

El entorno de Programación

Contacto

Lic. José N Castro <jncastro@gylgroup.com>

Lic. Aldo Fernández Villalba <avillalba@gylgroup.com>

2. Programación Cobol-DB2

Programación Cobol-DB2

Los accesos a las bases de datos **IBM DB2** se gestionan desde el programa utilizando instrucciones SQL.

El SQL es un lenguaje formal estándar, basado en el álgebra relacional, para administrar datos en tablas de formato relacional.

- Propuesto por E. Codd del Laboratorio de Investigación de IBM de California en 1970
- Elegido como estándar por ANSI en 1986
- Establecido como estándar por ISO, la Organización Internacional de Normalización en 1987
- Introducido por primera vez en un producto comercial por Oracle en 1979
- Hay muchas implementaciones del lenguaje SQL y los códigos no son 100% portables

Características del lenguaje DB2

El lenguaje SQL Está compuesto por varios sub conjuntos de instrucciones.

- Instrucciones de **definición de estructuras** o tablas de datos
- Instrucciones de **manipulación de datos**
- Instrucciones para control de **autorización de acceso** a los datos
- Hay otras para diferentes funciones

Definición de estructuras de datos

Generar una base de datos nueva

```
CREATE DATABASE prueba
```

Eliminar una tabla de la base de datos

```
DROP TABLE usuarios
```

Generar una tabla nueva en la base de datos

```
CREATE TABLE usuarios (  
    legajo        int(11),  
    nombre        varchar(100),  
    email         varchar(30),  
    domicilio     varchar(50),  
    PRIMARY KEY (legajo)  
);
```

Manipulación de datos

Insertar una fila en una tabla

```
INSERT INTO usuarios  
VALUES(45, "Carlos", "cs@gmail.com", "Chacabuco 1150")
```

Recuperar filas de una tabla

```
SELECT * FROM usuarios
```

```
SELECT legajo, nombre FROM usuarios
```

```
SELECT * FROM usuarios WHERE legajo = 45
```

```
SELECT usr.legajo, usr.nombre, usr.email ped.producto  
FROM usuarios AS usr, pedidos AS ped  
WHERE usr.legajo = ped.legajo
```

Documentación de lenguaje SQL para DB2

DB2 for z/OS SQL Reference

https://www.ibm.com/docs/en/SSEPEK/pdf/db2z_10_sqlrefbook.pdf

Ejemplo de un programa Cobol-DB2

...

WORKING-STORAGE SECTION.

EXEC SQL

INCLUDE SQLCA

END-EXEC.

EXEC SQL DECLARE PERSONAS TABLE

(DNI DECIMAL(8) NOT NULL,

 NOMBRE CHAR(50) NOT NULL,

 APELLIDO CHAR(50) NOT NULL,

 TELEFONO DECIMAL(9) NOT NULL)

END-EXEC.

01 WX-DATOS.

 05 WX-ARGUMENTO PIC 9(8) VALUE 12345678.

01 WX-PERSONAS.

 05 WX-DNI PIC X(08) .

 05 WX-NOMBRE PIC X(40) .

 05 WX-APELLIDO PIC X(30) .

 05 WX-TELEFONO PIC X(12) .

...

...

01 SQLCA.

```
05 SQLCAID PIC X(8) VALUE "SQLCA ".
05 SQLCABC PIC S9(9) COMP-5 VALUE 136.
05 SQLCODE PIC S9(9) COMP-5.
05 SQLERRM.
    49 SQLERRML PIC S9(4) COMP-5.
    49 SQLERRMC PIC X(70) .
05 SQLERRP PIC X(8) .
05 SQLERRD OCCURS 6 TIMES PIC S9(9) COMP-5.
05 SQLWARN.
    10 SQLWARN0 PIC X.
    10 SQLWARN1 PIC X.
    10 SQLWARN2 PIC X.
    10 SQLWARN3 PIC X.
    10 SQLWARN4 PIC X.
    10 SQLWARN5 PIC X.
    10 SQLWARN6 PIC X.
    10 SQLWARN7 PIC X.
05 SQLEXT.
    10 SQLWARN8 PIC X.
    10 SQLWARN9 PIC X.
    10 SQLWARNA PIC X.
    10 SQLSTATE PIC X(5) .
```

...

...

PROCEDURE DIVISION.

EXEC SQL

SELECT NOMBRE, APELLIDO, TELEFONO

INTO :WX-NOMBRE, :WX-APELLIDO, :WX-TELEFONO

FROM PERSONAS-01

WHERE DNI = :WX-ARGUMENTO

END-EXEC.

IF SQLCODE EQUAL ZEROES

DISPLAY 'Lectura correcta'.

...

Cobol-DB2

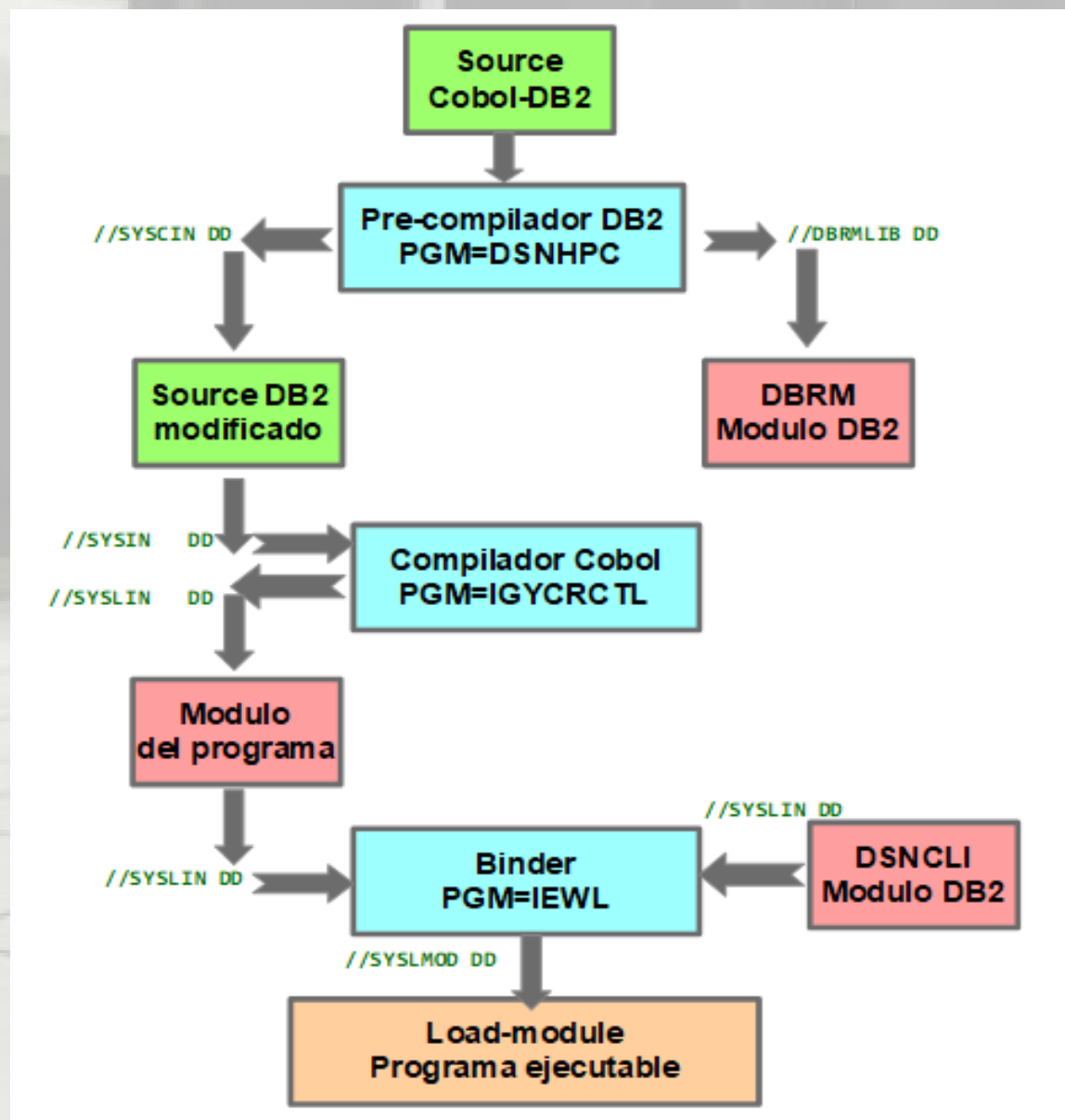
Cuando un programa utiliza SQL para acceder a bases de datos DB2 debe procesarse por un pre-compilador.

EL pre-compilador Cobol-DB2:

- Modifica el código fuente y genera un programa fuente nuevo
- Agrega áreas de datos en «*working-storage*», y procesos estándar de SQL al comienzo de la «*procedure division*»
- Reemplaza las sentencias «*Exec SQL*» por llamadas a módulos de la interfaz del DB2 válidos para Cobol
- Utiliza los datos de las sentencias SQL y la información de variables para generar un módulo DBRM «*DataBase Request Module*»

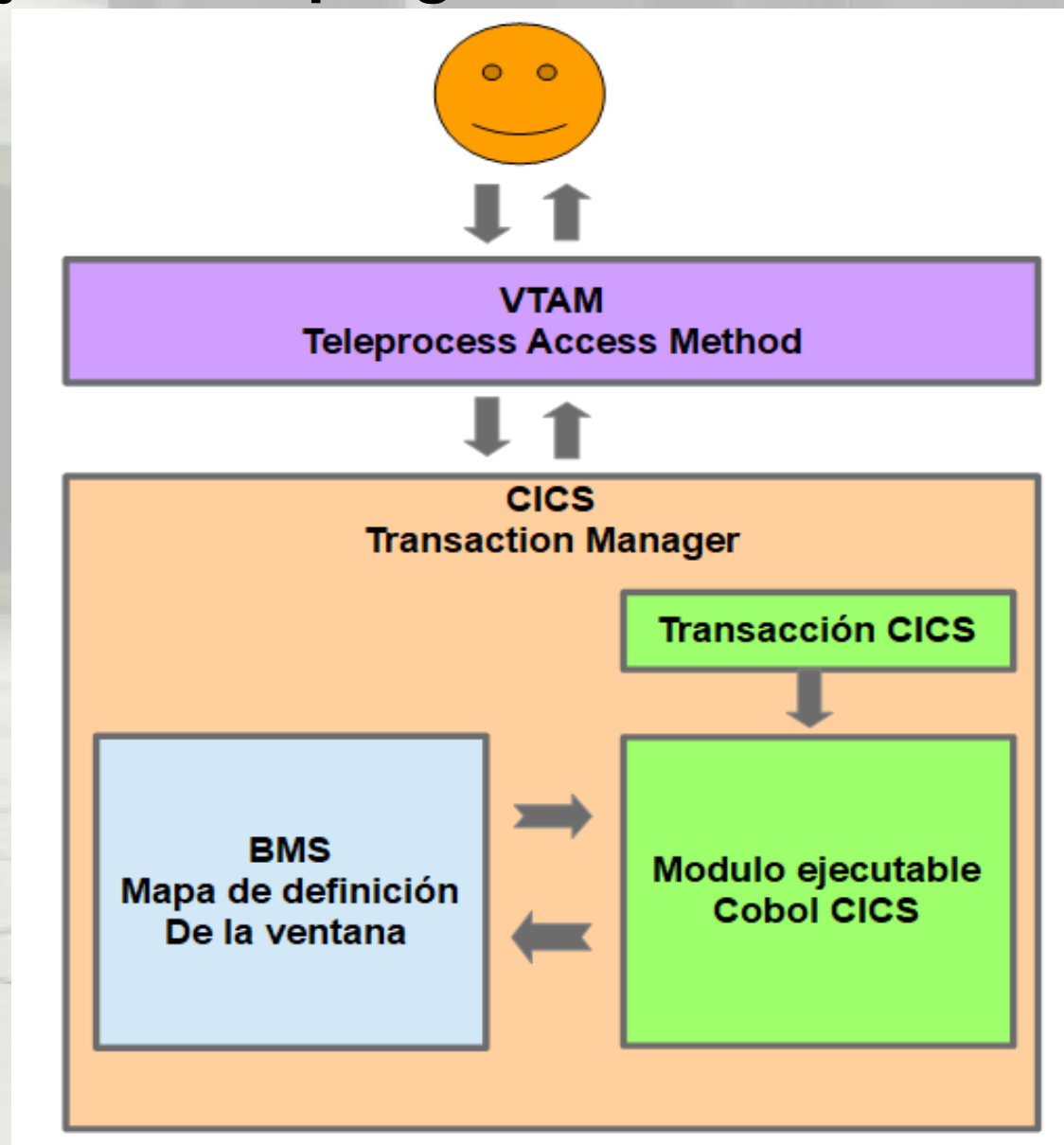
```
...
*****EXEC SQL
*****  SELECT NOMBRE, APELLIDO, TELEFONO
*****  INTO   :WX-NOMBRE, :WX-APELLIDO, :WX-TELEFONO
*****  FROM   PERSONAS-01
*****  WHERE  DNI = :WX-ARGUMENTO
*****END-EXEC.
      PERFORM SQL-INITIAL UNTIL SQL-INIT-DONE
      CALL 'DSNHLL' USING SQL-PTLIST5
      IF SQLCODE < 0 GO TO G999-ERROR-DB2  ELSE
      IF SQLCODE > 0 AND SQLCODE NOT = 100
      OR SQLWARN0 = 'W'  GO TO G999-ERROR-DB2  ELSE
      MOVE 1 TO SQL-INIT-FLAG
      END-IF
      END-IF.
...
```

Compilación Cobol-DB2



3. Programación Cobol-CICS

Como se ejecuta un programa Cobol-CICS



Definición de una ventana CICS usando macros BMS

```
*-----*  
*                               *  
*   Hola CICS!                 *  
*                               *  
*-----*
```

Tu consulta: **hola programa**

Definición del mapa CICS usando macros BMS

```

*-----
*   Mapa para el programa Hola CICS
*-----
HOLAMS  DFHMSD  TYPE=DSECT,MODE=INOUT,TERM=ALL,STORAGE=AUTO,LANG=COBOL
HOLAMP  DFHMDI  SIZE-(24,80),LINE=1,COLUMN=1,COLOR=GREEN,HILIGHT=OFF,
          MAPATTS=(COLOR,HILIGHT),DSATTS=HILIGHT,CTRL=FREEKB
          DFHMDF  POS=(10,10),LENGTH=20,INITIAL='*-----*',
          COLOR=BLUE.ATTRB=(ASKIP,NORM)
          DFHMDF  POS=(11,10),LENGTH=20,INITIAL='*                      *',
          COLOR=BLUE,ATTRB=(ASKIP,NORM)
          DFHMDF  POS*(12,10),LENGTH=20,INITIAL»'*      Hola CICS!      *',
          COLOR = BLUE.ATTRB=(ASKIP,NORM)
          DFHMDF  POS"(13,10),LENGTH=20,INITIAL='*                      *',
          COLOR=BLUE,ATTRB=(ASKIP,NORM)
          DFHMDF  POS=(14,10),LENGTH=20,INITIAL='*-----*',
          COLOR=BLUE,ATTRB=(ASKIP,NORM)
          DFHMDF  POS=(16,10),LENGTH=8,
          ATTRB=(ASKIP,NORM),INITIAL='Tu consulta:'
MENS01  DFHMDF  POS=(16,25),LENGTH=20,ATTRB=(UNPROT,BRT)
          DFHMSD  TYPE=FINAL
          END

```

Definición del mapa CICS usando macros BMS

Copy HOLAMP.

```
01 HOLAMPI.  
  02 MENS01L      PIC S9(4) COMP.  
  02 MENS01A      PIC X.  
  02 FILLER REDEFINES MENS01A.  
    03 MENS01F    PIC X.  
  02 MENS01I      PIC X(25).
```

Programa Cobol CICS conversacional

IDENTIFICATION DIVISION.

* Curso Mainframe G&L Group - UNLaM

PROGRAM-ID. HOLAPRG.

ENVIRONMENT DIVISION.

DATA DIVISION.

WORKING-STORAGE SECTION.

* Copiar o pegar aqui la definicion del panel BMS

PROCEDURE DIVISION.

MOstrar-PANEL

Exec CICS send map('HOLAMP')

End-exec.

Exec CICS receive map('HOLAMP')

End-exec.

Programa Cobol CICS pre compilado

```
...  
  
*   Exec CICS send map('HOLAMP')  
*   End-exec.  
CALL 'DFHEI1' USING BY CONTENT X'020A00000072400000000000  
-   '000000000000000000F0F1F8F1F2404040'  END-CALL  
  
...
```

Programa Cobol CICS «pseudo-conversacional»

```
...  
  IDENTIFICATION DIVISION.  
*-----*  
*  Curso Mainframe G&L Group - UNLaM  
*-----*  
  PROGRAM-ID.  HOLAPRG.  
  
  ENVIRONMENT DIVISION.  
  DATA DIVISION.  
    WORKING-STORAGE SECTION.  
    ...  
    01  WS-COMMAREA.  
        03  COM-FASE                               PIC 9 Value 0.  
            88  COM-FASE-CERO                        Value 0.  
            88  COM-FASE-UNO                         Value 1.  
    ...
```

Programa Cobol CICS «pseudo-conversacional»

```

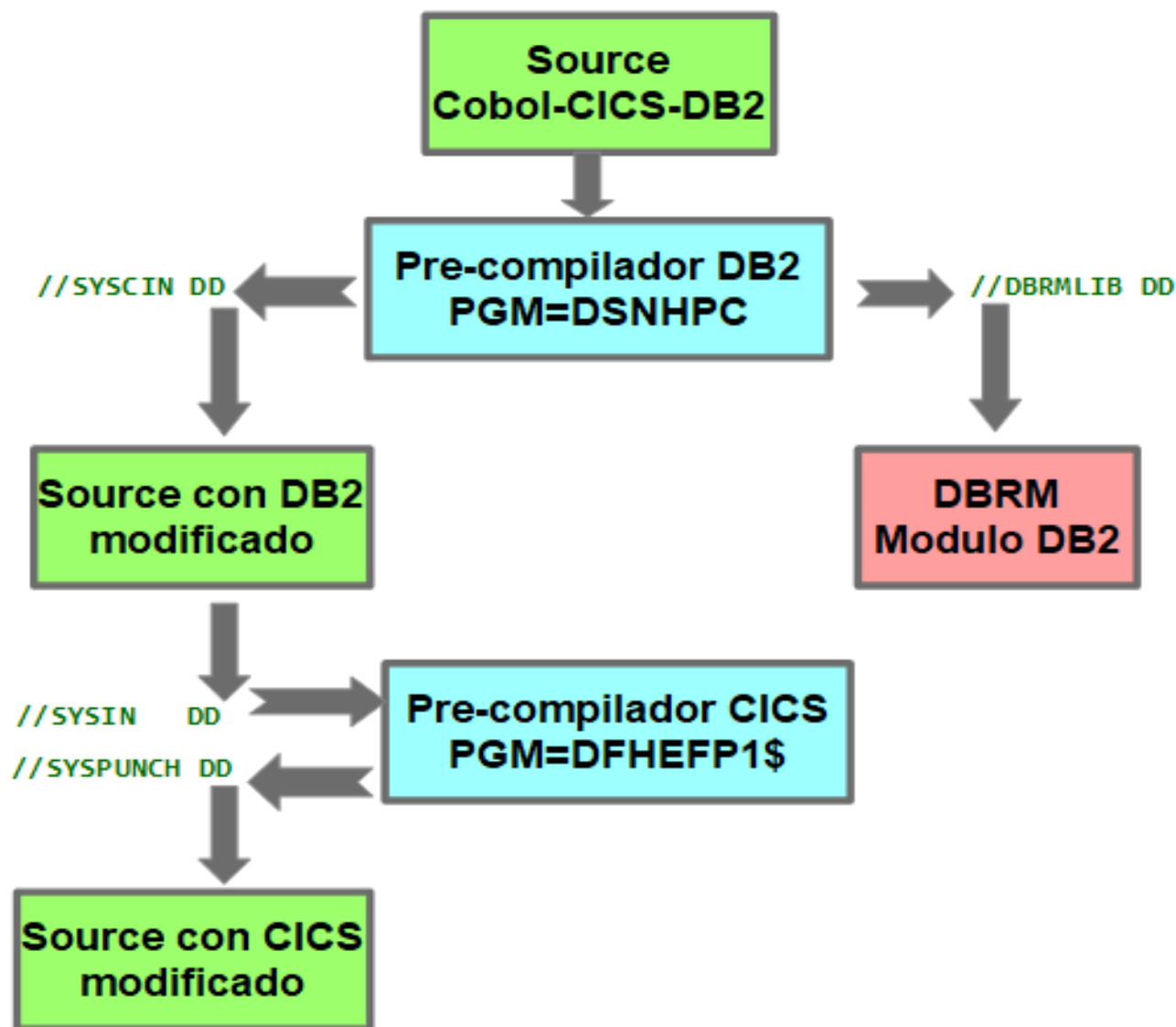
PROCEDURE DIVISION.
MOSTRAR-PANEL
*      -- Activa la señal de mapa enviado y vuelve al CICS -- *
      Exec CICS send map('HOLAMP')
      End-exec.

      Move 1 to COM-FASE.
      Exec CICS return transid ('mitransaccion')
            Commarea (WS-COMMAREA)
            Length (1)
      End-exec.

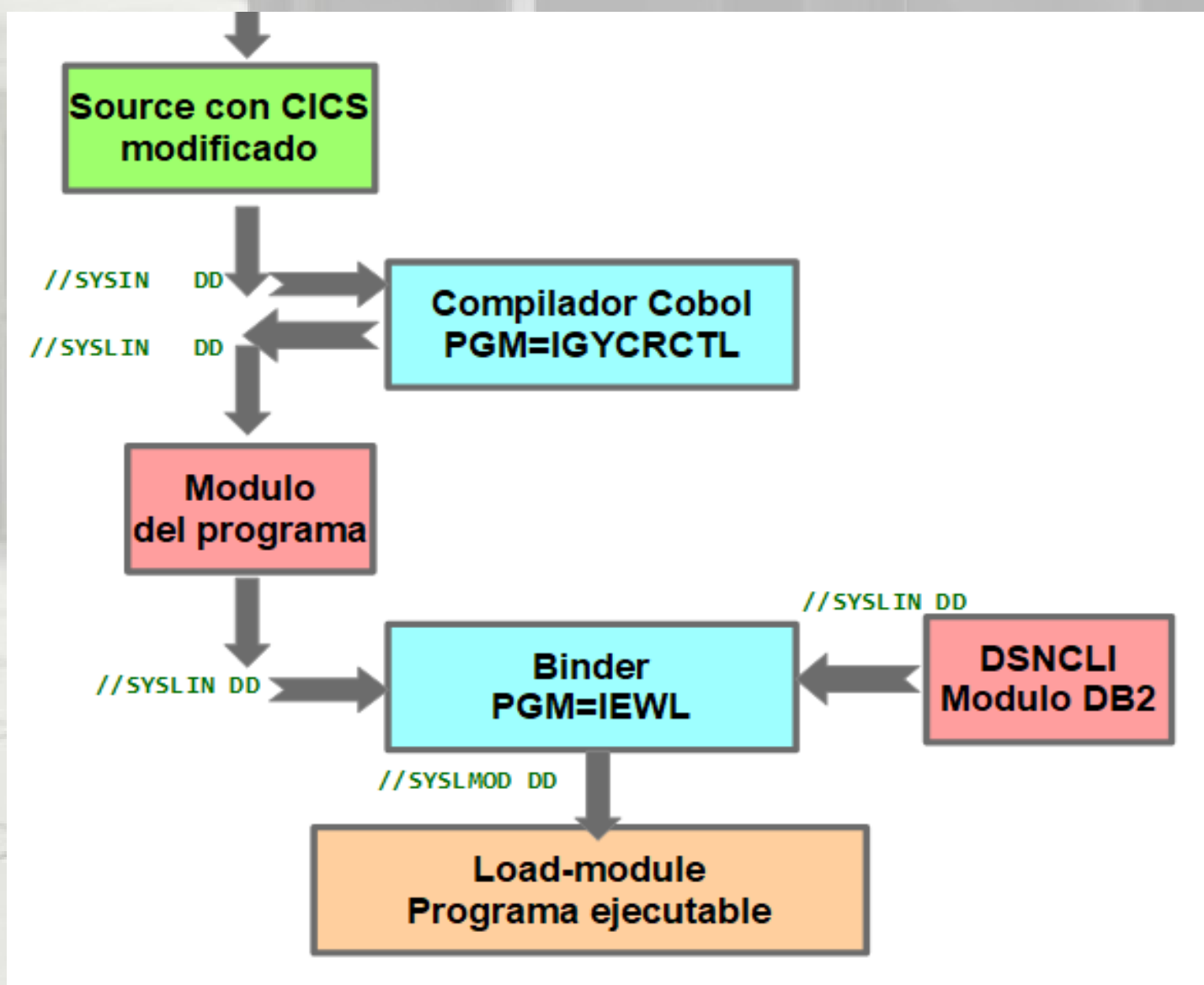
*      -- Recibir el mapa de CICS y procesar los datos -- *
      IF COM-FASE-UNO
          Exec CICS receive map ('HOLAMP')
          End-exec
          ...
      End-if.
  
```

4. Programa Cobol-CICS-DB2

Compilación Cobol-CICS-DB2



Compilación Cobol-CICS-DB2



¿Dudas, Preguntas?

Contactos

José N Castro <jncastro@gylgroup.com>

Aldo Fernández Villalba <avillalba@gylgroup.com>