Integrated Outpatient Code Editor (IOCE) Software

Java API Guide

Version 25.1 April 2024

Table of Contents

Integrated Outpatient Code Editor (IOCE) Software Java API Guide

This document describes how to call the IOCE Software from a Java® application. It also notes the necessary dependencies required to execute the IOCE software. Suggested system requirements are also provided. The use of IOCE software outside of a Java calling program is also possible but outside the scope of this document. There is also an included sources jar for IOCE.

System Requirements

Oracle®, OpenJDK, or other Java® version 8 or greater.

Program & Dependency Information

Table 1.   Open-Source Dependencies

Third party software name	Version	Are the distributed binaries modified by 3M?	Purpose	Licensing
Google Protocol Buffers: protobuf-java-3.22.2.jar URL: https://developers.google.com/protocol-buffers	3.22.2	No modification / Not a derivative work; only aggregated with 3M software.	Embedded content storage/processing speed optimization	Copyright 2021 Google Inc. All rights reserved License Type: New BSD License URL: https://ptolemy.berkeley.edu/ptolemyII/ptII11.0/ptII/lib/protobuf-license.htm
SLF4J API: slf4j-api-.jar URL: http://www.slf4j.org/	1.7.36	No modification / Not a derivative work; only aggregated with 3M software.	Logging	Copyright (c) 2004-2017 QOS.ch License Type: SLF4j (MIT License) URL: http://www.slf4j.org/license.html

Third party software name	Version	Are the distributed binaries modified by 3M?	Purpose	Licensing
gfc-base-api-3.4.9.jar	3.4.9	Grouper Foundation Class Common grouper calling framework and common business objects to standardize grouper development https://github.com/3mcloud/GFC-Grouper-Foundation-Classes	3M created open-source binary	

Table 2. Program

Jar name	Version	Purpose	Information
ioce-standalone-25.1.0.x.jar	25.1.0.x	Provide access to core component binary data necessary for proper run-time processing.	Contained in ZIP file. Separate *-sources Jar file also included

Calling the Application

Call the application using the following steps:

1. Include the program and required dependencies in your editor's/compiler's classpath.

2. Create an instance of an OceClaim object.

3. Populate the OceClaim with the provided input methods.

4. Create an instance of the IoceComponent. Note: if you'll need to lookup descriptions before processing the claim, create the IoceComponent earlier and use the version-less getLatestXXXDescription methods.

5. Pass the OceClaim object as a parameter to a new instance of IoceClaim.

6. Pass the IoceClaim instance as a parameter to the process method of the IoceComponent.

7. Check the return code to confirm that processing succeeded.

8. If successfully processed, call the OceClaim's output methods to retrieve the results.

Example Code

This example below illustrates the prior steps using hardcoded claim information.

```java
import com.mmm.his.cer.foundation.exception.FoundationException;
import gov.cms.oce.IoceClaim;
import gov.cms.oce.IoceComponent;
import gov.cms.oce.model.external.OceClaim;
import gov.cms.oce.model.external.OceClaimFactory;
import gov.cms.oce.model.external.OceDiagnosisCode;
import gov.cms.oce.model.external.OceHcpcsModifier;
import gov.cms.oce.model.external.OceLineItem;
import gov.cms.oce.model.external.OceValueCode;

public class OceCallerExample{

        public OceCallerExample(){
          super();
        }

        public final void runOceCallerExample(){

          IoceComponent ioceComponent=null;
          try{
            //STEP 1: CREATE COMPONENT
            ioceComponent=new IoceComponent();

            //STEP 2: CREATE CLAIM OBJECT AND LOAD INPUT
            //Retrieve an instance of the OceClaimFactory
            final OceClaimFactory factory=OceClaimFactory.getInstance();
            //Create an empty claim object
            final OceClaim ioceClaim=factory.createClaim();
            //Identify the claim
            ioceClaim.setClaimId( "Example Claim" );
            //Parameters: 3-digit age of patient
            ioceClaim.setAge( "65" );
            //Parameters: 1-digit sex of patient (0=unknown, 1=male, 2=female)
            ioceClaim.setSex( "1" );
            //Parameters: 8-digit start date (from date) of claim (YYYYMMDD)
            ioceClaim.setDateStarted( "20230101" );
            //Parameters: 8-digit end date (through date) of claim (YYYYMMDD)
            ioceClaim.setDateEnded( "20230102" );
            //Parameters: 3-character bill type
            ioceClaim.setBillType( "131" );
            //Parameters: 2-character patient discharge status value
            ioceClaim.setPatientStatus( "01" );
            //Parameters: 1-digit OPPS flag value (1=OPPS, 2=Non-OPPS)
```

```
ioceClaim.setOppsFlag( "1" );
//Parameters: 13-character national provider number
ioceClaim.setNationalProviderId( "1234567890" );
//Parameters: 6-character CMS certification number (previously OSCAR)
ioceClaim.setCmsCertificationNumber( "123456" );
//Parameters: 2-character occurrence code
ioceClaim.addOccurrenceCodeInput( "05" );
//Parameters: 2-character condition code
ioceClaim.addConditionCodeInput( "15" );
ioceClaim.addConditionCodeInput( "25" );
//Parameters: 2-character value code, 9-character value amount
ioceClaim.addValueCodeInput( "59", "000004302" );
//Parameters: 8-digit receipt date of claim (YYYYMMDD)
ioceClaim.setReceiptDate( "20230101" );

//Parameters: diagnosis code, present on admission value (currently unused)
ioceClaim.addReasonForVisitDiagnosisCode( "S3215XK", "U" );
//Parameters: diagnosis code, present on admission value (currently unused)
ioceClaim.setPrincipalDiagnosisCode( "S3215XK", null );
//Parameters: diagnosis code, present on admission value (currently unused)
ioceClaim.addSecondaryDiagnosisCode( "S72044D", null );
ioceClaim.addSecondaryDiagnosisCode( "17210", null );

//Create a line object
final OceLineItem line01=factory.createLineItem();
//Parameters: 8-digit service date of line (YYYYMMDD)
line01.setServiceDate( "20230101" );
//Parameters: 4-digit revenue center code of line
line01.setRevenueCode( "9999" );
//Parameters: 5-character CPT/HCPCS value of line
line01.setHcpcs( "27279" );
//Parameters: 9-digit unit value
line01.setUnitsInput( "1" );
//Parameters: 10-digit submitted charges value
line01.setCharge( "435.00" );
//add line object to claim
ioceClaim.addLineItem( line01 );

//Create a second line object (includes optional payer-only fields)
final OceLineItem line02=factory.createLineItem();
//Parameters: 8-digit service date of line (YYYYMMDD)
line02.setServiceDate( "20230101" );
//Parameters: 4-digit revenue center code of line
line02.setRevenueCode( "0360" );
//Parameters: 5-character CPT/HCPCS value of line
line02.setHcpcs( "29305" );
//Parameters: 2-character CPT/HCPCS modifier value for CPT/HCPCS
line02.addHcpcsModifierInput( "22" );
line02.addHcpcsModifierInput( "ZZ" );
//Parameters: 9-digit unit value
line02.setUnitsInput( "1" );
//Parameters: 10-digit submitted charges value
```

```
line02.setCharge( "191.78" );
//Parameters: 1-character action flag (payer-only field)
line02.setActionFlagInput( "0" );
//add line object to claim
ioceClaim.addLineItem( line02 );

//Create a third line object (includes optional payer-only fields)
final OceLineItem line03=factory.createLineItem();
//Parameters: 8-digit service date of line (YYYYMMDD)
line03.setServiceDate( "20230101" );
//Parameters: 4-digit revenue center code of line
line03.setRevenueCode( "0610" );
//Parameters: 5-character CPT/HCPCS value of line
line03.setHcpcs( "72196" );
//Parameters: 9-digit unit value
line03.setUnitsInput( "1" );
//Parameters: 10-digit submitted charges value
line03.setCharge( "140.67" );
//Parameters: 1-character action flag (payer-only field)
line03.setActionFlagInput( "0" );
//add line object to claim
ioceClaim.addLineItem( line03 );

//Create a fourth line object (includes optional payer-only fields)
final OceLineItem line04=factory.createLineItem();
//Parameters: 8-digit service date of line (YYYYMMDD)
line04.setServiceDate( "20230101" );
//Parameters: 4-digit revenue center code of line
line04.setRevenueCode( "0610" );
//Parameters: 5-character CPT/HCPCS value of line
line04.setHcpcs( "72197" );
//Parameters: 9-digit unit value
line04.setUnitsInput( "1" );
//Parameters: 10-digit submitted charges value
line04.setCharge( "600.25" );
//Parameters: 1-character action flag (payer-only field)
line04.setActionFlagInput( "0" );
//add line object to claim
ioceClaim.addLineItem( line04 );

//Create a fifth line object (includes optional payer-only fields)
final OceLineItem line05=factory.createLineItem();
//Parameters: 8-digit service date of line (YYYYMMDD)
line05.setServiceDate( "20230102" );
//Parameters: 4-digit revenue center code of line
line05.setRevenueCode( "0610" );
//Parameters: 5-character CPT/HCPCS value of line
line05.setHcpcs( "2010F" );
//Parameters: 9-digit unit value
line05.setUnitsInput( "2" );
//Parameters: 10-digit submitted charges value
line05.setCharge( "45.98" );
```

```
//Parameters: 1-character action flag (payer-only field)
line05.setActionFlagInput( "0" );
//add line object to claim
ioceClaim.addLineItem( line05 );

//STEP 3 (Optional): If descriptions are needed for input fields before the internal version is known,
//use the getLatestXXXDescription variants of the description methods to retrieve the latest
//description value for the input. This is useful for integrations with graphical user interfaces.
System.out.println( "==="+ioceClaim.getProcessingInformation().getClaimId()+"===" );
for( final OceLineItem ioceLine : ioceClaim.getLineItemList() ){
   final String latestHcpcsDescription=ioceComponent.getLatestHcpcsDescription( line01.getHcpcs() );
   System.out.println( "CPT/HCPCS = '"+line01.getHcpcs()+"' "+latestHcpcsDescription );
   for( OceHcpcsModifier ioceHcpcsModifier : ioceLine.getHcpcsModifierInputList() ){
      final String latestModifierDescription=ioceComponent.getLatestHcpcsModifierDescription(
ioceHcpcsModifier.getHcpcsModifier() );
         System.out.println( "\tCPT/HCPCS Modifier = '"+ioceHcpcsModifier.getHcpcsModifier()+"'
"+latestModifierDescription );
      }
   }

   System.out.println();
   final String latestPrincipalDiagnosisDescription=ioceComponent.getLatestDiagnosisDescription(
ioceClaim.getPrincipalDiagnosisCode().getDiagnosis() );
   System.out.println( "Principal Diagnosis = '"+ioceClaim.getPrincipalDiagnosisCode().getDiagnosis()+"'
"+latestPrincipalDiagnosisDescription );

   for( final OceDiagnosisCode oceReasonForVisitDiagnosis : ioceClaim.getReasonForVisitDiagnosisCodeList() ){
      final String latestDiagnosisDescription=ioceComponent.getLatestDiagnosisDescription(
oceReasonForVisitDiagnosis.getDiagnosis() );
         System.out.println( "Reason for Visit Diagnosis = '"+oceReasonForVisitDiagnosis.getDiagnosis()+"'
"+latestDiagnosisDescription );
      }

   for( final OceDiagnosisCode oceSecondaryDiagnosis : ioceClaim.getSecondaryDiagnosisCodeList() ){
      final String latestDiagnosisDescription=ioceComponent.getLatestDiagnosisDescription(
oceSecondaryDiagnosis.getDiagnosis() );
         System.out.println( "Secondary Diagnosis = '"+oceSecondaryDiagnosis.getDiagnosis()+"' "+latestDiagnosisDescription
);
      }

   //STEP 4: PROCESS CLAIM
   ioceComponent.process( new IoceClaim( ioceClaim ) );

   //STEP 5: ACCESS CLAIM OUTPUT GENERATED BY PROCESSING CLAIM
   //Check return code after processing
   final Integer returnCode=ioceClaim.getProcessingInformation().getReturnCode();
   final String returnCodeDescription=ioceComponent.getLatestErrorDescription( returnCode.toString() );
   System.out.println();
   System.out.println( "Return Code = '"+returnCode+"' "+returnCodeDescription );
   if( returnCode!=null
   && 0 < returnCode.intValue() ){
      //Some error condition prevented processing...
```

```
            }
            else{
               //Success condition occurred
               //Retrieve internal version number (used for description lookups)
               final short internalVersion=ioceClaim.getProcessingInformation().getInternalVersion();
               System.out.println( "Internal Version = "+internalVersion );
               /* Disposition Identifiers (used for description lookups)
                * 1=Overall Claim Disposition
                * 2=Claim Rejection Disposition
                * 3=Claim Denial Disposition
                * 4=Return to Provider Disposition
                * 5=Suspension Disposition
                * 6=Line Item Rejection Disposition
                * 7=Line Item Denial Disposition
                **/

               //Retrieve claim-level output generated by the component
               final String apcReturnBufferFlag=ioceClaim.getApcReturnBufferFlag();
               System.out.println( "APC Return Buffer = '"+apcReturnBufferFlag+"'" );
               final String nonOppsBillFlag=ioceClaim.getNoppsBillFlag();
               System.out.println( "Non-OPPS Bill Type Flag = '"+nonOppsBillFlag+"'" );

               final String claimProcessedFlag=ioceClaim.getClaimProcessedFlag();
               final String claimProcessedFlagDescription=ioceComponent.getClaimProcessedFlagDescription( claimProcessedFlag,
internalVersion );
               System.out.println( "Claim Processed Flag = '"+claimProcessedFlag+"' "+claimProcessedFlagDescription );

               final String claimDisposition=ioceClaim.getClaimDisposition();
               final String claimDispositionDescription=ioceComponent.getClaimDispositionDescription( "1", internalVersion );
               final String claimDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "2",
claimDisposition, internalVersion );
               System.out.println( "Claim Disposition = '"+claimDisposition+"' '"+claimDispositionValueDescription+"'
"+claimDispositionDescription );

               //Retrieve claim-level rejection disposition and edits generated by the component
               final String claimRejectionDisposition=ioceClaim.getClaimRejectionDisposition();
               final String claimRejectionDispositionDescription=ioceComponent.getClaimDispositionDescription( "2",
internalVersion );
               final String claimRejectionDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "2",
claimRejectionDisposition, internalVersion );
               System.out.println( "Claim Rejection Disposition = '"+claimRejectionDispositionValueDescription+"'
("+claimRejectionDisposition+") "+claimRejectionDispositionDescription );
               if( !ioceClaim.getClaimRejectionEditList().isEmpty() ){
                  final String editDispositionDescription=ioceComponent.getEditDispositionDescription( claimRejectionDisposition,
internalVersion );
                  for( final String edit : ioceClaim.getClaimRejectionEditList() ){
                     //Access edit value
                     final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                     System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
                  }
               }
```

```
//Retrieve claim-level denial disposition and edits generated by the component
final String claimDenialDisposition=ioceClaim.getClaimDenialDisposition();
final String claimDenialDispositionDescription=ioceComponent.getClaimDispositionDescription( "3", internalVersion );
final String claimDenialDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "3",
claimDenialDisposition, internalVersion );
System.out.println( "Claim Denial Disposition = '"+claimDenialDisposition+"'
'"+claimDenialDispositionValueDescription+"' "+claimDenialDispositionDescription );
if( !ioceClaim.getClaimDenialEditList().isEmpty() ){
  final String editDispositionDescription=ioceComponent.getEditDispositionDescription( claimDenialDisposition,
internalVersion );

  for( final String edit : ioceClaim.getClaimDenialEditList() ){
    //Access edit value
    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
    System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
  }
}

//Retrieve claim-level return to provider disposition and edits generated by the component
final String claimReturnToProviderDisposition=ioceClaim.getClaimReturnToProviderDisposition();
final String claimReturnToProviderDispositionDescription=ioceComponent.getClaimDispositionDescription( "4",
internalVersion );
final String claimReturnToProviderDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription(
"4", claimReturnToProviderDisposition, internalVersion );
System.out.println( "Return to Provider Disposition = '"+claimReturnToProviderDisposition+"'
'"+claimReturnToProviderDispositionValueDescription+"' "+claimReturnToProviderDispositionDescription );
if( !ioceClaim.getClaimReturnToProviderEditList().isEmpty() ){
  final String editDispositionDescription=ioceComponent.getEditDispositionDescription(
claimReturnToProviderDisposition, internalVersion );

  for( final String edit : ioceClaim.getClaimReturnToProviderEditList() ){
    //Access edit value
    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
    System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
  }
}

//Retrieve claim-level suspension disposition and edits generated by the component
final String claimSuspensionDisposition=ioceClaim.getClaimSuspensionDisposition();
final String claimSuspensionDispositionDescription=ioceComponent.getClaimDispositionDescription( "5",
internalVersion );
final String claimSuspensionDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "5",
claimSuspensionDisposition, internalVersion );
System.out.println( "Suspension Disposition = '"+claimSuspensionDisposition+"'
'"+claimSuspensionDispositionValueDescription+"' "+claimSuspensionDispositionDescription );
if( !ioceClaim.getClaimSuspensionEditList().isEmpty() ){
  final String editDispositionDescription=ioceComponent.getEditDispositionDescription( claimSuspensionDisposition,
internalVersion );

  for( final String edit : ioceClaim.getClaimSuspensionEditList() ){
    //Access edit value
    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
    System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
  }
}
```

```
//Retrieve line-level rejection disposition and edits generated by the component
final String lineRejectionDisposition=ioceClaim.getLineRejectionDisposition();
final String lineRejectionDispositionDescription=ioceComponent.getClaimDispositionDescription( "6", internalVersion );
final String lineRejectionDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "6", lineRejectionDisposition, internalVersion );
System.out.println( "Line Rejection Disposition = '"+lineRejectionDisposition+"' '"+lineRejectionDispositionValueDescription+"' "+lineRejectionDispositionDescription );
if( !ioceClaim.getLineRejectionEditList().isEmpty() ){
  final String editDispositionDescription=ioceComponent.getEditDispositionDescription( lineRejectionDisposition, internalVersion );
  for( final String edit : ioceClaim.getLineRejectionEditList() ){
    //Access edit value
    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
    System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
  }
}

//Retrieve line-level denial disposition and edits generated by the component
final String lineDenialDisposition=ioceClaim.getLineDenialDisposition();
final String lineDenialDispositionDescription=ioceComponent.getClaimDispositionDescription( "7", internalVersion );
final String lineDenialDispositionValueDescription=ioceComponent.getClaimDispositionValueDescription( "7", lineDenialDisposition, internalVersion );
System.out.println( "Line Denial Disposition = '"+lineDenialDisposition+"' '"+lineDenialDispositionValueDescription+"' "+lineDenialDispositionDescription );
if( !ioceClaim.getLineDenialEditList().isEmpty() ){
  final String editDispositionDescription=ioceComponent.getEditDispositionDescription( lineDenialDisposition, internalVersion );
  for( final String edit : ioceClaim.getLineDenialEditList() ){
    //Access edit value
    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
    System.out.println( "Disposition = '"+editDispositionDescription+"' Edit = '"+edit+"' "+editDescription );
  }
}

//Retrieve condition codes generated by the component
for( final String conditionCode : ioceClaim.getConditionCodeOutputList() ){
  //Access edit value
  //NOTE: condition code descriptions aren't maintained in the program
  System.out.println( "Input Condition Code = '"+conditionCode+"'" );
}

//Retrieve value codes generated by the component
for( final OceValueCode oceValueCode : ioceClaim.getValueCodeOutputList() ){
  final String code=oceValueCode.getCode();
  final String value=oceValueCode.getValue();
  //NOTE: value code descriptions aren't maintained in the program
  if( code!=null
  && !"".equals( code.trim() ) ){
    System.out.println( "Input Value Code = '"+code+"':'"+value+"'" );
  }
}
```

```java
                System.out.println();
                //Retrieve principal diagnosis edits generated by the component
                final String principalDiagnosisCodeDescription=ioceComponent.getDiagnosisDescription(
ioceClaim.getPrincipalDiagnosisCode().getDiagnosis(), internalVersion );
                System.out.println( "Principal Diagnosis = '"+ioceClaim.getPrincipalDiagnosisCode().getDiagnosis()+"' 
"+principalDiagnosisCodeDescription );
                if( !ioceClaim.getPrincipalDiagnosisCode().getEditList().isEmpty() ){
                  for( final String edit : ioceClaim.getPrincipalDiagnosisCode().getEditList() ){
                    //Access edit value
                    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                    System.out.println( "Edit = '"+edit+"' "+editDescription );
                  }
                }

                //Retrieve reason for visit diagnoses edits generated by the component
                if( !ioceClaim.getReasonForVisitDiagnosisCodeList().isEmpty() ){
                  for( final OceDiagnosisCode reasonForVisitDiagnosisCode : ioceClaim.getReasonForVisitDiagnosisCodeList() ){
                    //Retrieve secondary diagnoses descriptions
                    final String diagnosisCodeDescription=ioceComponent.getDiagnosisDescription(
reasonForVisitDiagnosisCode.getDiagnosis(), internalVersion );
                    System.out.println( "Reason for Visit Diagnosis = '"+reasonForVisitDiagnosisCode.getDiagnosis()+"' 
"+diagnosisCodeDescription );
                    if( !reasonForVisitDiagnosisCode.getEditList().isEmpty() ){
                      for( final String edit : reasonForVisitDiagnosisCode.getEditList() ){
                        //Access edit value
                        final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                        System.out.println( "Edit = '"+edit+"' "+editDescription );
                      }
                    }
                  }
                }

                //Retrieve secondary diagnoses edits generated by the component
                if( !ioceClaim.getSecondaryDiagnosisCodeList().isEmpty() ){
                  for( final OceDiagnosisCode secondaryDiagnosisCode : ioceClaim.getSecondaryDiagnosisCodeList() ){
                    //Retrieve secondary diagnoses descriptions
                    final String diagnosisCodeDescription=ioceComponent.getDiagnosisDescription(
secondaryDiagnosisCode.getDiagnosis(), internalVersion );
                    System.out.println( "Secondary Diagnosis = '"+secondaryDiagnosisCode.getDiagnosis()+"' 
"+diagnosisCodeDescription );
                    if( !secondaryDiagnosisCode.getEditList().isEmpty() ){
                      for( final String edit : secondaryDiagnosisCode.getEditList() ){
                        //Access edit value
                        final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                        System.out.println( "Edit = '"+edit+"' "+editDescription );
                      }
                    }
                  }
                }

                //Iterate over lines for line output
```

```java
                    for( final OceLineItem line : ioceClaim.getLineItemList() ){

                        System.out.println();

                        //Retrieve line-level output and descriptions generated by the component
                        final String actionFlagOutput=line.getActionFlagOutput();
                        final String actionFlagDescription=ioceComponent.getLineItemActionFlagDescription( actionFlagOutput,
internalVersion );
                        System.out.println( "Output Action Flag = '"+actionFlagOutput+"' "+actionFlagDescription );

                        final String rejectionDenialFlag=line.getRejectionDenialFlag();
                        final String rejectionDenialFlagDescription=ioceComponent.getLineItemRejectionDenialFlagDescription(
rejectionDenialFlag, internalVersion );
                        System.out.println( "Rejection Denial Flag = '"+rejectionDenialFlag+"' "+rejectionDenialFlagDescription );

                        final String paymentMethodFlag=line.getPaymentMethodFlag();
                        final String paymentMethodFlagDescription=ioceComponent.getPaymentMethodFlagDescription(
paymentMethodFlag, internalVersion );
                        System.out.println( "Payment Method Flag = '"+paymentMethodFlag+"' "+paymentMethodFlagDescription );

                        final String hcpcsApc=line.getHcpcsApc();
                        final String hcpcsApcDescription=ioceComponent.getApcDescription( hcpcsApc, internalVersion );
                        System.out.println( "Default CPT/HCPCS APC = '"+hcpcsApc+"' "+hcpcsApcDescription );

                        final String paymentApc=line.getPaymentApc();
                        final String paymentApcDescription=ioceComponent.getApcDescription( paymentApc, internalVersion );
                        System.out.println( "Payment APC = '"+paymentApc+"' "+paymentApcDescription );

                        final String unitsOutput=line.getUnitsOutput();
                        System.out.println( "Output Units = '"+unitsOutput+"'" );
                        //NOTE: no description for units values

                        final String statusIndicator=line.getStatusIndicator();
                        final String statusIndicatorDescription=ioceComponent.getStatusIndicatorDescription( statusIndicator,
internalVersion );
                        System.out.println( "Status Indicator = '"+statusIndicator+"' "+statusIndicatorDescription );

                        final String paymentIndicator=line.getPaymentIndicator();
                        final String paymentIndicatorDescription=ioceComponent.getPaymentIndicatorDescription( paymentIndicator,
internalVersion );
                        System.out.println( "Payment Indicator = '"+paymentIndicator+"' "+paymentIndicatorDescription );

                        final String packagingFlag=line.getPackagingFlag();
                        final String packagingFlagDescription=ioceComponent.getPackagingFlagDescription( packagingFlag, internalVersion
);
                        System.out.println( "Packaging Flag = '"+packagingFlag+"' "+packagingFlagDescription );

                        final String paymentAdjustmentFlag01=line.getPaymentAdjustmentFlag01();
                        final String paymentAdjustmentFlag01Description=ioceComponent.getPaymentAdjustmentFlagDescription(
paymentAdjustmentFlag01, internalVersion );
                        System.out.println( "Payment Adjustment Flag 01 = '"+paymentAdjustmentFlag01+"'
"+paymentAdjustmentFlag01Description );
```

```
                    final String paymentAdjustmentFlag02=line.getPaymentAdjustmentFlag02();
                    final String paymentAdjustmentFlag02Description=ioceComponent.getPaymentAdjustmentFlagDescription(
paymentAdjustmentFlag02, internalVersion );
                    System.out.println( "Payment Adjustment Flag 02 = '"+paymentAdjustmentFlag02+"'
"+paymentAdjustmentFlag02Description );

                    final String discountingFormula=line.getDiscountingFormula();
                    final String discountingFormulaDescription=ioceComponent.getDiscountFormulaDescription( discountingFormula,
internalVersion );
                    System.out.println( "Discounting Formula = '"+discountingFormula+"' "+discountingFormulaDescription );

                    final String compositeAdjustmentFlag=line.getCompositeAdjustmentFlag();
                    //NOTE: no description for incremental composite adjustment flag values
                    System.out.println( "Composite Adjustment Flag = '"+compositeAdjustmentFlag+"'" );

                    //Retrieve CPT/HCPCS modifiers generated by the component
                    for( final OceHcpcsModifier oceHcpcsModifier : line.getHcpcsModifierOutputList() ){
                        final String oceHcpcsModifierDescription=ioceComponent.getHcpcsModifierDescription(
oceHcpcsModifier.getHcpcsModifier(), internalVersion );
                        System.out.println( "Output CPT/HCPCS Modifier = '"+oceHcpcsModifier.getHcpcsModifier()+"'
"+oceHcpcsModifierDescription );
                    }

                    //Retrieve procedure edits generated by the component
                    for( final String edit : line.getHcpcsEditList() ){
                        //Access edit value
                        final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                        System.out.println( "Line Edit = '"+edit+"' "+editDescription );
                    }

                    //Retrieve revenue code edits generated by the component
                    for( final String edit : line.getRevenueEditList() ){
                        //Access edit value
                        final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                        System.out.println( "Revenue Code Edit = '"+edit+"' "+editDescription );
                    }

                    //Retrieve service date edits generated by the component
                    for( final String edit : line.getServiceDateEditList() ){
                        //Access edit value
                        final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                        System.out.println( "Service Date Edit = '"+edit+"' "+editDescription );
                    }

                    //Iterate over input CPT/HCPCS modifiers to retrieve modifier edits
                    for( final OceHcpcsModifier oceHcpcsModifier : line.getHcpcsModifierInputList() ){
                        final String oceHcpcsModifierDescription=ioceComponent.getHcpcsModifierDescription(
oceHcpcsModifier.getHcpcsModifier(), internalVersion );
                        System.out.println( "Input CPT/HCPCS Modifier = '"+oceHcpcsModifier.getHcpcsModifier()+"'
"+oceHcpcsModifierDescription );
                        for( final String edit : oceHcpcsModifier.getEditList() ){
```

```
                    //Access edit value
                    final String editDescription=ioceComponent.getEditDescription( edit, internalVersion );
                    System.out.println( "Input CPT/HCPCS Modifier Edit = '"+edit+"' "+editDescription );
                }
            }

        }

      }
    }
    catch( final FoundationException e ){
      e.printStackTrace( System.out );
    }
    catch( final Exception e ){
      e.printStackTrace( System.out );
    }

    System.out.println( "Completed." );
  }

  public static final void main( final String[] v ){
    final OceCallerExample oceCallerExample=new OceCallerExample();
    oceCallerExample.runOceCallerExample();
  }

}
```

Class Information

Class gov.cms.oce.IoceComponent

Constructor

IoceComponent()

Table 3. Method summary IoceComponent

Return type	Method	Description
void	process(IoceClaim claim) throws FoundationException	Process a claim.
String	getApcDescription(String value, short version)	Return description for ApcDescription parameter

Return type	Method	Description
String	getApcDescription(OceLineItem line, short version)	Return description for ApcDescription parameter
String	getLatestApcDescription(String value)	Return description for LatestApcDescription parameter
String	getLatestApcDescription(OceLineItem line)	Return description for LatestApcDescription parameter
String	getClaimDispositionDescription(String value, short version)	Return description for ClaimDispositionDescription parameter
String	getClaimDispositionDescription(OceClaim claim, short version)	Return description for ClaimDispositionDescription parameter
String	getLatestClaimDispositionDescription(String value)	Return description for LatestClaimDispositionDescription parameter
String	getLatestClaimDispositionDescription(OceClaim claim)	Return description for LatestClaimDispositionDescription parameter
String	getClaimDispositionValueDescription(String claimDispositionId, String value, short version)	Return description for ClaimDispositionValueDescription parameterString
String	getClaimProcessedFlagDescription(OceClaim claim, short version)	Return description for ClaimProcessedFlagDescription parameter
String	getLatestClaimProcessedFlagDescription(String value)	Return description for LatestClaimProcessedFlagDescription parameter
String	getLatestClaimProcessedFlagDescription(OceClaim claim)	Return description for LatestClaimProcessedFlagDescription parameter
String	getDiscountFormulaDescription(String value, short version)	Return description for DiscountFormulaDescription parameter
String	getDiscountFormulaDescription(OceLineItem line, short version)	Return description for DiscountFormulaDescription parameter

Return type	Method	Description
String	getLatestDiscountFormulaDescription(String value)	Return description for LatestDiscountFormulaDescription parameter
String	getLatestDiscountFormulaDescription(OceLineItem line)	Return description for LatestDiscountFormulaDescription parameter
String	getDiagnosisDescription(String value, short version)	Return description for DiagnosisDescription parameter
String	getLatestDiagnosisDescription(String value)	Return description for LatestDiagnosisDescription parameter
String	getEditDescription(String value, short version)	Return description for EditDescription parameter
String	getLatestEditDescription(String value)	Return description for LatestEditDescription parameter
String	getEditDispositionDescription(String value, short version)	Return description for EditDispositionDescription parameter
String	getLatestEditDispositionDescription(String value)	Return description for LatestEditDispositionDescription parameter
String	getErrorDescription(String value, short version)	Return description for ErrorDescription parameter
String	getLatestErrorDescription(String value)	Return description for LatestErrorDescription parameter
String	getHcpcsDescription(String value, short version)	Return description for HcpcsDescription parameter
String	getLatestHcpcsDescription(String value)	Return description for LatestHcpcsDescription parameter
String	getLineItemActionFlagDescription(String value, short version)	Return description for LineItemActionFlagDescription parameter
String	getLatestLineItemActionFlagDescription(String value)	Return description for LatestLineItemActionFlagDescription parameter

Return type	Method	Description
String	getLineItemDenialRejectionFlagDescription(String value, short version)	Return description for LineItemDenialRejectionFlagDescription parameter
String	getLatestLineItemDenialRejectionFlagDescription(String value)	Return description for LatestLineItemDenialRejectionFlagDescription parameter
String	getHcpcsModifierDescription(String value, short version)	Return description for HcpcsModifierDescription parameter
String	getLatestHcpcsModifierDescription(String value)	Return description for LatestHcpcsModifierDescription parameter
String	getPaymentAdjustmentFlagDescription(String value, short version)	Return description for PaymentAdjustmentFlagDescription parameter
String	getLatestPaymentAdjustmentFlagDescription(String value)	Return description for LatestPaymentAdjustmentFlagDescription parameter
String	getPaymentIndicatorDescription(String value, short version)	Return description for PaymentIndicatorDescription parameter
String	getLatestPaymentIndicatorDescription(String value)	Return description for LatestPaymentIndicatorDescription parameter
String	getPackagingFlagDescription(String value, short version)	Return description for PackagingFlagDescription parameter
String	getLatestPackagingFlagDescription(String value)	Return description for LatestPackagingFlagDescription parameter
String	getPaymentMethodFlagDescription(String value, short version)	Return description for PaymentMethodFlagDescription parameter
String	getLatestPaymentMethodFlagDescription(String value)	Return description for LatestPaymentMethodFlagDescription parameter

Return type	Method	Description
String	getRevenueCodeDescription(String value, short version)	Return description for RevenueCodeDescription parameter
String	getLatestRevenueCodeDescription(String value)	Return description for LatestRevenueCodeDescription parameter
String	getStatusIndicatorDescription(String value, short version)	Return description for StatusIndicatorDescription parameter
String	getLatestStatusIndicatorDescription(String value)	Return description for LatestStatusIndicatorDescription parameter
String	getValueCodeDescription(String value, short version)	Return description for ValueCodeDescription parameter
String	getLatestValueCodeDescription(String value)	Return description for LatestValueCodeDescription parameter

Class gov.cms.oce.IoceClaim

Extends

com.mmm.his.cer.foundation.transfer.IClaim

Constructors

IoceClaim(gov.cms.oce.external.OceClaim)

Table 4. Method summary IoceClaim

Return Type	Method	Description
OceClaim	getModel()	Returns the wrapped model interface

Interface gov.cms.oce.model.external.OceClaim

Table 5. Method summary OceClaim

Return Type	Method	Description
void	setClaimId(String value)	Set claim identifier
String	getClaimId()	Return claim identifier
void	setAge(String value)	Set age
String	getAge()	Return age
void	setSex(String value)	Set sex
String	getSex()	Return sex
void	setDateStarted(String value)	Set date started
String	getDateStarted()	Return date started
void	setDateEnded(String value)	Set date ended
String	getDateEnded()	Return date ended
void	setBillType(String value)	Set bill type
String	getBillType()	Return bill type
void	setPatientStatus(String value)	Set patient discharge status
String	getPatientStatus()	Return patient discharge status
void	setOppsFlag(String value)	Set OPPS flag
String	getOppsFlag()	Return OPPS flag
void	setNationalProviderId(String value)	Set national provider identifier
String	getNationalProviderId()	Return national provider identifier
void	setCmsCertificationNumber(String value)	Set CMS certification number
String	getCmsCertificationNumber()	Return CMS certification number
void	addOccurrenceCodeInput(String value)	Add an occurrence code

Return Type	Method	Description
void	setOccurrenceCodeInput(int index, String value)	Replace the occurrence code at the index position with the value parameter.
int	getOccurrenceCodeInputListLength()	Return number of occurrence codes
List<String>	getOccurrenceCodeInputList()	Return list of occurrence codes
void	addConditionCodeInput(String value)	Add a condition code
void	setConditionCodeInput(int index, String value)	Replace the condition code at the index position with the value parameter.
int	getConditionCodeInputListLength()	Return number of condition codes
List<String>	getConditionCodeInputList()	Return list of condition codes
void	addValueCodeInput(String code, String value)	Add a value code and value
void	addValueCodeInput(OceValueCode oceValueCode)	Add a value code object
void	setValueCodeInput(int index, String code, String value)	Replace the value code at the index position with the value parameter.
void	setValueCodeInput(int index, OceValueCode oceValueCode)	Replace the value code at the index position with the object parameter.
int	getValueCodeInputListLength()	Return number of value codes
List<OceValueCode>	getValueCodeInputList()	Return list of value codes
void	setReceiptDate(String value)	Set receipt date
String	getReceiptDate()	Return receipt date
void	addReasonForVisitDiagnosisCode(String code, String presentOnAdmission)	Add the diagnosis code

Return Type	Method	Description
void	addReasonForVisitDiagnosisCode(OceDiagnosisCode oceDiagnosisCode)	Add the diagnosis code object
void	setReasonForVisitDiagnosisCode(int index, String code, String presentOnAdmission)	Replace the diagnosis code at the index position with the diagnosis code and POA parameters.
void	setReasonForVisitDiagnosisCode(int index, OceDiagnosisCode oceDiagnosisCode)	Replace the diagnosis code at the index position with the diagnosis code object parameter.
int	getReasonForVisitDiagnosisCodeListLength()	Return number of reason for visit diagnosis codes
List<OceDiagnosisCode>	getReasonForVisitDiagnosisCodeList()	Return list of diagnosis codes
void	addSecondaryDiagnosisCode(String code, String presentOnAdmission)	Add secondary diagnosis code
void	addSecondaryDiagnosisCode(OceDiagnosisCode oceDiagnosisCode)	Add the diagnosis code object
void	setSecondaryDiagnosisCode(int index, String code, String presentOnAdmission)	Replace the diagnosis code at the index position with the diagnosis code and POA parameters.
void	setSecondaryDiagnosisCode(int index, OceDiagnosisCode oceDiagnosisCode)	Replace the diagnosis code at the index position with the diagnosis code object parameter.
int	getSecondaryDiagnosisCodeListLength()	Return number of secondary diagnosis objects
List<OceDiagnosisCode>	getSecondaryDiagnosisCodeList()	Return list of secondary diagnosis objects
void	setPrincipalDiagnosisCode(String code, String presentOnAdmission)	Add principal diagnosis code

Return Type	Method	Description
void	setPrincipalDiagnosisCode(OceDiagnosisCode oceDiagnosisCode)	Add principal diagnosis code
OceDiagnosisCode	getPrincipalDiagnosisCode()	Returns principal diagnosis code object
void	addLineItem(OceLineItem)	Add the line item object parameter to claim

void	addLineItem(
	  String actionFlagInput,
	  String serviceDate,
	  String revenueCode,
	  String hcpcs,
	  List<OceHcpcsModifier> hcpcsModifierInputList,
	  String units,
	  String charge,
	  List<String> contractorEditBypassList,
	  String contractorApc,
	  String contractorStatusIndicator,
	  String contractorPaymentIndicator,
	  String contractorDiscountingFormula,
	  String contractorRejectionDenialFlag,
	  String contractorPackagingFlag,
	  String contractorPaymentAdjustmentFlag01,
	  String contractorPaymentMethodFlag,
	  String contractorPaymentAdjustmentFlag02
	)
		Add line item object to claim using the supplied parameters.

Return Type	Method	Description
void	setLineItem(int index, OceLineItem oceLineItem)	Replace the line item object at the index position with the line item object parameter.

void	setLineItem(
	  int index,
	  String actionFlagInput,
	  String serviceDate,
	  String revenueCode,
	  String hcpcs,
	  List<OceHcpcsModifier> hcpcsModifierInputList,
	  String units,
	  String charge,
	  List<String> contractorEditBypassList,
	  String contractorApc,
	  String contractorStatusIndicator,
	  String contractorPaymentIndicator,
	  String contractorDiscountingFormula,
	  String contractorRejectionDenialFlag,
	  String contractorPackagingFlag,
	  String contractorPaymentAdjustmentFlag01,
	  String contractorPaymentMethodFlag,
	  String contractorPaymentAdjustmentFlag02
	)
		Replace the line item object at the index position using the supplied parameters.

int	getLineItemListLength()	Return number of line items

Return Type	Method	Description
List<OceLineItem>	getLineItemList()	Return list of line item objects
ProcessingInformation	getProcessingInformation()	Returns the processing information object
String	getVersion()	Returns software version that processed claim
String	getClaimProcessedFlag()	Returns claim processed flag
String	getApcReturnBufferFlag()	Returns APC return buffer flag
String	getNoppsBillFlag()	Returns Non-OPPS bill flag
String	getClaimDisposition()	Returns claim disposition
String	getClaimRejectionDisposition()	Returns claim rejection disposition
List<String>	getClaimRejectionEditList()	Returns claim rejection edits
String	getClaimDenialDisposition()	Returns claim denial disposition
List<String>	getClaimDenialEditList()	Returns claim denial edits
String	getClaimReturnToProviderDisposition()	Returns claim return to provider disposition
List<String>	getClaimReturnToProviderEditList()	Returns claim return to provider edits
String	getClaimSuspensionDisposition()	Returns claim suspension disposition
List<String>	getClaimSuspensionEditList()	Returns claim suspension edits
String	getLineRejectionDisposition()	Returns line rejection disposition
List<String>	getLineRejectionEditList()	Returns claim rejection edits
String	getLineDenialDisposition()	Returns line denial disposition
List<String>	getLineDenialEditList()	Returns line denial edits

Return Type	Method	Description
List<String>	getConditionCodeOutputList()	Returns condition codes generated by the component
List<OceValueCode>	getValueCodeOutputList()	Returns value code objects generated by the component
OceDiagnosisCode	getPrincipalDiagnosisCode()	Returns the principal diagnosis object
List<OceDiagnosisCode>	getSecondaryDiagnosisCodeList()	Returns list of secondary diagnoses objects
List<OceDiagnosisCode>	getReasonForVisitDiagnosisCodeList()	Returns list of reason for visit diagnoses objects

Interface gov.cms.oce.model.external.OceLineitem

Table 6. Method summary OceLineItem

Return Type	Method	Description
void	setServiceDate(String value)	Set date of service
void	setRevenueCode(String value)	Set revenue code
void	setHcpcs(String value)	Set CPT/HCPCS
void	addHcpcsModifierInput(String value)	Set CPT>HCPCS modifier
void	setUnitsInput(String value)	Set units
void	setCharge(String value)	Set submitted line charge
void	setActionFlagInput(String value)	Set action flag
void	addContractorEditBypass(String value)	Add an applicable edit to bypass
void	setContractorApc(String value)	Add APC override value
void	setContractorStatusIndicator(String value)	Add status indicator override value
void	setContractorPaymentIndicator(String value)	Add payment indicator override value

Return Type	Method	Description
void	setContractorDiscountingFormula(String value)	Add discounting formula override value
void	setContractorRejectionDenialFlag(String value)	Add rejection denial flag override value
void	setContractorPackagingFlag(String value)	Add packaging flag override value
void	setContractorPaymentAdjustmentFlag01(String value)	Add payment adjustment flag override value
void	setContractorPaymentMethodFlag(String value)	Add payment method flag override value
void	setServiceDate(String value)	Set date of service
String	getActionFlagOutput()	Returns action flag output
String	getRejectionDenialFlag()	Returns rejection denial flag
String	getPaymentMethodFlag()	Returns payment method flag
String	getHcpcsApc()	Returns CPT/HCPCS APC
String	getPaymentApc()	Returns payment APC
String	getUnitsOutput()	Returns units output
String	getStatusIndicator()	Returns status indicator
String	getPaymentIndicator()	Returns payment indicator
String	getPackagingFlag()	Returns packaging flag
String	getPaymentAdjustmentFlag01()	Returns payment adjustment flag01
String	getPaymentAdjustmentFlag02()	Returns payment adjustment flag02
String	getDiscountingFormula()	Returns discountingm formula
String	getCompositeAdjustmentFlag()	Returns composite adjustment flag
List<OceHcpcsModifier>	getHcpcsModifierInputList()	Returns list of CPT/HCPCS modifier objects passed into the component
List<OceHcpcsModifier>	getHcpcsModifierOutputList()	Returns list of CPT/HCPCS modifier objects generated by the component

Return Type	Method	Description
List<String>	getHcpcsEditList()	Returns line-level edits
List<String>	getRevenueEditList()	Returns revenue code edits
List<String>	getServiceDateEditList()	Returns service date edits

Interface gov.cms.oce.model.external.OceDiagnosisCode

Table 7. Method summary OceDiagonsisCode

Return Type	Method	Description
void	setDiagnosis(String value)	Set diagnosis code
String	getDiagnosis()	Return diagnosis code
void	setPresentOnAdmission(String value)	Set present on admission value
String	getPresentOnAdmission()	Return present on admission value
List<String>	getEditList()	Returns edit values associated with this diagnosis code.

Interface gov.cms.oce.model.external.OceHcpcsModifier

Table 8. Method summary OceHcpcsModifier

Return Type	Method	Description
void	setHcpcsModifier(String value)	Set CPT/HCPCS modifier
String	getHcpcsModifier()	Return CPT/HCPCS modifier
List<String>	getEditList()	Returns edit values associated with this diagnosis code.

Interface gov.cms.oce.model.external.OceValueCode

Table 9. Method summary OceValueCode

Return Type	Method	Description
void	setCode(String value)	Set value code
String	getCode()	Return value code
void	setValue(String value)	Set value (amount)
String	getValue()	Return value (amount)

Interface gov.cms.oce.model.external.OceProcessingInformation

Table 10. Method summary OceProcessingInformation

Return Type	Method	Description
void	setClaimId(String value)	Set claim identifier
String	getClaimId()	Return claim identifier
void	setDebugFlag(String value)	Set debug flag
String	getDebugFlag()	Return debug flag
void	setCommentData(String value)	Set comment data
String	getCommentData()	Return comment data
Integer	getReturnCode()	Return code
Integer	getLinesProcessed()	Return lines processed
String	getVersion()	Return version string
long	getTimeStarted()	Return time processing started
long	getTimeEnded()	Return time processing ended

Class gov.cms.oce.model.external.OceClaimFactory

Table 11. Method summary OceClaimFactory

Return Type	Method	Description
OceClaimFactory	getInstance()	Returns the single instance of this factory
OceClaim	createClaim()	Returns an empty OceDiagnosisCode object
OceDiagnosisCode	createDiagnosisCode()	Returns an empty OceDiagnosisCode object
OceDiagnosisCode	createDiagnosisCode(String diagnosis, String presentOnAdmission)	Returns an OceDiagnosisCode object given the parameter values
OceHcpcsModifier	createHcpcsModifier()	Returns an empty OceHcpcsModifier object
OceHcpcsModifier	createHcpcsModifier(String value)	Returns an OceHcpcsModifier object given the parameter values
OceLineItem	createLineItem()	Returns an empty OceLineItem object

Return Type	Method	Description

OceLineItem	createLineItem(
	
	  String actionFlagInput,
	
	  String serviceDate,
	
	  String revenueCode,
	
	  String hcpcs,
	
	  List<OceHcpcsModifier> hcpcsModifierInputList,
	
	  String units,
	
	  String charge,
	
	  List<String> contractorEditBypassList,
	
	  String contractorApc,
	
	  String contractorStatusIndicator,
	
	  String contractorPaymentIndicator,
	
	  String contractorDiscountingFormula,
	
	  String contractorRejectionDenialFlag,
	
	  String contractorPackagingFlag,
	
	  String contractorPaymentAdjustmentFlag01,
	
	  String contractorPaymentMethodFlag,
	
	  String contractorPaymentAdjustmentFlag02
	
	)
		Returns an OceHcpcsModifier object given the parameter values

OceValueCode	createValueCode()	Returns an empty OceDiagnosisCode object
OceValueCode	createValueCode(String code, String value)	Returns an OceValueCode object given the parameter values