

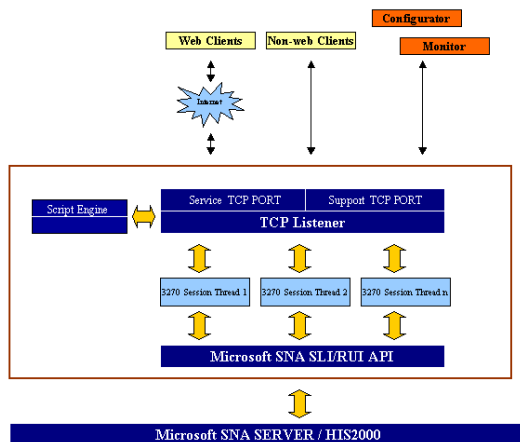
Descripción del Producto

El Servidor de Comunicaciones SNA/3270 es una aplicación que permita la conectividad y la administración de sesiones entre diversos ambientes clientes y un equipo MainFrame IBM. En este entorno, diversas aplicaciones Web y No-Web podrían solicitar los servicios de información almacenados en un equipo Main-Frame, y que su consulta sólo está disponible por medio de una Terminal 3270 (SNA LU Tipo 2).

Con este esquema propuesto, las aplicaciones clientes podrán consultar un servicio aplicativo en particular del Host IBM, mediante el envío de una solicitud utilizando un formato transaccional del tipo TCP al Servidor de Comunicaciones 3270, el cual encargaría de transportarlo a la infraestructura SNA. La solicitud de servicio corresponde a un "script" que es ejecutado por el servidor, y en el cual se describe la característica de la extracción de información deseada del Mainframe. Esta extracción puede ser del tipo transaccional, de consolidación o de tipo Batch.

Arquitectura Tecnológica

A continuación presentamos un diagrama que presenta la arquitectura tecnológica del Servidor de Comunicaciones 3270.



Servidor de Comunicaciones

El Servicio de Comunicaciones SNA/3270 CommServer3270 corresponde a un servicio Win32 multi-threading que se ejecuta en "background" y se encargara de atender los requerimientos de comunicaciones 3270.

Entre las características que presenta dicho servidor, relacionamos las siguientes:

- o Posibilidad de ejecución de múltiples instancias del CommServer3270 dentro de un mismo servidor ó en granja de servidores.
- o Soporte de múltiples sesiones simultáneas solicitadas por los clientes, con capacidades Multi-Threading.
- o Acceso a múltiples Servidores SNA bajo un esquema de POOL de terminales 3270, de forma tal que la escalabilidad de recursos de sesiones disponible se determina por configuración.
- o La capacidad máxima de sesiones concurrentes que soportaría esta aplicación estaría limitada exclusivamente al número de servidores SNA disponibles, en número de PU's configuradas en dicho servidores, al número de LU's disponibles por PU, y al número de terminales configuradas en los POOL de terminales y que son accesibles directamente por el CommServer3270.

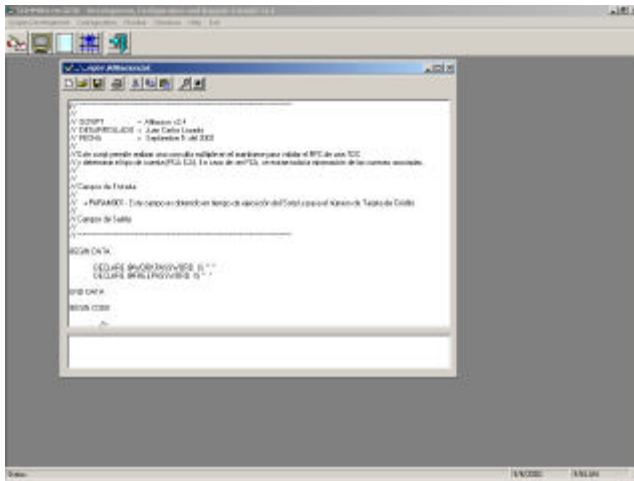
A continuación describimos algunas otras funciones implementadas en este servidor de comunicaciones 3270:

- o Validación del formato transaccional.
- o Autenticación de la aplicación cliente.
- o Establecimiento y administración de la sesión LU2 con el equipo Mainframe.
- o Interpretación del "script SNA" desarrollado para implementar la función de extracción de información deseada.
- o Procesamiento del formato 3270 Datastream para la extracción de los datos definidos, soportando la navegación y extracción de múltiples datos a través de diversas pantallas.
- o Conversiones recíprocas entre las representaciones de caracteres EBCDIC y ASCII.

- o Reporte del estado de la sesión a la consola de configuración y monitoreo.
- o Manejo de Excepciones y Recuperación de errores con reporte a la bitácora de eventos del sistema operativo.

Entorno de desarrollo de Scripts SNA

Una de las características fundamentales del producto consiste en poder acceder de forma discrecional información de diversas aplicaciones que se ejecutan en el Mainframe. La diversidad de la información requerida nos impone la necesidad de tener un mecanismo de extracción de datos que sea fácil y confiable. Razón por la cual, el producto dispone de un entorno integral para el desarrollo de "Scripts SNA", en el cual es posible desarrollar poderosos guiones de extracción de información, utilizando un lenguaje particularmente diseñado para la extracción de información, y que está orientado a la manipulación de pantallas.

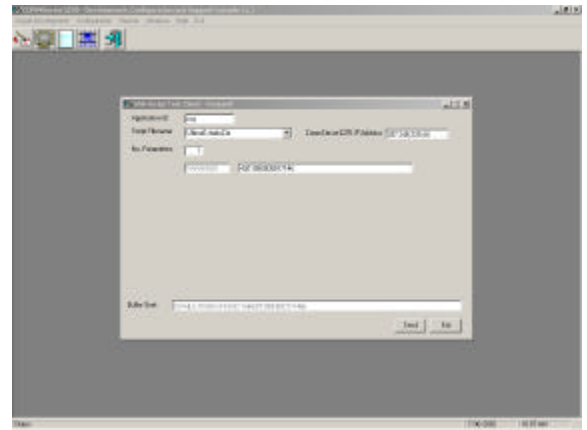


El proceso de desarrollo de "Scripts SNA" se desarrolla en dos fases.

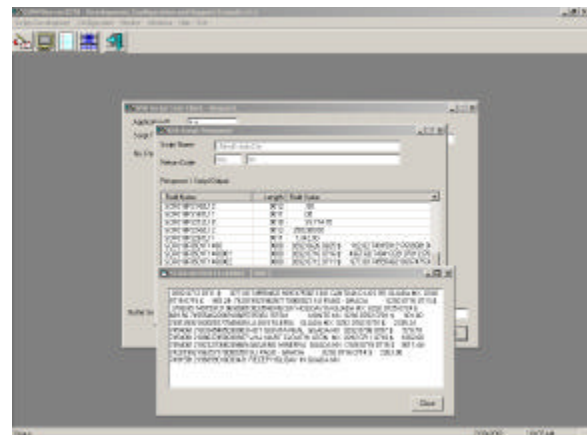
- o **Compilación:** se encargará de tomar un "Script SNA" en programa fuente, lo pasa por un proceso de análisis sintáctico y semántico, y en caso de ser válido, genera un programa adicional en pseudo-código (p-code). Con la finalidad de que pueda ser procesado de una forma más rápida y eficiente por el servidor de comunicaciones. Esta función está integrada a la consola de desarrollo, configuración y monitoreo del producto.

Ejecución: se encarga de tomar un programa de "Script SNA" en formato p-code, y lo ejecuta administrando su contexto, e interpretando las sentencias definidas en el script mediante la asociación con los correspondientes servicios aplicativos SNA soportados por el lenguaje.

Dentro de este entorno de desarrollo es posible ejecutar un "Script SNA" para efectos de pruebas, tal y como lo apreciamos en la siguiente pantalla.



Además es posible obtener la información resultante del proceso de extracción, con base a coordenadas de regiones de diversas pantallas:



Ejemplo de un Script SNA

A continuación presentamos un Script SNA que se encarga de extraer el último estado de cuenta de una aplicación de Tarjeta de Crédito que reside en un Mainframe IBM. En términos generales, podríamos mencionar que este script contiene desde la lógica de conexión y desconexión del equipo Host, como el proceso de ejecución de la aplicación CICS (Transacción IGB) y la extracción de información selectiva de diversas pantallas.

Este ejemplo es significativo, ya que nos permite mostrar las capacidades de navegación y extracción del producto, ya que la complejidad de esta consulta consiste, en que el Estado de Cuenta de una Tarjeta de Crédito está condicionada un número indeterminado de movimientos. Aún en estos casos, el producto CommServer3270 es capaz de resolver de forma eficiente una lógica de negocio tan compleja.

A continuación presentamos el citado ejemplo.

```

// *****
// SCRIPT          = Último Estado de Cuenta v2.3
// FECHA           = Julio 11 del 2002
// Este script permite extraer el último estado de cuenta de una Tarjeta de
// Crédito
// Campos de Entrada:
// + PARAM001 - Este campo es obtenido en tiempo de ejecución del Script y pasa el
// número de Tarjeta de Crédito
// *****
BEGIN DATA

  DECLARE @WORKPASSWORD IS " "
  DECLARE @FREEPASSWORD IS " "

END DATA

BEGIN CODE

  //--
  // Envía comando de conexión
  // --
  WAIT SESSION LULU
  SEND "SONI", @@ENTER

  //--
  // Envía clave de usuario y de password
  // --
  SEND @@PASSWORD, @@USERID, @@ENTER
  SET  @@PASSWORD TO @WORKPASSWORD
  SET  @@NEWPASSWORD TO @FREEPASSWORD

  //--
  // Si el password es inválido, entonces se intenta el SIGN ON con la
  segunda derivación
  // --
  GET SCREEN
  IF SCREEN = "PASSWORD ERROR - SIGN-ON REJECTED"
    SEND @CLEAR
    SEND "SONI", @@ENTER
    SEND @@NEWPASSWORD, @@USERID, @@ENTER
    SET  @@NEWPASSWORD TO @WORKPASSWORD
    SET  @@PASSWORD TO @FREEPASSWORD
    GET SCREEN

  ENDF

```

```

//--
// En este paso se entró al IBM exitosamente, entonces se almacena el
último PASSWORD existoso
// y se asocia a la terminal
// --
SET  @WORKPASSWORD TO @@LUPASSWORD

//--
// Manda la Transacción CICS IGB para obtener variaciones en
CLASIFICACION & TRANSCOD
// --
SEND "IGB", PARAM001, @@ENTER
GET SCREEN

//--
// Se regresa si el número de tarjeta de crédito no existe
// --
IF SCREEN = "ACCOUNT NUMBER DOES NOT SELF-CHECK"
  SEND @CLEAR
  SEND "SOFT", @@ENTER
  SET "NUMERO DE CUENTA NO EXISTE" TO
@@APPERRORMSG
  EXIT CODE
ENDIF

//--
// Se obtienen los datos deseados
// --
GET FIELD_NEXT "CLASIFICACION"
GET FIELD_NEXT "TRANSCOD"

SEND "IPS", @@ENTER

GET FIELD_POS 3,51,8
GET FIELD_POS 3,73,8

GET FIELD_POS 16,12,18
GET FIELD_POS 16,42,12
GET FIELD_POS 16,67,11

GET FIELD_POS 21,12,18
GET FIELD_POS 21,42,12
GET FIELD_POS 21,67,11

DO
GET SCREEN 5,1,14,80
  IF SCREEN <> LAST_SCREEN
    SEND @@PF8
  ENDF
WHILE SCREEN <> LAST_SCREEN
//--
// Se sale de la aplicación y se termina la sesión
// --
SEND @CLEAR
SEND "SOFT", @@ENTER

END CODE

```

Información de OPTIMA Technology

Para mayor información, favor de contactarnos en:

OPTIMA Technology, SA de CV
 Av. Revolución #973, Oficina 201
 Col. Mixcoac, C.P. 03910
 Tels: (52) 55 5615 5496/ 5615 5607
 Fax (52) 55 5615 5611
<http://www.optima.com.mx>
 Email: optima@optima.com.mx