

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Migración Interfaces SSiCC a NSSiCC

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Tabla de contenido

1. Objetivos.....	4
2. Consideraciones Previas	5
2.1. SSiCC	5
2.2. Nomenclatura	6
2.3. Estructura	6
2.4. Modularidad del NSSiCC	7
2.6. Carpeta target del NSSiCC	9
2.7. Uso de Anotaciones.....	9
2.8. Variables de Sesión.....	9
2.9. Ubicación Clase Action	10
2.10. Métodos Bases.....	11
2.10.1. setViewAttributes:.....	11
2.10.2. setFindAttributes:	11
2.11. Uso de AJAX.....	11
2.12. Páginas WEB y templates.....	12
2.13. Componentes Personalizados	12
2.13.1. Etiqueta asociada a la caja de texto	12
2.13.2. Caja de Texto	13
2.13.3. Campo Fechas	14
2.13.4. Campo Periodo ó Campaña.....	14
2.13.5. Combox o Lista (Selección Única)	15
2.13.6. Combox o Lista (Selección Múltiple)	16
2.13.7. Campo TextArea	17
2.13.8. CheckBox	17
2.13.9. RadioButton	18
2.13.10. Grilla Datatable (Modo Consulta o de Selección simple)	18
2.13.11. CommandButton	20
2.14. Ubicación archivos XML de ibatis.....	20
2.15. Reportes.....	21
2.16. Asignando Página xhtml al Menú.....	22
2.17. Pruebas Internas	23
3. Migración de Interfaces	25
3.1. Conceptos Previos	25
3.1.1. Visualización Interfaz de Salida tipo Paquete	26
3.2. Verificando el Tipo de Interfaz	28

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3.3.	Creando Clase Form en el NSSiCC	29
3.4.	Creando Clase Action en el NSSiCC	29
3.5.	Variables en el método setViewAtributtes	30
3.6.	Revisando invocaciones a Clase Service y DAO	31
3.7.	Eliminar try catch	31
3.8.	Verificar método Reset en la Clase Form	31
3.9.	Creando página xhtml	31
3.10.	Agregando los Filtros en página xhtml.....	33
3.11.	Campos Fechas	35
3.12.	Campos Periodos que invoquen Ajax.....	35
3.13.	Revisar código javascript en la página jsp del SSiCC.....	35
3.14.	Creando Clase Service (Caso Especial solo para Interfaz de Tipo Paquete)	36
3.15.	Modificando Clase Service (Caso Especial solo para Interfaz Unitaria)	37
3.16.	Verificando Lista de Interfaces (Solo para Interfaz de Tipo Paquete).....	38
3.17.	Realizando pruebas internas (Solo para Interfaz de Salida).....	40
3.18.	Realizando pruebas internas (Solo para Interfaz de Entrada)	40

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

1. Objetivos

El presente documento tiene como finalidad servir como Documento Base para la migración del Código Fuente correspondiente a las Pantallas de Interfaces del SSiCC al Nuevo SSiCC

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

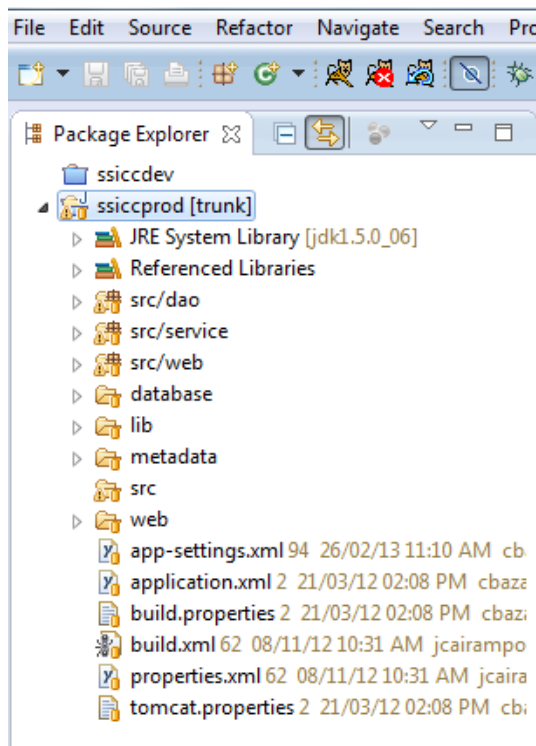
2. Consideraciones Previas

2.1. SSiCC

Debemos ingresar al proyecto SSiCC bajo el Eclipse configurado para dicho sistema y en base proyecto ssicccprod debemos realizar la migración respectiva.

Dicho proyecto se encuentra en el siguiente repositorio:

<https://pents73.galileo.ebel:8082/svn/ssicc/trunk>



Para acceder al Sistema SSiCC ingresar a la siguiente ruta (**Ejecutarlo solo bajo Internet Explorer en Modo de Compatibilidad**)

<http://pelinx3196:7777/ssicccpeesdev/login.jsp>

Usuario: ADMIN

Password: BELCORP

País: PERU ESIKA

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



2.2. Nomenclatura

Las clases a migrar en el NSSiCC deberán respetar la nomenclatura existente en el SSiCC. Una excepción son las páginas WEB que en el SSiCC son de extensión jsp, mientras que en el NSSiCC se le colocará con el mismo nombre, pero con **extensión xhtml**

2.3. Estructura

El SSiCC trabaja bajo una estructura propietaria manejada por ANT para la generación del WAR respectivo, mientras NSSiCC se adecua a la estructura establecida para los proyectos JAVA generados por MAVEN

Debido a lo mencionado el SSiCC tiene la siguiente estructura (Existe excepciones en algunos casos):

- Clase Java de Capa WEB (Action):
biz.belcorp.ssicc.<modulo>.web.action;
- Clase Java de Capa WEB (Form):
biz.belcorp.ssicc.<modulo>.web.form;
- Clase Java de Capa Service (Interface):
biz.belcorp.ssicc.<modulo>.service;
- Clase Java de Capa Service (Implementación):
biz.belcorp.ssicc.<modulo>.service.impl;

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

- Clase Java de Capa DAO (Interface):
biz.belcorp.ssicc.<modulo>.dao;
- Clase Java de Capa DAO (Implementación):
biz.belcorp.ssicc.<modulo>.dao.ibatis;
- Clase Java de Capa DAO (Pojo):
biz.belcorp.ssicc.<modulo>.dao.model;
- Archivo XML de Capa DAO:
biz / belcorp / ssicc / <modulo> / dao / ibatis / sql

En el NSSiCC varia la estructura de archivos debido que debe cumplir con el estándar de estructura establecido por MAVEN:

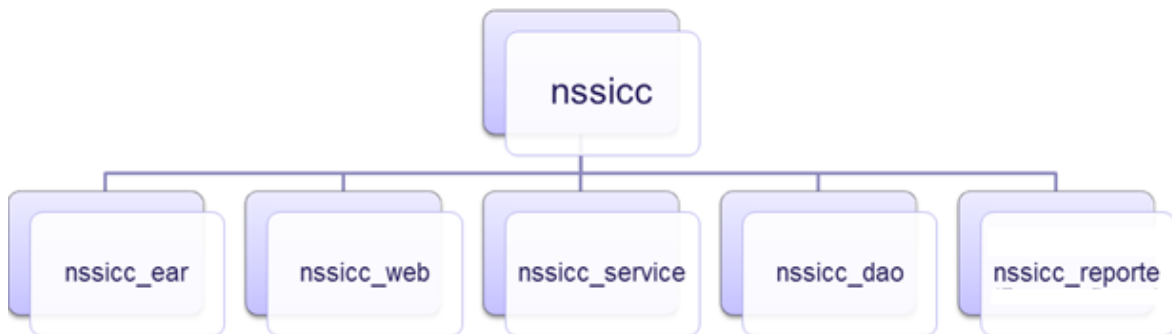
- Clase Java de Capa WEB (ManageBeans → Reemplaza al Action):
biz.belcorp.ssicc.web.<modulo>.action;
- Clase Java de Capa WEB (Form):
biz.belcorp.ssicc.web.<modulo>.form;
- Clase Java de Capa Service (Interface):
biz.belcorp.ssicc.service.<modulo>;
- Clase Java de Capa Service (Implementación):
biz.belcorp.ssicc.service.<modulo>.impl;
- Clase Java de Capa DAO (Interface):
biz.belcorp.ssicc.dao.<modulo>;
- Clase Java de Capa DAO (Implementación):
biz.belcorp.ssicc.dao.<modulo>.ibatis;
- Clase Java de Capa DAO (Pojo):
biz.belcorp.ssicc.dao.<modulo>.model;
- Archivo XML de Capa DAO:
biz / belcorp / ssicc / dao / <modulo> / ibatis / sql (Se encuentran a nivel de la carpeta resources)

2.4. Modularidad del NSSiCC

Podrán apreciar que al ver el proyecto en el Eclipse se verá varios proyectos. Sin embargo no son proyectos separados, sino son subproyectos del proyecto padre que es el nssicc.

Se ha subdivido el proyecto bajo los siguientes subproyectos:

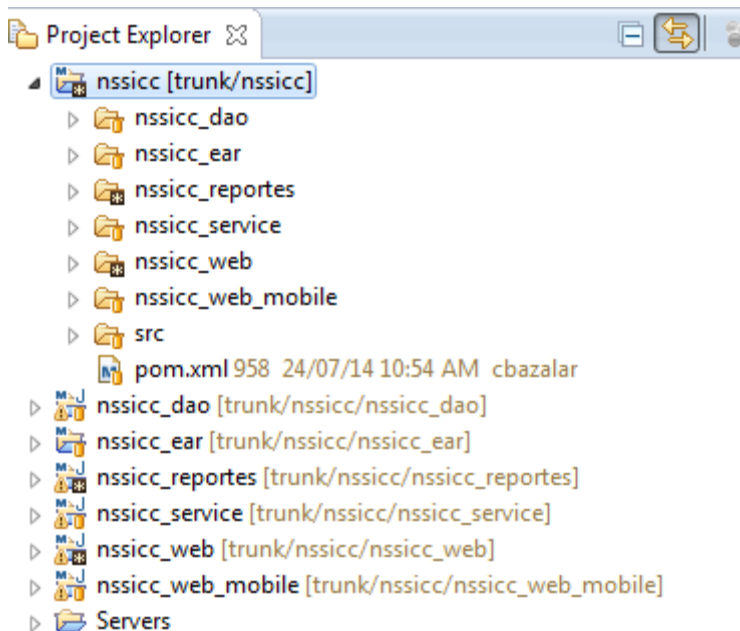
	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



Donde:

- nssicc → Proyecto Padre
- nssicc_web → Proyecto correspondiente a la Capa Web
- nssicc_service → Proyecto correspondiente a la Capa Service
- nssicc_dao → Proyecto correspondiente a la Capa DAO
- nssicc_reportes → Proyecto que contendrá los reportes del Sistema
- nssicc_ear → Proyecto que genera EAR del proyecto
- nssicc_web_mobile → Proyecto correspondiente a la Capa Web (Mobile)

Debido a lo mencionado en el eclipse se verá de la siguiente forma:



	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.6. Carpeta target del NSSiCC

Al momento de compilar las clases del NSSiCC genera la carpeta target con los archivos class generados. Dicha carpeta NO DEBERA ser sincronizada al Repositorio cuando realicen el desarrollo respectivo. Para ello deben indicar al Subversión que dicha carpeta deberá ser ignorada al momento de querer subir los cambios al Repositorio de código fuente (Ver Manual de Instalación capítulo 6.2)

Asimismo en dicha carpeta target se vuelve a grabar las paginas xhtml desarrolladas por uno.

Verificar siempre que al momento de realizar cambios en la pantalla xhtml NO se realice sobre la versión que se encuentre en la versión target, ya que luego esos cambios se perderán

2.7. Uso de Anotaciones

El NSSiCC trabaja a nivel de anotaciones tanto para el mapeo de los Manage Beans como las clases Java Spring de Capa Service y de capa DAO.

En el SSiCC las clases java Spring de Capa Service como de capa DAO se mapean en los archivos applicationContext*.xml

Tener en consideración al momento de realizar la migración de las Clases Spring que debe colocarse el id del beans en formato de anotación (Existen algunas excepciones como el mapeo de interfaces)

2.8. Variables de Sesión

El SSiCC hace bastante uso de la Session para colocar variables en ella, generalmente las listas y combobox mostradas en las pantallas.

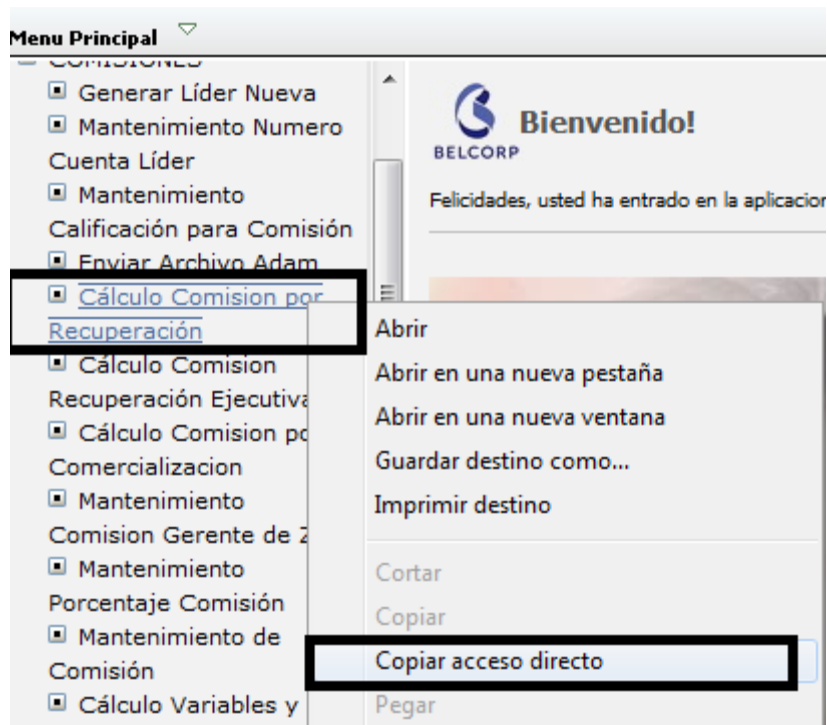
El NSSiCC no trabaja con variables de Session, para ello al momento de migrar dichas variables del SSiCC, deberán ser creadas en el NSSiCC como nuevos atributos del Manage Beans.

De esta forma evitamos el uso de la memoria de Sesión, ya que las variables quedan encapsuladas como atributos de la Clase ManageBeans.

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.9. Ubicación Clase Action

Las opciones del Menú del SSiCC traen consigo una URL. Para ello al momento de migrar una opción del menú hacer click botón derecho en la opción para obtener el enlace respectivo



Por ejemplo si quisiéramos migrar la opción "Cálculo Comisión por Recuperación" obtendremos la siguiente URL:

<http://pelnx3196:7777/ssicpeesdev/viewProcesoCOMComisionRecuperacion.do?codigoSistema=COM&codigoProcesoBatch=01&codigoMenu=20010708>

Observamos que la siguiente cadena:

viewProcesoCOMComisionRecuperacion.do

Por convención del SSiCC la palabra view corresponde al método View que va invocar la cual se encontrará en la Clase Java Action **ProcesoCOMComisionRecuperacion**

Por tanto la base para comenzar a migrar será revisar la clase **ProcesoCOMComisionRecuperacion.java**

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.10. Métodos Bases

Como hemos visto en el punto anterior todas las opciones del Sistema SSiCC (salvo algunas clases Action que se crearon al inicio del desarrollo del SSiCC) al hacer click en la opción del menú invocan al Método View de la clase Java Struts Action asociada, la cual realizara la carga de data inicial que se mostrará al ejecutar dicha pantalla.

Asi de la misma forma existe métodos que se utilizar para funcionalidades específicas las cual comentamos las principales:

2.10.1. *setViewAttributes:*

Método que se encontrará en todas las clases Java Action la cual es aquí donde se pondrá el código que se necesitará para mostrar la data a mostrar al momento de ejecutar la pantalla desde el menú del Sistema

2.10.2. *setFindAttributes:*

Método que devuelve una Lista con la información mostrada en la Grilla de Resultados. Dicho método es utilizado al hacer Búsqueda de Mantenimientos y Consultas

Asimismo también es invocado en ciertas Interfaces que previamente deben buscar y generar una grilla de resultados

2.11. Uso de AJAX

El SSiCC debido que usa Struts 1.x, no dispone de AJAX Nativo, para ello usa DWR para poder hacer efecto de Ajax, la cual básicamente se usa en los combos anidados.

Para el uso de AJAX en DWR, utiliza javascript para invocar a una clase definida en Spring (AjaxService.java y AjaxServiceImpl.java) para poder invocar a los métodos establecidos. Asimismo dichos métodos deben estar definidos en el archivo de configuración dwr.xml

El NSSiCC usa tecnología nativa de JSF2, debido a ello, **NO usa Javascript para el uso de AJAX**. Al momento de migrar se deberá revisar el código javascript a migrar y si hacen uso del DWR, **deberá ser migrado usando el atributo update del componente y creando el método ActionListener en la clase ManageBeans respectiva (Clase Action Migrada)**

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.12. Páginas WEB y templates

El SSiCC maneja archivos jsp para las pantallas WEB mientras el NSSiCC maneja archivos xhtml para las pantallas WEB. Cuando se realice la migración de las pantallas JSP del SSiCC se deberá crear en el NSSiCC el mismo archivo pero con extensión xhtml y se deberá crear en base al template asociado.

El SSiCC no maneja template padre para las paginas jsp. Eso quiere decir que al momento de crear una página jsp similar a otra ya existente, actualmente se copia toda la página a la nueva y en la nueva se hacen los cambios.

En el NSSiCC si maneja template padre para cada tipo de página WEB a desarrollar (Ya sea un Mantenimiento, Interfaz, Consulta, Reporte, Proceso Batch, etc)

Debido a lo mencionado al momento de migrar la página jsp del SSiCC a su análogo xhtml en NSSiCC **NO se deberá copiar todo el código**, sino primero crear el archivo en base al template respectivo y **SOLO pasar las partes que son propias a la página** (Ej: Los filtros de búsqueda, ingreso de campos de mantenimiento, etc)

2.13. Componentes Personalizados

El NSSiCC trata de no utilizar directamente los componentes que disponemos de JSF2 y de Primefaces para los componentes comunes. Para ello hace uso de los componentes personalizado que se han creado

Debido a lo mencionado se deberá utilizar siempre estos componentes para estos casos (puede haber excepciones las cuales deberán ser consultadas previamente para su implementación):

2.13.1. Etiqueta asociada a la caja de texto

Se deberá utilizar el componente `ssicc:label`

Ejemplo:

```
<ssicc:label key="#{i18n['interfazSearchForm.descripcionSistema']}" />
```

En caso la caja de texto asociada su valor sea obligatorio, se deberá colocar en el componente `ssicc:label` el atributo `required="true"`

Ejemplo:

```
<ssicc:label id="labelPais" key="#{i18n['interfazForm.codigoPais']}"
required="true"/>
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:label`

Ejemplo: Si el id es labelPais

Desde javascripts: → `document.getElementById("labelPais:label")`

Desde otro componente → `:principalForm:labelPais:label`

2.13.2. Caja de Texto

Se deberá utilizar el componente `ssicc:inputText`

Ejemplo:

```
<ssicc:inputText id="descripcion" value="#{mAlias.formMantenimiento.descripcion}"
size="100" maxlength="300" />
```

En caso la caja de texto tenga valor que sea obligatorio colocar el atributo `required="true"`

```
<ssicc:inputText id="descripcion" value="#{mAlias.formMantenimiento.descripcion}"
size="100" maxlength="300" required="true" />
```

Por defecto este componente todo lo que se escriba lo pasa a mayúsculas, en caso se desee poder colocar valores tanto mayúsculas como minúsculas colocar el atributo `esMayuscula="false"`

Ejemplo:

```
<ssicc:inputText id="descripcion" value="#{mAlias.formMantenimiento.descripcion}"
size="100" maxlength="300" required="true" esMayuscula="false" />
```

Asimismo si solo se desea que ingrese valores Numéricos colocar el atributo `esNumerico="true"`

Ejemplo:

```
<ssicc:inputText id="anho" value="#{mAlias.formBusqueda.anho}" size="7"
maxlength="4" required="true" esNumerico="true" />
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:input`
 Ejemplo: Si el id es anho
 Desde javascripts: → `document.getElementById("anho:input")`
 Desde otro componente → `:principalForm:anho:input`

2.13.3. Campo Fechas

Para las cajas de texto que se ingrese valores de tipo Fecha se deberá colocar el componente `ssicc:calendar`
 Ejemplo:

```
<ssicc:calendar id="fecIni" value="#{mAlias.formReporte.fechaInicio}"/>
```

Consideración: El atributo del Manage Bean asociado a dicho componente deberá ser de tipo `java.util.Date` y no `String`. Si es `String` generará ERROR



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:calendar`
 Ejemplo: Si el id es fecIni
 Desde javascripts: → `document.getElementById("fecIni:calendar")`
 Desde otro componente → `:principalForm:fecIni:calendar`

2.13.4. Campo Periodo ó Campaña

El SSiCC maneja el concepto de Periodo (también denominado Campaña), la cual consiste en la división del año por un grupo de días (aprox 20 días). En el año se han establecido 18 campañas, y podemos ver que existen muchas pantallas del SSiCC, donde se pide un valor de Campaña (Ejemplo: 201405)

Para cuando la pantalla pida este tipo de valores se deberá colocar el componente `ssicc:campaignCalendar2`

Ejemplo:

```
<ssicc:campaignCalendar2 id="codigoPeriodo" required="true" manage="#{mAlias}" value="#{mAlias.formReporte.codigoPeriodo}"/>
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando <nombreFormulario>:<id>:campanya
Ejemplo: Si el id es codigoPeriodo
Desde javascripts: → document. getElementById ("codigoPeriodo: campanya")
Desde otro componente → :principalForm:codigoPeriodo:campanya

2.13.5. Combox o Lista (Selección Única)

Las lista o comboBox cuya selección sea un valor único debe utilizar el componente

ssicc:selectMenu

Ejemplo:

En este caso trae una lista obtenida en el ManageBeans:

```
<ssicc:selectMenu id="codigoComision" value="#{mAlias.formReporte.codigoComision}"
required="true">

    <f:selectItems value="#{mAplicacionBean.listaComision}"
        var="lista"
        itemLabel="#{lista.descripcion}" itemValue="#{lista.codigo}"/>
</ssicc:selectMenu>
```

En este caso trae una lista obtenida en el ManageBeans y el 1er Registro es un valor en blanco:

```
<ssicc:selectMenu id="codigoComision" value="#{mAlias.formReporte.codigoComision}"
required="true">
    <f:selectItem itemLabel="" itemValue=""/>
    <f:selectItems value="#{mAplicacionBean.listaComision}"
        var="lista"
        itemLabel="#{lista.descripcion}" itemValue="#{lista.codigo}"/>
</ssicc:selectMenu>
```

Si se desea invocar a un método y refrescar usando AJAX:

Ejemplo:

```
<ssicc:selectMenu id="codigoCanal"
value="#{mAlias.formReporte.codigoCanal}" required="true"
accionListener="#{reporteINCPuntObtenidosPuntFaltantesAction.LoadRegionesCanal}"
update=":principalForm:codigoRegion:selectMany :principalForm:codigoZona:selectMany
:principalForm:codigoConcurso:select">
    <f:selectItem itemLabel="" itemValue=""/>
    <f:selectItems value="#{mAlias.siccCanalList}"
        var="lista" itemLabel="#{lista.descripcion}" itemValue="#{lista.codigo}"/>
</ssicc:selectMenu>
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando <nombreFormulario>:<id>:select
Ejemplo: Si el id es codigoCanal
Desde javascripts: → document. getElementById ("codigoCanal:select")
Desde otro componente → :principalForm:codigoCanal:select

2.13.6. Combox o Lista (Selección Múltiple)

Aquellas listas o combobox donde se pueda seleccionar más de un valor (Selección Múltiple) deberá utilizarse el componente **ssicc: selectManyMenu**

Ejemplo:

```
<ssicc:selectManyMenu id="nivelConcurso"
    value="#{mAlias.formReporte.nivelConcursoList}" required="true">

    <f:selectItems value="#{mAlias.siccNivelesConcursoList}"
        var="Lista"
        itemLabel="#{lista.Label}" itemValue="#{lista.value}"/>
</ssicc:selectManyMenu>
```

Si se desea invocar a un método y refrescar usando AJAX:

Ejemplo:

```
ssicc:selectManyMenu id="codigoRegion"
    value="#{mAlias.formReporte.codigoRegion}" required="true"
    accionListener="#{reporteINCPuntObtenidosPuntFaltantesAction.showZonasxRegion}"
    update=":principalForm:codigoZona:selectMany">
    <f:selectItems value="#{mAlias.siccRegionList}"
        var="ListaRegion" itemLabel="#{listaRegion.Label}"
        itemValue="#{listaRegion.value}"/>
</ssicc:selectManyMenu>
```



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando <nombreFormulario>:<id>:select
Ejemplo: Si el id es codigoRegion
Desde javascripts: → document. getElementById ("codigoRegion:select")
Desde otro componente → :principalForm:codigoRegion:select

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.13.7. Campo TextArea

Se deberá utilizar el componente `ssicc:inputTextArea`

Ejemplo:

```
<ssicc:inputTextArea id="observación" rows="3" cols="100"
value="#{mAlias.formInterfaz.observaciones}" />
```



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:inputtextarea`

Ejemplo: Si el id es observacion

Desde javascripts: → `document. getElementById ("observacion:inputtextarea")`

Desde otro componente → `:principalForm:observacion:inputtextarea`

2.13.8. CheckBox

Para los checkbox a mostrar en el Sistema se deberá utilizar el componente

`ssicc: selectBooleanCheckbox.xhtml`

Ejemplo:

```
<ssicc:selectBooleanCheckbox value="#{mAlias.habilitaUbigeo}"
id="habilitaUbigeo" onChange="reiniciaTerritorioConsultora()"/>
```

Si se desea invocar a un método y refrescar usando AJAX:

Ejemplo:

```
<ssicc:selectBooleanCheckbox value="#{mAlias.habilitaUbigeo}"
id="habilitaUbigeo" onChange="reiniciaTerritorioConsultora()"
accionListener="#{reporteINCGanadorasConcurso2Action.changeStatus1}"
update=":principalForm:codigoRegion:select,:principalForm:codigoZona:select,:principalForm:cb1:selectBooleanCheckbox,:principalForm:cb2:selectBooleanCheckbox" />
```



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:selectBooleanCheckbox`

Ejemplo: Si el id es habilitaUbigeo

Desde javascripts: → `document. getElementById ("habilitaUbigeo:selectBooleanCheckbox_input")`

Desde otro componente → `:principalForm:habilitaUbigeo:selectBooleanCheckbox_input`

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.13.9. RadioButton

Se deberá utilizar el componente `ssicc:selectOneRadio`

Ejemplo:

```
<ssicc:selectOneRadio id="flagHabilitado"
  value="#{mAlias.formMantenimiento.flagHabilitado}" required="true">
  <f:selectItem itemLabel="#{i18n['interfazForm.habilitado']}" itemValue="#{SI}" />
  <f:selectItem itemLabel="#{i18n['interfazForm.deshabilitado']}" itemValue="#{NO}" />
</ssicc:selectOneRadio>
```

Si se desea invocar a un método y refrescar usando AJAX:

Ejemplo:

```
<ssicc:selectOneRadio id="tipoGeneracion"
  value="#{mAlias.formMantenimiento.tipoGeneracion}"
  ActionListener="#{interfazSearchAction.generarCodigoInterfazPorTipoGeneracion}"
  update=":principalForm:panelCodigoInterfaz :principalForm:panelUnitario1
:principalForm:panelPaquete :principalForm:panelFTP
:principalForm:panelParametrosInterfaz
:principalForm:panelGeneraArchivoServidor"
  required="true">
  <f:selectItem itemLabel="#{i18n['interfazForm.unitario']}" itemValue="#{SI}" />
  <f:selectItem itemLabel="#{i18n['interfazForm.paquete']}" itemValue="#{NO}" />
</ssicc:selectOneRadio>
```



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:selectOneRadio`

Ejemplo: Si el id es tipoGeneracion

Desde javascripts: → `document.getElementById("tipoGeneracion:selectOneRadio")`

Desde otro componente → `:principalForm:tipoGeneracion:selectOneRadio`

2.13.10. Grilla Datatable (Modo Consulta o de Selección simple)

Para el uso de grillas (datatable) que no se haga uso de seleccionar registro o en los casos que se pueda seleccionar solo un solo registro se deberá utilizar el componente

`ssicc:dataTable`

Ejemplo:

DataTable sin selección de registros:

```
<ssicc:dataTable id="dataTableBusqueda"
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```

mostrarTituloPanel="false"
toggleable = "true"
draggable = "false"
draggableColumns = "true"
exportable="true"
titulo="#{msg['dataTable.resultadoBusqueda']}"
value="#{mAlias.datatableBusqueda}"
fileExport="dataBusqueda"
rendered="#{not empty mAlias.listaBusqueda}"
resizableColumns="true"
mostrarMenuMantenimiento = "true"
ejecutarDBCclick = "true"
manage = "#{mAlias}"
>

```

DataTable de selección única:

```

<ssicc:dataTable id="dataTableBusqueda"
mostrarTituloPanel="false"
toggleable = "true"
draggable = "false"
draggableColumns = "true"
exportable="true"
titulo="#{msg['dataTable.resultadoBusqueda']}"
value="#{mAlias.datatableBusqueda}"
fileExport="dataBusqueda"
rendered="#{not empty mAlias.listaBusqueda}"
resizableColumns="true"
seleccionable = "true"
mostrarMenuMantenimiento = "true"
ejecutarDBCclick = "true"
manage = "#{mAlias}"
selection = "#{mAlias.beanRegistroSeleccionado}"
>

```

Para las grillas de selección múltiple en ese caso usar el componente del primefaces directamente (p:dataTable)



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando <nombreFormulario>:<id>:

Ejemplo: Si el id es tipoGeneracion

Desde javascripts: → document.getElementById("tipoGeneracion:selectOneRadio")

Desde otro componente → :principalForm:tipoGeneracion:selectOneRadio

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

2.13.11. *CommandButton*

El SSiCC al ejecutar la botonería del Sistema ejecuta submit respectivo, mientras los botones del NSSiCC NO USA submit y utiliza AJAX para invocar al método en la clase JAVA ManageBeans

La mayoría de los botones estándar ya se encuentran implementados en los templates hechos en el NSSiCC. Sin embargo pueda darse el caso de la necesidad de migrar botones que son exclusivos a la pantalla WEB para ello se deberá utilizar el componente `ssicc:linkAjax`

Ejemplo:

```
<ssicc:linkAjax value="#{i18n['button.structfile']}"
  id="botonEjecutar"
  title="#{i18n['button.structfile']}"
  icon="/resources/images/estructura.gif"
  accion="#{interfazSearchAction.estructuraArchivo}"
  update=":principalForm:idSeleccionoRegistro :principalForm:idTieneEstructura"
  oncomplete="verificarTieneEstructura();"
/>
```



Si se desea hacer referencia al objeto desde javascript o desde otro componente deberá invocarlo usando `<nombreFormulario>:<id>:linkAjax`

Ejemplo: Si el id es botonEjecutar

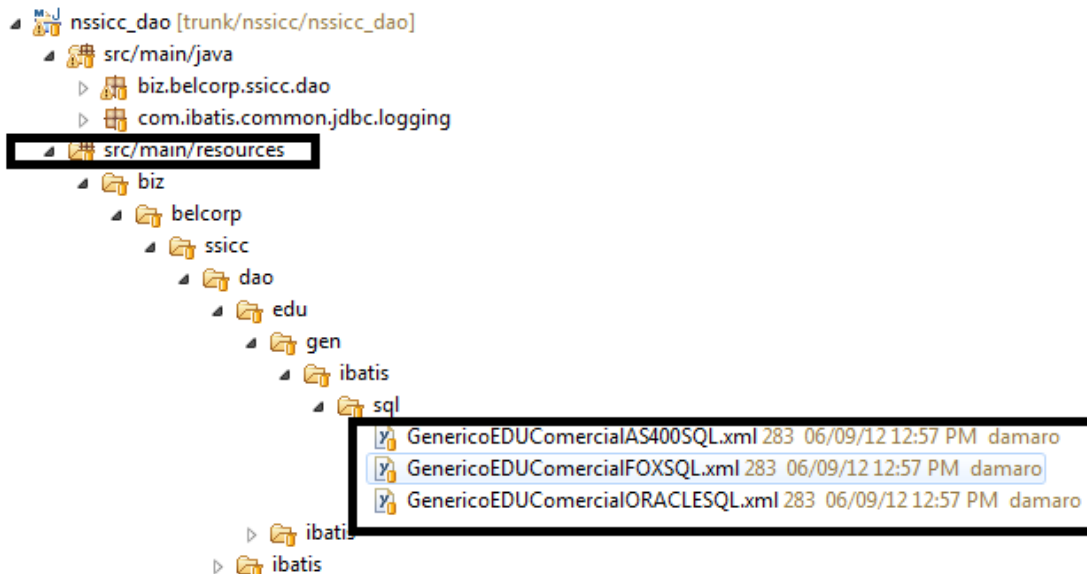
Desde javascripts: → `document. getElementById ("botonEjecutar:linkAjax")`

Desde otro componente → `:principalForm:botonEjecutar:linkAjax`

2.14. Ubicación archivos XML de ibatis

En el NSSiCC los archivos JAVA y los archivos RESOURCES (xml) se encuentran en carpetas separadas. Debido a ello al migrar los archivos xml de ibatis del SSiCC al NSSiCC se deberá colocar en la carpeta resources del subproyecto nssicc_dao

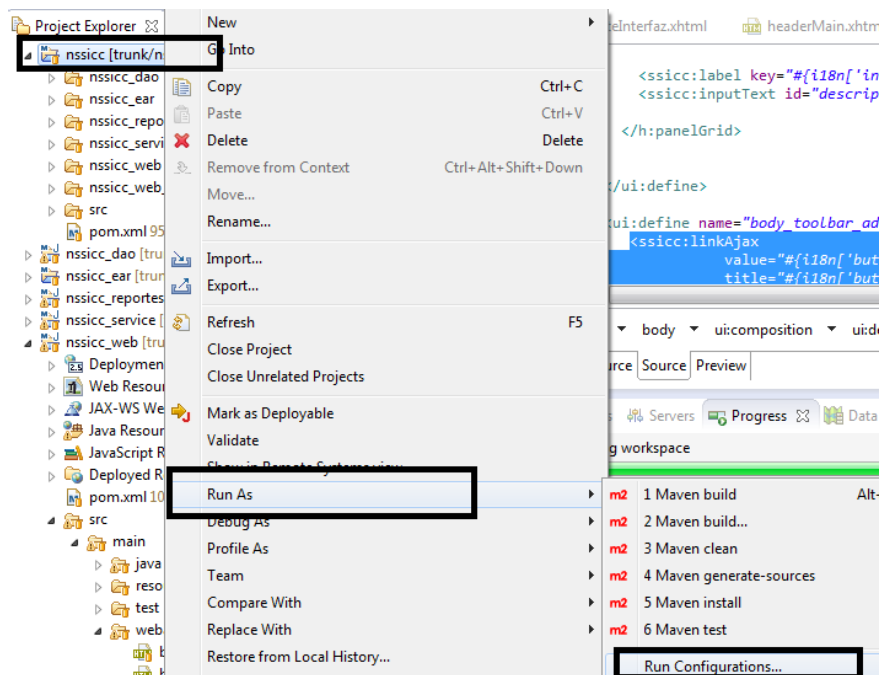
	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



2.15. Reportes

En el SSiCC solo se coloca el jrxml ya que el archivo.jasper asociado se genera al deployar la aplicación mediante el ANT

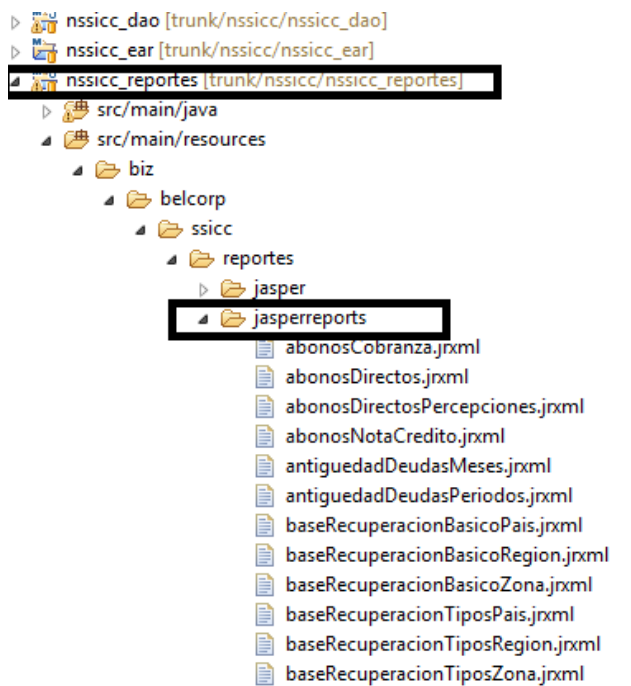
En el NSSiCC los.jasper son generados cuando se ejecuta maven a través de la sentencia mvn -install



	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Tanto el jrxml como el.jasper deben ser sincronizados al Repositorio SVN en el proyecto NSSiCC

Los jrxml deberán ser colocados en la carpeta *jasperreports* del proyecto *nssicc_reportes*



2.16. Asignando Página xhtml al Menú

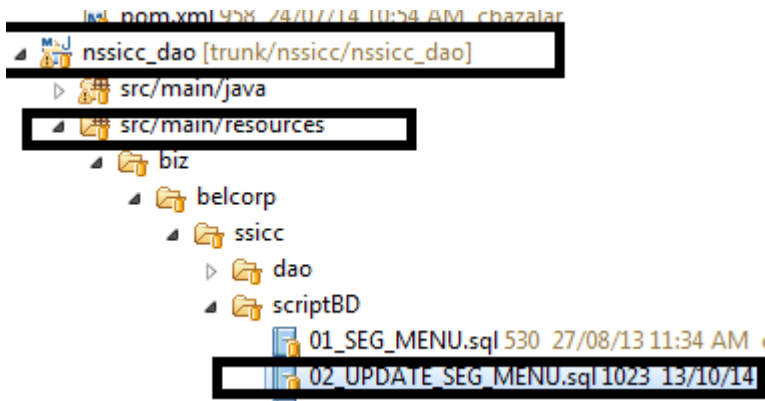
Una vez migrado una opción del menú del SSiCC al NSSiCC se deberá mapear la página xhtml en la tabla SEG_MENU

Para ello deben hacer update a dicha tabla y al campo PAG_XML por el código del Menú
Ejemplo:

```
UPDATE SEG_MENU X
SET X.PAG_XML = '/pages/scsicc/reporteSICLimitesVentaConfiguradosForm.xhtml'
WHERE X.COD_MENU = '40021815';
```

Aparte de ejecutarse dicho update en la BD, se deberá grabar dicho update en el archivo *02_UPDATE_SEG_MENU.sql* que se encuentra en el subproyecto *nssicc_dao*

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	



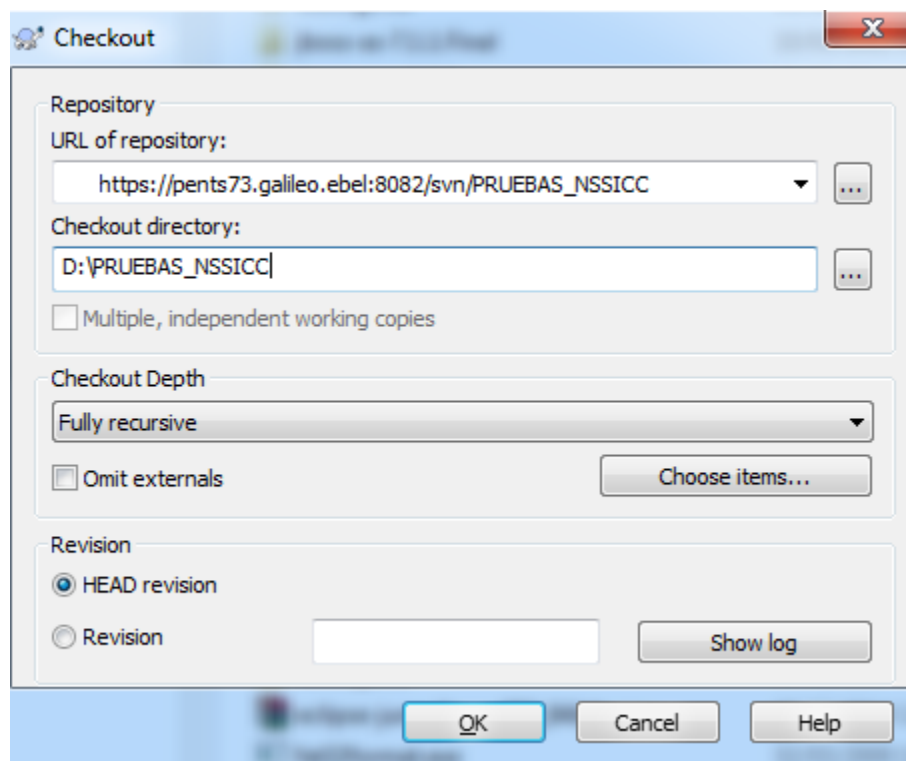
2.17. Pruebas Internas

Una vez terminado el desarrollo se deberá hacer prueba interna del mismo.

Hacer click sobre la opción SVN Checkout..., la cual le mostrara la pantalla siguiente, ingresar los datos mencionados en dicha pantalla

Repository: https://pents73.galileo.ebel:8082/svn/PRUEBAS_NSSICC

Export Directory: D:\PRUEBAS_NSSICC



	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Una vez terminado hacer click sobre el botón OK.

Para ello se deberá crear un archivo WORD con el siguiente formato:

<Codigo Menu> - <Nombre de Opcion del Menu>.doc

Ejemplo:

40020807 - Calculo Comisión por Comercializacion.doc

En dicho documento se deberá colocar:

- Imagen de Pantalla Original del SSiCC
- Imagen de Pantalla Migrada del NSSiCC
- Imágenes de Funcionamiento de la pantalla migrada en el NSSiCC

Una vez realizado dicho documento deberá ser sincronizado en el Repositorio de Pruebas

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3. Migración de Interfaces

3.1. Conceptos Previos

El Sistema maneja el concepto de Interfaz para enviar o recibir archivos TXT, CSV, XLS que se encuentran en el Servidor FTP.

De esta forma se denomina Interfaz de Salida a aquella interfaz que luego de un proceso a ejecutar en la BD genera el archivo de salida y lo coloca en una ruta determinada del Servidor.

Se denomina Interfaz de Entrada a aquella interfaz que obtiene un archivo del Servidor FTP, procesa su información la que luego será grabada en las tablas del Sistema

El Sistema SSiCC dispone de pantallas para lo mencionado, sin embargo a veces se requiere ejecutar varias interfaces al mismo tiempo dentro de la misma pantalla, debido a ello se habla del concepto de PAQUETE de interfaz, la cual será la agrupación de interfaces en otra interfaz denominada interfaz de paquete

Una interfaz simple tiene la siguiente denominación:

<CODIGO SISTEMA>-<NRO>:

Ejemplo:

DAT-1

DAT-12

COM-10

Una interfaz de tipo paquete tiene la siguiente denominación:

<CODIGO SISTEMA>-P<NRO>:

Ejemplo:

DAT-P1

DAT-P2

COM-P1

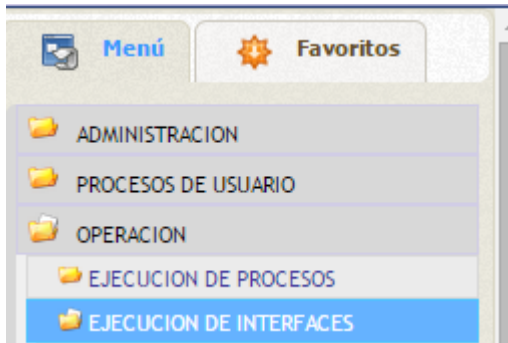
Cuando migren una pantalla de tipo Interfaz al obtener la URL tendrá la siguiente característica:

<http://pelnx3196:7777/ssiccpesdev/viewInterfazDATEnviarAdministracionFlujos.do?codigoInterfaz=DAT-P2&codigoProcesoBatch=01&codigoMenu=30021302>

Como podrán apreciar en la URL se encuentra indicado el código de Interfaz que la pantalla va a utilizar. En este ejemplo vemos que la pantalla va a invocar a una interfaz de tipo Paquete

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Las pantallas a migrar generalmente se encuentran en esta parte del Menu



3.1.1. Visualización Interfaz de Salida tipo Paquete

En el SSiCC dichas pantallas generalmente tendrán la siguiente visualización:

The image shows a web application interface titled 'DAT - Administración de Flujos' with the 'BELCORP' logo. It contains several filter fields:

- * País: PERU ESIKA
- Número de Lote: (empty)
- * Descripción Lote: DATAMART VENTAS
- * Marca: Grupo Todas
- * Acceso: Gerentes
- * Fecha de Facturación: 27/09/2014
- * Canal: Venta Directa
- * Campaña de Facturación: 201404
- Observaciones: (empty)

 Below these filters is a table titled 'Archivos' with a 'Seleccionar' column. The table lists several interfaces, all of which are checked for selection.

Archivos	Seleccionar
Interfaces	<input checked="" type="checkbox"/>
DAT-62 ENVIAR MAESTRO DE MARCAS DE PRODUCTO	<input checked="" type="checkbox"/>
DAT-66 ENVIAR ESTADÍSTICO CDR CONSULTORA	<input checked="" type="checkbox"/>
DAT-2 ENVIAR CLIENTES	<input checked="" type="checkbox"/>
DAT-117 ENVIAR PROYECCION DE RESULTADOS DEL PROGRAMA	<input checked="" type="checkbox"/>
DAT-12 ENVIAR BOLETAS DE DESPACHO	<input checked="" type="checkbox"/>
DAT-24 ENVIO TRANSACCIONES DE CLIENTES	<input checked="" type="checkbox"/>
DAT-119 ENVIAR MAESTRO DE CANAL DE INGRESO DE PEDIDO	<input checked="" type="checkbox"/>

Como la pantalla va a invocar a una interfaz de tipo PAQUETE cuando se migre finalmente en el NSSiCC deberá aparecer de la siguiente forma:

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

DAT - Administración de Flujos [Iconos de configuración, flujo, usuarios, agregar, ayuda] [Show]

Interfaz

Criterios Generación

Listado Interfaces

(*) Pais : PERU ESIKA
 (*) Marca : Grupo Todas (*) Canal : Venta Directa
 (*) Campaña de Facturación : 201404 (*) Fecha de Facturación : 26/11/2014
 Observaciones :

DAT - Administración de Flujos [Iconos de configuración, flujo, usuarios, agregar, ayuda] [Show]

Interfaz

Criterios Generación

Listado Interfaces

Seleccione Interfaces a Ejecutar:

<input checked="" type="checkbox"/>	Interfaz	Descripción Interfaz	Nivel	Hilo	Orden Ejec
<input checked="" type="checkbox"/>	DAT-02	ENVIAR MAESTRO DE MARCAS DE PRODUCTO	1	0	0
<input checked="" type="checkbox"/>	DAT-06	ENVIAR ESTADÍSTICO CDR CONSULTORA	1	0	1
<input checked="" type="checkbox"/>	DAT-2	ENVIAR CLIENTES	1	0	2
<input checked="" type="checkbox"/>	DAT-117	ENVIAR PROYECCION DE RESULTADOS DEL PROGRAMA	1	0	3
<input checked="" type="checkbox"/>	DAT-12	ENVIAR BOLETAS DE DESPACHO	1	0	4
<input checked="" type="checkbox"/>	DAT-24	ENVIO TRANSACCIONES DE CLIENTES	1	0	5
<input checked="" type="checkbox"/>	DAT-119	ENVIAR MAESTRO DE CANAL DE INGRESO DE PEDIDO	1	0	6
<input checked="" type="checkbox"/>	DAT-15	ENVIAR HOMOLOGACION ESTATUS CLIENTE	1	0	7
<input checked="" type="checkbox"/>	DAT-113	ENVIAR RANGO DE PEDIDOS	1	0	8
<input checked="" type="checkbox"/>	DAT-38	ENVIAR CONTROL DE CIERRES	1	0	9
<input checked="" type="checkbox"/>	DAT-108	ENVIAR GANANCIA LOGROS	1	0	10

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

DAT - Administración de Flujos

Resultados

Proceso Batch

Nro	Log	En Ejecución	Status	Sistema	Proceso Batch	Nro Lote	Periodo	Fecha Inicio Ejecución	Fecha Fin Ejecución	Etapas Proceso	Usuario
1		NO	✗	DAT	ENVIAR DATAMART VENTAS	201411210004	201404	21/11/2014 12:00:58	21/11/2014 12:01:08	FINALIZADO	ADMIN

Interfaces Ejecutadas

Nro	Log	Número Lote	Hilo	Orden Hilo	Orden Ejec.	Interfaz	Descripción Interfaz	Status	Fecha Inicio Ejecución	Fecha Fin Ejecución
1		201411210004	0	0	0	DAT-P2	DATAMART VENTAS	✗	21/11/2014 12:01:33	21/11/2014 12:01:41
2		201411210004	1	0	1	DAT-88	ENVIAR ESTADÍSTICO CDR CONSULTORA	✗	21/11/2014 12:01:34	21/11/2014 12:01:35
3		201411210004	1	0	3	DAT-117	ENVIAR PROYECCION DE RESULTADOS DEL PROGRAMA	✗	21/11/2014 12:01:36	21/11/2014 12:01:37
4		201411210004	1	0	4	DAT-12	ENVIAR BOLETAS DE DESPACHO	✗	21/11/2014 12:01:37	21/11/2014 12:01:39
5		201411210004	1	0	5	DAT-24	ENVIO TRANSACCIONES DE CLIENTES	✗	21/11/2014 12:01:39	21/11/2014 12:01:40

3.2. Verificando el Tipo de Interfaz

Para identificar si una interfaz es de tipo Paquete debemos ver el código de la interfaz. Si tiene la siguiente denominación es una interfaz de tipo paquete caso contrario será una interfaz unitaria.

<CODIGO SISTEMA>-**P**<NRO>:

Ejemplo:

DAT-P1

DAT-P2

COM-P1

Asimismo las interfaces pueden ser de Entrada o de Salida. Una forma rápida de identificarlo es el nombre de la Opción del Menú.

Si la opción de Menú indica Enviar (Ejemplo: Enviar Archivos Asistente Virtual) nos indica que la interfaz es de Salida

Si la opción de Menú indica Recepcionar (Ejemplo: Recepcionar Datos Telefonicos) nos indica que la interfaz es de Entrada

Otra forma segura de identificarlo es haciendo el siguiente query en la BD:

```
SELECT tip_inte FROM bas_inter x
WHERE x.cod_inte = '<codigo_interfaz>'
```

Si el valor devuelto es E entonces es Interfaz de Entrada

Si el valor devuelto es S entonces es Interfaz de Salida

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3.3. Creando Clase Form en el NSSiCC

Buscar en el SSiCC la clase FORM asociada al ACTION. Por convención deberá tener la siguiente estructura de nombre: <NOMBRE INTERFAZ>Form.java

Grabarla en el NSSiCC, en la misma carpeta indicada en el SSiCC

3.4. Creando Clase Action en el NSSiCC

```

20 * Created on 26/09/2006 11:32:06 AM
5 package biz.belcorp.ssicc.sisicc.web.action;
6
7 import java.util.ArrayList;
8
9 /**
10  * TODO Include class description here.
11  * <p>
12  * <a href="InterfazDATEnviarAdministracionFlujosAction.java.html">
13  * <i>View Source </i> </a>
14  * </p>
15  *
16  * @author <a href="mailto:jcairampoma@belcorp.biz">José A. Cairampoma</a>
17  *
18  * @struts.action input="mainMenu" validate="false" scope="request"
19  *                path="/viewInterfazDATEnviarAdministracionFlujos"
20  *                parameter="method" name="interfazDATEnviarAdministracionFlujosForm"
21  *
22  * @struts.action input="view" validate="true" scope="request"
23  *                path="/executeInterfazDATEnviarAdministracionFlujos"
24  *                parameter="method" name="interfazDATEnviarAdministracionFlujosForm"
25  *
26  * @struts.action-forward path="/WEB-INF/pages/sisicc/interfazDATEnviarAdministracionFlujosForm.jsp"
27  *                name="view"
28  */
29 public class InterfazDATEnviarAdministracionFlujosAction extends
30     BaseInterfazHiloAbstractAction {

```

Lo primero es colocar bien el package ya que como hemos visto en el punto 2.3 del documento las rutas varían en el NSSiCC

En este caso cambiar lo mostrado en verde

package biz.belcorp.ssicc.sisicc.web.action;

Por:

package biz.belcorp.ssicc.web.sisicc.action;

Asimismo eliminar lo señalado en amarillo y colocar lo siguiente

```

@ManagedBean
@SessionScoped
public class InterfazDATEnviarAdministracionFlujosAction extends
BaseInterfazAbstractAction

```

Las clases Action de Interfaz deben heredar de BaseInterfazAbstractAction tal como se muestra

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Debe quedar como se muestra a continuación:

```

+ * Created on 26/09/2006 11:32:06 AM
package biz.belcorp.ssicc.web.sisicc.action;

+ import java.util.HashMap;

- /**
 * TODO Include class description here.
 * <p>
 * <a href="InterfazDATEnviarAdministracionFlujosAction.java.html">
 * <i>View Source </i> </a>
 * </p>
 *
 * @author <a href="mailto:jcairampoma@belcorp.biz">Jose Cairampoma</a>
 *
 */
@ManagedBean
@SessionScoped
public class InterfazDATEnviarAdministracionFlujosAction extends BaseInterfazAbstractAction {

```

Cambiar la declaración de los métodos siguientes:

```

protected void setViewAttributes(HttpServletRequest request, ActionForm form)
por
protected void setViewAtributtes() throws Exception

```

```

protected Map prepareParamsBeforeExecute(ActionForm form, HttpServletRequest
request) throws Exception
por
protected Map<String, Object> prepareParamsBeforeExecute(Map params, BaseForm
form) throws Exception

```

3.5. Variables en el método setViewAtributtes

En caso encuentren la siguientes variables en el código del método setViewAtributtes del SSiCC

- codigoProcesoBatch
- codigoSistema

No pasarlo al NSSiCC, debido que la Clase Padre implementada en el NSSiCC ya obtiene los valores de dichas variables

Asimismo si encuentran alguna de las siguientes líneas no migrarlas:

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```
request.getSession().setAttribute(Constants.INTERFAZ_DATA,
(Interfaz)request.getAttribute(Constants.INTERFAZ_DATA));
request.getSession().setAttribute("archivoslist",
(List)request.getAttribute(Constants.INTERFAZ_FILES));
List archivosList =
(List)request.getAttribute(Constants.INTERFAZ_FILES);
LabelArchivos labelArchivos = (LabelArchivos)archivosList.get(0);
String pesoArchivo = labelArchivos.getPesoArchivo();
f.setPesoArchivo(pesoArchivo);
```

3.6. Revisando invocaciones a Clase Service y DAO

Se deberá verificar que los métodos en el Service existan y que no haya ningún error al respecto.

Si hubiera error se deberá migrar las Clases Service, Clases DAO y archivo XML de ibatis. Asimismo se deberá verificar en el archivo *sql-map-config.xml* que este declarado el XML de ibatis

3.7. Eliminar try catch

En los métodos del *setViewAttributes* y *prepareParamsBeforeExecute* del SSiCC existe código dentro de try catch.

Los try catch deben ser eliminados en el código migrado del NSSiCC

3.8. Verificar método Reset en la Clase Form

Verificar la existencia del método *Reset* en la Clase Form, si existe dicho método su contenido debe pasarse al método *setViewAttributes* de la Clase Action y se debe borrar el método *Reset* de la Clase Form

3.9. Creando página xhtml

Se deberá crear la pagina xhtml en base a la plantilla *templateInterfaz.xhtml*, con el nombre del jsp asociado al Action, para ello revisar lo siguiente en el SSiCC

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```

24 * Created on 26/09/2006 11:32:06 AM
5 package biz.belcorp.ssicc.sisicc.web.action;
6
7 import java.util.ArrayList;
33
34 /**
35  * TODO Include class description here.
36  * <p>
37  * <a href="InterfazDATEnviarAdministracionFlujosAction.java.html">
38  * <i>View Source </i> </a>
39  * </p>
40  *
41  * @author <a href="mailto:jcairampoma@belcorp.biz">José A. Cairampoma</a>
42  *
43  * @struts.action input="mainMenu" validate="false" scope="request"
44  *                path="/viewInterfazDATEnviarAdministracionFlujos"
45  *                parameter="method" name="interfazDATEnviarAdministracionFlujosForm"
46  *
47  * @struts.action input="view" validate="true" scope="request"
48  *                path="/executeInterfazDATEnviarAdministracionFlujos"
49  *                parameter="method" name="interfazDATEnviarAdministracionFlujosForm"
50  *
51  * @struts.action-forward path="/WEB-INF/pages/sisicc/interfazDATEnviarAdministracionFlujosForm.jsp"
52  *
53  */
54 public class InterfazDATEnviarAdministracionFlujosAction extends
55 BaseInterfazHiloAbstractAction {

```

Con el ese nombre señalado en amarillo se deberá crear la pagina en el NSSiCC

`interfazDATEnviarAdministracionFlujosForm.xhtml`

Se deberá grabar en la misma ruta que se encuentra en el SSiCC

Crearlo usando el siguiente código base:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:c="http://java.sun.com/jsp/jstl/core"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:p="http://primefaces.org/ui"
      xmlns:ssicc="http://java.sun.com/jsf/composite/componentes/ssicc"
      xmlns:bel="http://www.belcorp.biz/ssicc/components">

<body>

<ui:composition template="/templates/templateInterfaz.xhtml">
  <ui:param name="mAlias" value="#{interfazDATEnviarAdministracionFlujosAction}"/>

  <ui:define name="titulo_Pantalla">
    <h:outputText value="#{i18n['interfazDATEnviarAdministracionFlujosForm.title']}"
styleClass="clsEncabezadoPantalla" />
  </ui:define>

  <ui:define name="body_criterio_interfaz">

</ui:define>

```


	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```
</ui:composition>
```

```
</body>
```

```
</html>
```

Modificar lo señalado en amarillo

3.10. Agregando los Filtros en página xhtml

Luego de creado la pagina xhtml en base al código base del punto anterior se deberá colocar los filtros propios de la pantalla.

Para ello se debe revisar el código del SSiCC y en base a ello crear lo análogo en el NSSiCC en la sección *body_criterio_interfaz*

Para este caso sería:

SSiCC

```
<%@ include file="/common/taglibs.jsp"%>
<%@ include file="/common/variables.jsp"%>
<%@ include file="/common/calendar.jsp"%>
<%@ include file="/common/campaign.jsp"%>

<title><fmt:message key="interfazDATEnviarAdministracionFlujosForm.title"/></title>
<content tag="heading"><fmt:message key="interfazDATEnviarAdministracionFlujosForm.heading"/></content>

<body >

<div class="separator"></div>

<html:form action="executeInterfazDATEnviarAdministracionFlujos"
method="post"
styleId="interfazDATEnviarAdministracionFlujosForm"
onsubmit="return validateInterfazDATEnviarAdministracionFlujosForm(this) && customValidateInterfazDATEnviarAdministracionFlujos(this)">

<table class="detail">
<:set var="sistema" value="${interfazData.codigoSistema}" />
<:if test="${sistema == null}">
<:set var="sistema" value="${interfazDATEnviarAdministracionFlujosForm.codigoSistema}" />
</:if>
<html-el:hidden property="codigoSistema" value="${sistema}"/>
<html-el:hidden property="codigoConexionExterna" />
<html-el:hidden property="validarFechaFacturacion" />

<:set var="interfaz" value="${interfazData.codigo}" />
<:if test="${interfaz == null}">
<:set var="interfaz" value="${interfazDATEnviarAdministracionFlujosForm.codigoInterfaz}" />
</:if>
<html-el:hidden property="codigoInterfaz" value="${interfaz}"/>
</table>
</html:form>
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```

<html-el:hidden property="codigoInterfaz" value="{interfaz}"/>

<tr>
  <th>
    <ssicc:label key="interfazDATEnviarAdministracionFlujosForm.codigoPais"/>
  </th>
  <td colspan="3">
    <c:out value="{currentCountry.descripcion}" />
    <html-el:hidden property="codigoPais" value="{currentCountry.codigo}"/>
    <html:errors property="codigoPais"/>
  </td>
</tr>

<tr>
  <th>
    <fmt:message key="interfazDATEnviarAdministracionFlujosForm.numeroLote"/>:
  </th>
  <td colspan="3">
    <c:out value="{interfazDATEnviarAdministracionFlujosForm.numeroLote}"/>
  </td>
</tr>

<tr>
  <th>
    <ssicc:label key="interfazDATEnviarAdministracionFlujosForm.descripcion"/>
  </th>
  <td colspan="3">
    <c:set var="desc" value="{interfazData.descripcion}" />
    <c:if test="{desc == null}">
      <c:set var="desc" value="{interfazDATEnviarAdministracionFlujosForm.descripcion}" />
    </c:if>
  </td>
</tr>

```

No está todo el código continua (Ver código fuente en el Proyecto SSiCC)

En el NSSiCC quedaría como:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:h="http://java.sun.com/jsf/html"
  xmlns:f="http://java.sun.com/jsf/core"
  xmlns:c="http://java.sun.com/jsp/jstl/core"
  xmlns:ui="http://java.sun.com/jsf/facelets"
  xmlns:p="http://primefaces.org/ui"
  xmlns:ssicc="http://java.sun.com/jsf/composite/componentes/ssicc"
  xmlns:bel="http://www.belcorp.biz/ssicc/componentes">

  <body>

    <ui:composition template="/templates/templateInterfaz.xhtml">
      <ui:param name="mAlias" value="{interfazDATEnviarAdministracionFlujosForm.action}"/>

      <ui:define name="titulo_Pantalla">
        <h:outputText value="{i18n['interfazDATEnviarAdministracionFlujosForm.title']}" styleClass="clsEncabezadoPantalla" />
      </ui:define>

      <ui:define name="body_criterio_interfaz">
        <h:panelGrid columns="4">
          <ssicc:label key="{i18n['interfazSearchForm.descripcionPais']}" required="true" />
          <ssicc:label value="{mPantallaPrincipalBean.currentUser.pais.descripcion}" header="true" required="true" />
          <ssicc:labelBlank />
          <ssicc:labelBlank />

          <ssicc:label key="{i18n['interfazDATEnviarAdministracionFlujosForm.codigoMarca']}" required="true" />
          <ssicc:selectMenu id="codigoMarca" value="{mAlias.formInterfaz.codigoMarca}" required="true">
            <f:selectItems value="{mAlias.siccMarcaList}" var="Lista" itemLabel="{Lista.descripcion}" itemValue="{Lista.codigo}"/>
          </ssicc:selectMenu>
          <ssicc:label key="{i18n['interfazDATEnviarAdministracionFlujosForm.codigoCanal']}" required="true" />
          <ssicc:selectMenu id="codigoCanal" value="{mAlias.formInterfaz.codigoCanal}" required="true">

```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3.11. Campos Fechas

Al momento de migrar los atributos de tipo Fecha del SSiCC al NSSiCC debemos tener en cuenta lo siguiente:

En el SSiCC los atributos mencionados son de tipo String

En el NSSiCC se deberá crear adicionalmente al atributo migrado de tipo String, uno de tipo java.util.Date, la cual deberá ser usado dentro del componente ssicc:calendar

Luego se deberá pasar el valor del atributo de tipo Date al atributo de tipo String, ya que dicho atributo será utilizado en la generación del Reporte. Para realizar lo mencionado utilizar el método prepareParamsBeforeExecute

3.12. Campos Periodos que invoquen Ajax

Para estos casos si un Periodo invoca Ajax para luego refrescar otra parte de la pantalla (Ejemplo: Fechas de Inicio y Fecha de Fin), se deberá utilizar el componente ssicc:campaignCalendarListener

3.13. Revisar código javascript en la página jsp del SSiCC

Siempre se debe revisar todo el código Javascript para detectar si hay invocaciones a AJAX. Si las hubiera deberán implementar los métodos ActionListener en la Clase Action (ManageBeans) para la implementación del AJAX

Asimismo puede ser que existan validaciones previas realizadas en Javascript por ejemplo:

Validar que la Fecha Inicial sea menor igual que la Fecha Final ingresada

Validar que el Periodo Inicial sea menor igual que el Periodo Final ingresado

Para estos casos se deberá sobrecribir el método en el NSSiCC *setValidarInterfaz* la cual devolverá el mensaje de Error en caso no cumpla alguna validación respectiva

Ejemplo:

```
public String setValidarReporte() {

    if (<variable fechaFin>.compareTo(<variable fechaIni>) < 0)
        return <valor mensaje error>;

    return null;
}
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3.14. Creando Clase Service (Caso Especial solo para Interfaz de Tipo Paquete)

Existe un CASO ESPECIAL en la que se deberá crear una Clase SERVICE en el NSSiCC (**nssicc_service**) (Clase que no existe en el SSiCC)

Este caso se da, cuando existan los siguientes métodos en la clase Action del SSiCC:

- **beforeExecuteInterfaz**
- **afterExecuteInterfaz**

Ejemplo: Si revisan la clase Action *InterfazDATEnviarAdministracionFlujosAction.java* en el SSiCC verán que existe dicho método, pero en el NSSiCC no existe tal método.

El método *beforeExecuteInterfaz* deberá ser creado en una clase Service con el prefijo *paquete* y con el mismo nombre del Action pero sin el sufijo *Action* y se debe colocar el sufijo *ServiceImpl*

Ejemplo: **PaqueteInterfazDATEnviarAdministracionFlujosServiceImpl.java**

Debe heredar de **BaseInterfazPaqueteAbstractServiceImpl**

Y debe grabarse en la ruta:

`biz.belcorp.ssiCC.service.sisicc.impl;`

Los métodos migrados deben tener la siguiente estructura:

protected void beforeExecuteInterfaz(Map params) **throws** Exception
protected void afterExecuteInterfaz(Map params) **throws** Exception

Debe quedar como se muestra a continuación:

```
package biz.belcorp.ssiCC.service.sisicc.impl;

import java.util.Map;

@Service("sisicc.paqueteInterfazDATEnviarAdministracionFlujosService")
@Transactional(propagation=Propagation.REQUIRED, rollbackFor=Exception.class)
public class paqueteInterfazDATEnviarAdministracionFlujosServiceImpl extends BaseInterfazPaqueteAbstractServiceImpl {

    @Resource(name="sisicc.interfazDATService")
    private InterfazDATService interfazDATService;

    /* (non-Javadoc)
     * @see biz.belcorp.ssiCC.service.sisicc.framework.impl.BaseInterfazAbstractService#beforeExecuteInterfaz(java.util.Map)
     */
    @Override
    protected void beforeExecuteInterfaz(Map params) throws Exception {
        super.beforeExecuteInterfaz(params);
        log.debug("Entering 'before' method");

        String codigoConexionExterna = (String) params.get("codigoConexionExterna");
        if (StringUtils.equals(codigoConexionExterna, Constants.CONEXION_EXTERNA_ORACLE)) {
            interfazDATService.executeCargarTablaInterfaz(params);
        }
        return;
    }
}
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Una vez creado el Service debemos relacionarlo con el Manage, para ello se hará lo siguiente:

Abrir el archivo *applicationContext-service-sisicc-framework.xml*

Ir a la sección: `<!-- INI PAQUETE NSSiCC -->`

Declarar el paquete creado ahí tal como se muestra a continuación:

Deben colocar el código de la interfaz obtenida en la URL del Menú en el campo value

```

<!-- FIN NSSiCC -->

<!-- INI PAQUETE NSSiCC -->
<entry>
  <key>
    <value>DAT-P2</value>
  </key>
  <ref bean="sisicc.paqueteInterfazDATEnviarAdministracionFlujosService" />
</entry>

<!-- FIN PAQUETE NSSiCC -->

<entry>
  <key>
    <value>SAT-13</value>
  </key>
  <ref bean="sisicc.interfazSATRecepcionarCentrosAcopioService" />
</entry>
<entry>
  <key>
    <value>SAT-14</value>
  </key>
  <ref bean="sisicc.interfazSATRecepcionarCentrosAcopioService" />
</entry>

```



Si por algún motivo no lo declararon correctamente al momento de deployar el NSSiCC en el Servidor JBoss va a generar error y no va a levantar la aplicación

Se deberá verificar que este invocando a dicho service. Para ello colocar un punto de interrupción en alguna parte del Service y en modo debug verificar que este ingresando a dicho service

3.15. Modificando Clase Service (Caso Especial solo para Interfaz Unitaria)

Existe un CASO ESPECIAL en la que se deberá modificar la Clase SERVICE en el NSSiCC (**nssicc_service**)

Este caso se da, cuando existan los siguientes métodos en la clase Action del SSiCC:

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

- `beforeExecuteInterfaz`
- `afterExecuteInterfaz`

Ejemplo: Si revisan la clase Action `InterfazDATEnviarAdministracionFlujosAction.java` en el SSiCC verán que existe dicho método, pero en el NSSiCC no existe tal método.

Abrir el archivo `applicationContext-service-sisicc-framework.xml`

Con el código de Interfaz obtenido en la URL de la pantalla deberán buscarlo en dicho archivo

Ejemplo:

Si el código de interfaz en SAM-7

Al buscarlo en el archivo encontraran lo siguiente:

```
<entry>
  <key>
    <value>SAM-7</value>
  </key>
  <ref bean="sisicc.interfazSAMEnviarMovimientosAlmacenService" />
</entry>
```

En este caso buscamos la Clase Java `InterfazSAMEnviarMovimientosAlmacenServiceImpl.java`

En dicha clase se deberán migrar los métodos respectivos

Los métodos migrados deben tener la siguiente estructura:

```
protected void beforeExecuteInterfaz(Map params) throws Exception
protected void afterExecuteInterfaz(Map params) throws Exception
```

3.16. Verificando Lista de Interfaces (Solo para Interfaz de Tipo Paquete)

Deberán realizar el siguiente select:

```
SELECT * FROM bas_compo_paquete x
WHERE x.inte_cod_inpa = <codigo interfaz>
```

Ejemplo:

```
SELECT * FROM bas_compo_paquete x
WHERE x.inte_cod_inpa = 'DAT-P2'
```

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

```
SELECT * FROM bas_compo_paque x
WHERE x.inte_cod_inpa = 'DAT-P2'
```

	PAIS_COD_PAIS	SIST_COD_SIST	INTE_COD_INPA	INTE_COD_INTE	ORD_EJEC	ORD_EJEC_MULT_HILO	NIV_HILO	IND_C1
1	PE	DAT	DAT-P2	DAT-103	18	0	1	0
2	PE	DAT	DAT-P2	DAT-67	19	0	1	0
3	PE	DAT	DAT-P2	DAT-71	37	0	1	0
4	PE	DAT	DAT-P2	DAT-74	38	0	1	0
5	PE	DAT	DAT-P2	DAT-107	41	0	1	0
6	PE	DAT	DAT-P2	DAT-112	42	0	1	0
7	PE	DAT	DAT-P2	DAT-8	60	0	1	0
8	PE	DAT	DAT-P2	DAT-11	61	0	1	0
9	PE	DAT	DAT-P2	DAT-63	64	0	1	0
10	PE	DAT	DAT-P2	DAT-68	65	0	1	0
11	PE	DAT	DAT-P2	DAT-72	14	0	1	0
12	PE	DAT	DAT-P2	DAT-7	15	0	1	0
13	PE	DAT	DAT-P2	DAT-12	4	0	1	0
14	PE	DAT	DAT-P2	DAT-36	9	0	1	0
15	PE	DAT	DAT-P2	DAT-35	16	0	1	0
16	PE	DAT	DAT-P2	DAT-26	24	0	1	0
17	PE	DAT	DAT-P2	DAT-127	27	0	1	0

110 rows selected in 0.141 seconds

En este caso apreciamos que contiene 110 interfaces unitarias contenidas en la interfaz de paquete DAT-P2

Esa misma cantidad de interfaces debe mostrarse en el TAB Listado Interfaces de la pantalla migrada
Ejemplo:

DAT - Administración de Flujos																																																																																										
<div>Interfaz</div> <div> <div>Criterios Generación</div> <div>Listado Interfaces</div> </div>																																																																																										
<div> <div>Seleccione Interfaces a Ejecutar:</div> <table> <tr> <th><input type="checkbox"/></th><th>Nro</th><th>Interfaz</th><th>Descripción Interfaz</th><th>Nivel</th><th>Hilo</th><th>Orden Eje</th></tr> <tr><td><input type="checkbox"/></td><td>100</td><td>DAT-160</td><td>ENVIAR MAESTRO DE LIDERES (DAT-160)</td><td>1</td><td>1</td><td>115</td></tr> <tr><td><input type="checkbox"/></td><td>101</td><td>DAT-161</td><td>ENVIAR MAESTRO CLASIFICACION DE LIDERES (D</td><td>1</td><td>1</td><td>116</td></tr> <tr><td><input type="checkbox"/></td><td>102</td><td>DAT-162</td><td>ENVIAR MAESTRO SECCION (DAT-162)</td><td>1</td><td>1</td><td>117</td></tr> <tr><td><input type="checkbox"/></td><td>103</td><td>DAT-163</td><td>ENVIAR MAESTRO BAJAS (DAT-163)</td><td>1</td><td>1</td><td>118</td></tr> <tr><td><input type="checkbox"/></td><td>104</td><td>DAT-164</td><td>ENVIAR MAESTRO CALENDARIO FERIADOS (DAT-1</td><td>1</td><td>1</td><td>119</td></tr> <tr><td><input type="checkbox"/></td><td>105</td><td>DAT-165</td><td>ENVIAR MAESTRO EXIGENCIA WEB (DAT-165)</td><td>1</td><td>1</td><td>120</td></tr> <tr><td><input type="checkbox"/></td><td>106</td><td>DAT-167</td><td>ENVIAR RANKING NIVEL</td><td>1</td><td>1</td><td>121</td></tr> <tr><td><input type="checkbox"/></td><td>107</td><td>DAT-168</td><td>ENVIAR ARCHIVO DE CONTROL</td><td>1</td><td>1</td><td>122</td></tr> <tr><td><input type="checkbox"/></td><td>108</td><td>DAT-169</td><td>ENVIAR CONSULTORAS EXCLUIDAS</td><td>1</td><td>1</td><td>123</td></tr> <tr><td><input type="checkbox"/></td><td>109</td><td>DAT-168</td><td>ENVIAR DESCUENTOS NUEVAS</td><td>1</td><td>1</td><td>124</td></tr> <tr><td><input type="checkbox"/></td><td>110</td><td>DAT-170</td><td>ENVIAR FECHAS DE ENTREGA EXACTA</td><td>1</td><td>1</td><td>125</td></tr> </table> </div>							<input type="checkbox"/>	Nro	Interfaz	Descripción Interfaz	Nivel	Hilo	Orden Eje	<input type="checkbox"/>	100	DAT-160	ENVIAR MAESTRO DE LIDERES (DAT-160)	1	1	115	<input type="checkbox"/>	101	DAT-161	ENVIAR MAESTRO CLASIFICACION DE LIDERES (D	1	1	116	<input type="checkbox"/>	102	DAT-162	ENVIAR MAESTRO SECCION (DAT-162)	1	1	117	<input type="checkbox"/>	103	DAT-163	ENVIAR MAESTRO BAJAS (DAT-163)	1	1	118	<input type="checkbox"/>	104	DAT-164	ENVIAR MAESTRO CALENDARIO FERIADOS (DAT-1	1	1	119	<input type="checkbox"/>	105	DAT-165	ENVIAR MAESTRO EXIGENCIA WEB (DAT-165)	1	1	120	<input type="checkbox"/>	106	DAT-167	ENVIAR RANKING NIVEL	1	1	121	<input type="checkbox"/>	107	DAT-168	ENVIAR ARCHIVO DE CONTROL	1	1	122	<input type="checkbox"/>	108	DAT-169	ENVIAR CONSULTORAS EXCLUIDAS	1	1	123	<input type="checkbox"/>	109	DAT-168	ENVIAR DESCUENTOS NUEVAS	1	1	124	<input type="checkbox"/>	110	DAT-170	ENVIAR FECHAS DE ENTREGA EXACTA	1	1	125
<input type="checkbox"/>	Nro	Interfaz	Descripción Interfaz	Nivel	Hilo	Orden Eje																																																																																				
<input type="checkbox"/>	100	DAT-160	ENVIAR MAESTRO DE LIDERES (DAT-160)	1	1	115																																																																																				
<input type="checkbox"/>	101	DAT-161	ENVIAR MAESTRO CLASIFICACION DE LIDERES (D	1	1	116																																																																																				
<input type="checkbox"/>	102	DAT-162	ENVIAR MAESTRO SECCION (DAT-162)	1	1	117																																																																																				
<input type="checkbox"/>	103	DAT-163	ENVIAR MAESTRO BAJAS (DAT-163)	1	1	118																																																																																				
<input type="checkbox"/>	104	DAT-164	ENVIAR MAESTRO CALENDARIO FERIADOS (DAT-1	1	1	119																																																																																				
<input type="checkbox"/>	105	DAT-165	ENVIAR MAESTRO EXIGENCIA WEB (DAT-165)	1	1	120																																																																																				
<input type="checkbox"/>	106	DAT-167	ENVIAR RANKING NIVEL	1	1	121																																																																																				
<input type="checkbox"/>	107	DAT-168	ENVIAR ARCHIVO DE CONTROL	1	1	122																																																																																				
<input type="checkbox"/>	108	DAT-169	ENVIAR CONSULTORAS EXCLUIDAS	1	1	123																																																																																				
<input type="checkbox"/>	109	DAT-168	ENVIAR DESCUENTOS NUEVAS	1	1	124																																																																																				
<input type="checkbox"/>	110	DAT-170	ENVIAR FECHAS DE ENTREGA EXACTA	1	1	125																																																																																				

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

3.17. Realizando pruebas internas (Solo para Interfaz de Salida)

Para ello seleccionar todas las interfaces de la Lista de Interfaces y hacer click en el Icono



Una vez que termine la ejecución en la sección Resultado deben aparecer con Status Ir a la Grilla de Interfaces Ejecutadas

Interfaces Ejecutadas											
	Nro	Log	Número Lote	Hilo	Orden Hilo	Orden Ejec.	Interfaz	Descripción Interfaz	Status	Fecha Inicio Ejecución	Fecha Fin Ejecución
	1		201411270001	0	0	0	DAT-P2	DATAMART VENTAS		27/11/2014 09:54:00	27/11/2014 10:09:38
	2		201411270001	1	0	0	DAT-62	ENVIAR MAESTRO DE MARCAS DE PRODUCTO		27/11/2014 10:09:23	27/11/2014 10:09:27
	3		201411270001	1	0	3	DAT-117	ENVIAR PROYECCION DE RESULTADOS DEL PROGRAMA		27/11/2014 10:09:28	27/11/2014 10:09:33
	4		201411270001	1	0	4	DAT-12	ENVIAR BOLETAS DE DESPACHO		27/11/2014 10:09:34	27/11/2014 10:09:38

Presionar en el campo Log en cada uno de los Registros.

Si aparece los siguientes mensajes:

- Error en la ejecución del paquete
- biz.belcorp.ssicc.service.sisicc.framework.exception.InterfazException: Error al recuperar el archivo ZIP: DAT-62_201411270001.ZIP

DAT-62 - ENVIAR MAESTRO DE MARCAS DE PRODUCTO

Log del Proceso

```

biz.belcorp.ssicc.service.sisicc.framework.exception.InterfazException: Error al recuperar el archivo ZIP: DAT-62_201411270001.ZIP

```

Status

Procesos Previos a la Ejecución Interfaz

Fecha Inicio : 27/11/2014 10:09:24 Fecha Fin : 27/11/2014 10:09:25

Ejecución Interfaz

Fecha Inicio : 27/11/2014 10:09:26 Fecha Fin : 27/11/2014 10:09:27

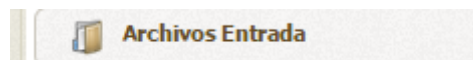
Procesos Posteriores a la Ejecución Interfaz

Fecha Inicio : Fecha Fin :

Si saliese otro Error diferente a los mencionados significa que hay un error en la programación en el código migrado

3.18. Realizando pruebas internas (Solo para Interfaz de Entrada)

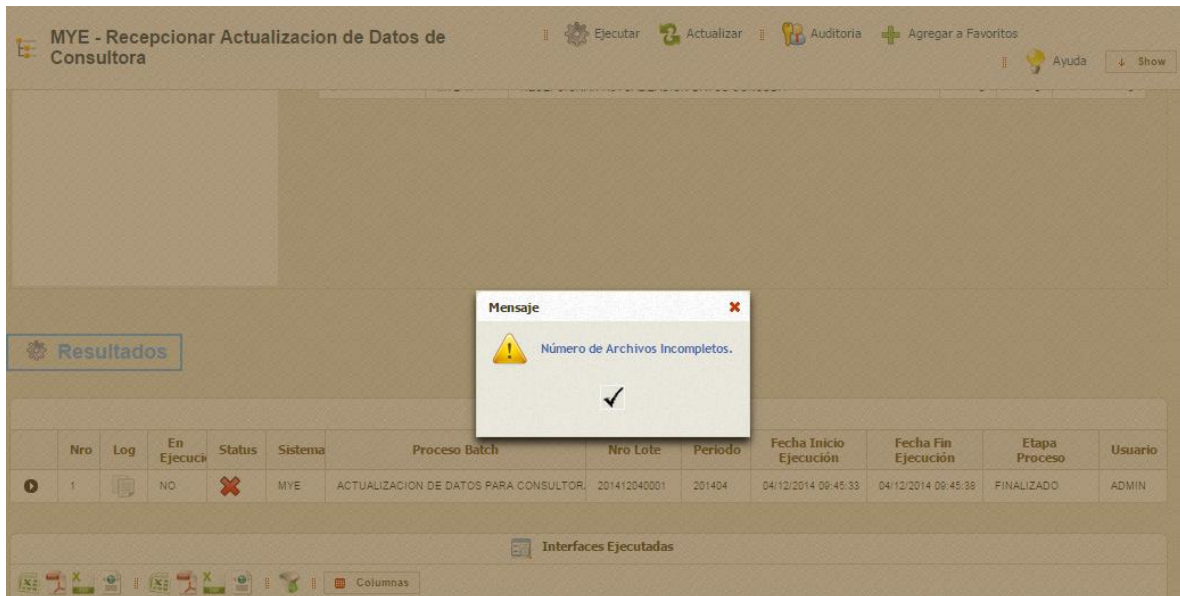
En este caso verificar que aparezca la pestaña



Asimismo debe haber la misma cantidad de interfaces entre lo mostrado en la pestaña *Listado Interfaces* y la pestaña *Archivos Entrada*

	Migración Código SSiCC a NSSiCC	
	Proyecto SSiCC	

Al momento de ejecutar debe salirle el siguiente mensaje



Si sale dicho mensaje esta OK