - AATS Testing review and feedback
- Transmission / Acknowledgement status
- Business Rules / Error Code resolution
- Move TCC from Test to Production, and to retrieve Receipt IDs when they are lost or not returned

Note: The Help Desk does not support Tax law, refer to IRS.gov/aca

Affordable Care Act (ACA) Tax Provisions

General AIR Information

Help Desk Reminders:

- All Responsible Officials and Contacts need to Register and confirm your registration in e-Services before the application for TCC can be submitted.

- Ensure your ACA Application for TCC is kept up to date.

- Software Developers should ensure their transmission has been Accepted before contacting the help desk for their testing results.

- **Review the answer key located on IRS.gov before submitting your test scenarios. The answer keys are provided for your convenience, if your test does not contain all of the information shown in the answer key, your submission has not passed AATS testing.**

  Important!

- Once you have passed AATS testing and your Software Package is placed in Production, allow 48 hours before transmitting into the AIR Production environment.

- When submitting to Production, be sure to use your Transmitter or Issuer TCC, as appropriate.

- Review Publication 5164 and Publication 5165 for Acknowledgement Status and actions needed.