



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

Z.106

(08/2002)

SERIE Z: LENGUAJES Y ASPECTOS GENERALES DE
SOPORTE LÓGICO PARA SISTEMAS DE
TELECOMUNICACIÓN

Técnicas de descripción formal – Lenguaje de
especificación y descripción

**Formato de intercambio común para el lenguaje
de especificación y descripción**

Recomendación UIT-T Z.106

RECOMENDACIONES UIT-T DE LA SERIE Z
**LENGUAJES Y ASPECTOS GENERALES DE SOPORTE LÓGICO PARA SISTEMAS DE
 TELECOMUNICACIÓN**

TÉCNICAS DE DESCRIPCIÓN FORMAL	
Lenguaje de especificación y descripción	Z.100–Z.109
Aplicación de técnicas de descripción formal	Z.110–Z.119
Gráficos de secuencias de mensajes	Z.120–Z.129
Lenguaje ampliado de definición de objetos	Z.130–Z.139
Notación combinada arborescente y tabular	Z.140–Z.149
Notación de requisitos de usuarios	Z.150–Z.159
LENGUAJES DE PROGRAMACIÓN	
CHILL: el lenguaje de alto nivel del UIT-T	Z.200–Z.209
LENGUAJE HOMBRE-MÁQUINA	
Principios generales	Z.300–Z.309
Sintaxis básica y procedimientos de diálogo	Z.310–Z.319
LHM ampliado para terminales con pantalla de visualización	Z.320–Z.329
Especificación de la interfaz hombre-máquina	Z.330–Z.349
Interfaces hombre-máquina orientadas a datos	Z.350–Z.359
Interfaces hombre-máquina para la gestión de las redes de telecomunicaciones	Z.360–Z.369
CALIDAD	
Calidad de soportes lógicos de telecomunicaciones	Z.400–Z.409
Aspectos de la calidad de las Recomendaciones relativas a los protocolos	Z.450–Z.459
MÉTODOS	
Métodos para validación y pruebas	Z.500–Z.519
SOPORTE INTERMEDIO	
Entorno del procesamiento distribuido	Z.600–Z.609

Para más información, véase la Lista de Recomendaciones del UIT-T.

Recomendación UIT-T Z.106

Formato de intercambio común para el lenguaje de especificación y descripción

Resumen

La presente Recomendación define el formato de intercambio común (CIF) del lenguaje de especificación y descripción (Rec. UIT-T Z.100 – SDL). El CIF está destinado al intercambio de especificaciones SDL gráficas (SDL-GR, *graphical SDL specifications*) preparadas con diferentes herramientas que no utilizan el mismo formato de almacenamiento. Antes de la definición del CIF, se utilizaba la representación textual del SDL (SDL-PR, *textual representation of SDL*) para intercambiar especificaciones, con el inconveniente de que toda la información gráfica se perdía y con el resultado de que frecuentemente las mismas especificaciones parecen muy diferentes en entornos diferentes. Con el CIF se minimiza este inconveniente, ya que contiene la mayor parte de la información gráfica. El CIF mejora la independencia con respecto a los vendedores de herramientas específicas y permite que los órganos de normalización acepten especificaciones en SDL-CIF prescindiendo de la herramienta que emplean en sus trabajos internos. Esto mejora también la productividad, al permitir que las especificaciones se confeccionen con las herramientas acostumbradas. Se insta a los vendedores de herramientas SDL a que proporcionen medios para importar y exportar el SDL-CIF.

Esta Recomendación define la forma de almacenar las descripciones SDL de modo que puedan ser intercambiadas entre herramientas procedentes de vendedores diferentes. No se tiene en cuenta la notación MSC. SDL-CIF es una ampliación de SDL. CIF está basada en la sintaxis SDL-PR, la representación de frases textuales de SDL, definida también en esta Recomendación. CIF puede ser leída y escrita por las herramientas y por los usuarios. Todos las construcciones disponibles en SDL pueden ser expresadas en forma gráfica o en la forma SDL-PR puramente textual. Las constricciones de la presentación gráfica se expresan en CIF añadiendo anotaciones específicas de SDL-PR. En consecuencia, la mayor parte de las descripciones SDL-PR son descripciones SDL-CIF legales. SDL-CIF es un formato de almacenamiento abierto, porque incluye un mecanismo de directrices específicas de la herramienta. Este mecanismo permite a una herramienta conforme a CIF ampliar el formato añadiendo información específica. SDL-CIF es también fácilmente realizable y proporciona a los vendedores de herramientas dos niveles de conformidad y los conceptos de directrices obligatorias y opcionales.

SDL-PR es una sintaxis alternativa de texto solamente para SDL. Antes de 2002, el SDL-PR se publicaba como una parte de la Recomendación Z.100, pero como el uso principal de esta notación es para comunicación dentro y entre herramientas, se ha trasladado la definición a la presente Recomendación. SDL-PR es el CIF de nivel 0 y permite intercambiar descripciones SDL sintácticamente completas, por lo general como un solo fichero por sistema. La conformidad con SDL-PR requiere que el modelo se conforme con la correspondiente semántica definida en la Rec. UIT-T Z.100.

La presente Recomendación introduce otros dos niveles de SDL-CIF. Se definen otros dos niveles de conformidad, uno en un nivel SDL-PR más liberal y el segundo que incluye información gráfica. Se describe la gramática completa con la semántica correspondiente. Se describen directrices obligatorias y opcionales, así como el formato para las directrices específicas de herramientas. En el apéndice I se describen las directrices específicas de las herramientas actuales.

Se definen dos niveles de conformidad CIF, a saber, nivel 1 y nivel 2. El nivel 1 está muy próximo del SDL-PR, pero admite especificaciones SDL incompletas. El nivel 2 incluye el nivel 1 y puede capturar la mayor parte de la información gráfica de los diagramas SDL-GR. Una especificación CIF debe identificar con cuál de los dos niveles se conforma. De manera similar, los vendedores de

herramientas que utilizan el CIF deben también identificar el nivel CIF que cumplen para sus funciones de importación y exportación.

Orígenes

La Recomendación UIT-T Z.106, revisada por la Comisión de Estudio 17 (2001-2004) del UIT-T, fue aprobada por el procedimiento de la Resolución 1 de la AMNT el 6 de agosto de 2002.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Asamblea Mundial de Normalización de las Telecomunicaciones (AMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

NOTA

En esta Recomendación, la expresión "Administración" se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT no ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

© UIT 2003

Reservados todos los derechos. Ninguna parte de esta publicación puede reproducirse por ningún procedimiento sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

1	Alcance	1
2	Referencias	1
3	Abreviaturas.....	1
4	Notación utilizada en esta Recomendación	1
5	Nivel 0 de CIF (SDL-PR).....	1
5.1	Principios generales.....	1
5.2	Reglas generales	3
5.3	Organización de especificaciones SDL	3
5.4	Conceptos estructurales	4
5.5	Agentes	7
5.6	Comunicación.....	9
5.7	Comportamiento	10
5.8	Data.....	16
5.9	Definición genérica de sistemas	17
6	Nivel 1 de CIF (CIF-PR)	18
6.1	Principios generales.....	18
6.2	Unidades transferibles de las especificaciones SDL	18
6.3	Sintaxis CIF-PR.....	18
7	Nivel 2 de CIF (CIF-GR).....	19
7.1	Principios generales.....	19
7.2	Principios generales de la información gráfica	19
7.3	Reglas léxicas CIF-GR.....	22
7.4	Sintaxis CIF/GR: Reglas A de CIF	23
7.5	Sintaxis CIF-GR – Reglas B de CIF:	45
7.6	Comentarios CIF específicos de herramienta:.....	51
8	Ejemplos	51
8.1	DemonGame.....	52
8.2	Construcciones SDL intrincadas	55
8.3	Situaciones que CIF no puede tratar.....	59
9	Criterios de conformidad CIF	60
9.1	Acerca de herramientas lectoras de un fichero CIF.....	60
9.2	Presentación automática y obligada	61
9.3	Retención y uso de información específica de herramienta	61

Apéndice I – Comentarios CIF específicos de herramienta.....	61
I.1 Mantenimiento de CIF.....	61
I.2 Comentarios CIF específicos de herramienta actuales.....	61

Introducción

Desde hace algunos años, el lenguaje de especificación y descripción (SDL) se utiliza cada vez más, tanto en la industria como para las normas y Recomendaciones. En la industria, el SDL se usa a menudo en un entorno con una sola herramienta SDL, mientras que para la elaboración de normas y Recomendaciones se requiere con frecuencia la integración de especificaciones SDL de muchas herramientas empleadas por diferentes organizaciones. A menudo, esto es también una exigencia en los proyectos internacionales.

Hasta el momento en que se ha propuesto esta Recomendación, el único medio para intercambiar especificaciones en SDL ha sido el SDL-PR (la representación textual de SDL), lo que ha resultado en la pérdida de información gráfica. Aunque no es necesaria desde el punto de vista formal, muchas veces la información gráfica ha tenido una repercusión importante en la legibilidad y comprensibilidad. Con el formato de intercambio común, esta Recomendación llena una necesidad desde hace tiempo reconocida para el intercambio de especificaciones SDL sin pérdida de la información gráfica.

Recomendación UIT-T Z.106

Formato de intercambio común para el lenguaje de especificación y descripción

1 Alcance

Esta Recomendación define el formato de intercambio común para especificaciones escritas en el lenguaje de especificación y descripción (SDL) [1]. Está destinada a los vendedores de herramientas como un formato de importación y exportación para permitir el intercambio de especificaciones SDL con herramientas ofrecidas por otros vendedores. Aunque el formato permite escribir especificaciones en CIF directamente, no es ésta su finalidad, sino que más bien deben ser generadas por una especificación SDL existente escrita de acuerdo con [1].

2 Referencias

Las siguientes Recomendaciones UIT-T y otras referencias contienen disposiciones que, mediante la referencia hecha en este texto, constituyen disposiciones de la presente Recomendación. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y otras referencias son objeto de revisiones, por lo que se aconseja a los usuarios de esta Recomendación que investiguen la posibilidad de aplicar las ediciones más reciente de las Recomendaciones y otras referencias citadas a continuación. Se publica periódicamente una lista de las Recomendaciones UIT-T actualmente vigentes. En esta Recomendación, la referencia a un documento, en tanto que autónomo, no le otorga el rango de una Recomendación.

[1] Recomendación UIT-T Z.100 (2002), *Lenguaje de especificación y descripción*.

3 Abreviaturas

En esta Recomendación se utilizan las siguientes abreviaturas.

CIF Formato de intercambio común (*common interchange format*)

SDL Lenguaje de especificación y descripción (*specification and description language*)

4 Notación utilizada en esta Recomendación

Para la definición de las propiedades y sintaxis del CIF se ha utilizado el metalenguaje definido en 5.4.2 de [1] con la siguiente adición. Cuando la primera palabra de un símbolo no terminal está separada de las siguientes palabras por dos puntos, este símbolo es una referencia a otra regla CIF. Los símbolos no terminales hacen referencia a no terminales de la Rec. UIT-T Z.100 o a no terminales introducidos en la presente Recomendación.

5 Nivel 0 de CIF (SDL-PR)

5.1 Principios generales

El nivel 0 de CIF introduce una forma sintáctica adicional que puede ser utilizada cuando se representa un sistema: la representación de frase textual (SDL-PR). SDL-PR se puede usar en vez de la representación gráfica de [1], que se denominará SDL-GR en este documento. Como ambas son representaciones concretas del mismo SDL, son equivalentes, en particular con respecto a la gramática abstracta para los conceptos correspondientes.

Un subconjunto de SDL-GR es común con SDL-PR. Este subconjunto se denomina gramática textual común.

Aunque el SDL puede ser escrito en SDL-PR o SDL-GR, el lenguaje ha sido diseñado con la certeza de que SDL-PR se utilizará con muy poca frecuencia para fines tales como el intercambio entre herramientas, aunque el formato de intercambio común especificado en el resto de este documento reduce aún más la utilización de SDL-PR. La mayoría de los usuarios emplean SDL-GR.

La figura 1 muestra las relaciones entre SDL-PR, SDL-GR, las gramáticas concretas y la gramática abstracta.

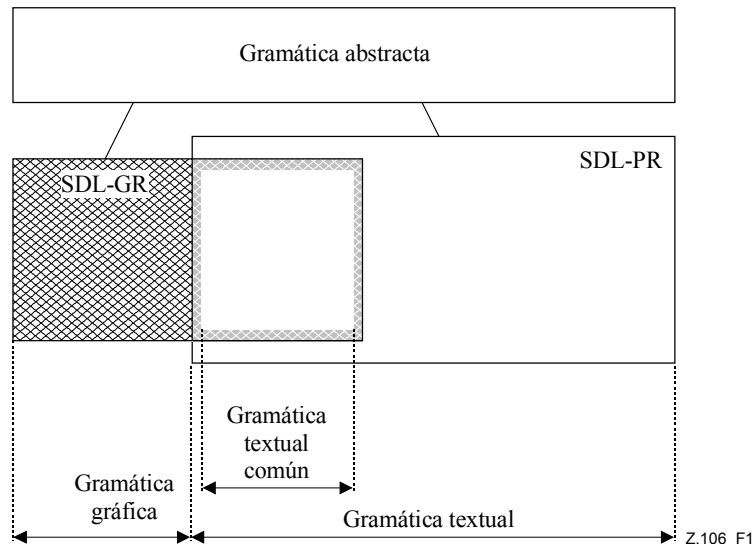


Figura 1/Z.106 – Gramáticas SDL

Cada una de las gramáticas concretas tiene una definición de su propia sintaxis y de su relación con la gramática abstracta (es decir, cómo efectuar la transformación en la sintaxis abstracta). Utilizando este enfoque, sólo hay una definición de la semántica de SDL; cada una de las gramáticas concretas hereda la semántica por su relación con la gramática abstracta, lo que asegura también que SDL-PR y SDL-GR son equivalentes.

Para algunas construcciones, no hay sintaxis abstracta directamente equivalente. En estos casos, se da un modelo para la transformación de la sintaxis concreta en la sintaxis concreta de otras construcciones que (directa o indirectamente por otros modelos) tienen una sintaxis abstracta. Los ítems que no tienen correspondencia con las sintaxis abstracta (tales como los comentarios), no tienen ningún significado formal.

La sintaxis textual concreta para SDL-PR se especifica en la forma Backus-Naur ampliada de la descripción de sintaxis definida en 5.4.2 de [1].

La sintaxis textual puede estar seguida por párrafos que definen las condiciones estáticas que deben ser satisfechas en un texto bien formado y que pueden ser verificadas sin interpretación de una instancia.

En muchos casos, hay una relación simple entre las sintaxis concreta y abstracta, porque la regla de la sintaxis concreta es representada simplemente por una sola regla en la sintaxis abstracta. Cuando se usa el mismo nombre en las sintaxis abstracta y concreta para significar que representa el mismo concepto, el texto "<x> en la sintaxis concreta representa X en la sintaxis abstracta" está implícito en la descripción del lenguaje y a menudo se omite. En este contexto, se pasa por alto si la letra es mayúscula o minúscula pero las subcategorías semánticas subrayadas son importantes (véase 5.4.2 de [1]).

Junto con las producciones para la sintaxis concreta de SDL-PR, se dan las secciones *Modelo* para las construcciones que son "sintaxis concreta derivada" para otras construcciones de sintaxis concreta equivalentes (véase 5.3.2 de [1]).

NOTA – En lo que sigue, los títulos de las cláusulas corresponden con los títulos de las cláusulas correspondientes de [1].

5.2 Reglas generales

En SDL-PR, el nombre o identificador opcional en una definición después de las palabras clave de fin (**endsystem**, **endblock**, etc.) en las definiciones debe ser sintácticamente igual al nombre o identificador que sigue a la palabra de comienzo correspondiente (**system**, **block**, etc., respectivamente).

5.3 Organización de especificaciones SDL

5.3.1 Marco

La regla de producción de arranque de la Rec. UIT-T Z.100, <sdl specification> (véase [1], 7.1), se sustituye con la siguiente producción:

```
<sdl specification>: :=
    { <package> | <textual system specification> } <package>* <referenced definition>*
    <package definition>
<textual system specification>: :=
    <agent definition>
    | { <package use clause>* } <textual typebased agent definition>
```

Modelo

Una <textual system specification> que es un <process definition> o un <textual typebased process definition> es sintaxis derivada para una <system definition> que tiene el mismo nombre que el proceso, contiene canales implícitos y contiene la <process definition> o <textual typebased process definition> como la definición única.

Una <textual system specification> que es una <block definition> o una <textual typebased block definition> es sintaxis derivada para una <system definition> que tiene el mismo nombre que el bloque, contiene canales implícitos y contiene la <block definition> o <textual typebased block definition> como la definición única.

Una <package use clause> antes de una <textual typebased agent definition> de una <textual system specification> es sintaxis derivada para una <package use clause> antes de <system heading> en la <system definition> derivada de la <textual typebased agent definition>.

5.3.2 Lote

```
<package definition>: :=
    { <package use clause>* }
    <package heading> <end>
    { <entity in package>* }
    endpackage [ <package name> ] <end>
```

```
<entity in package>: :=
    <agent type definition>
    | <agent type reference>
    | <package definition>
    | <package reference>
    | <signal definition>
    | <signal reference>
    | <signal list definition>
    | <remote variable definition>
    | <textual data definition>
    | <data definition>
```

- | <data type reference>
- | <textual procedure definition>
- | <procedure definition>
- | <procedure reference>
- | <remote procedure definition>
- | <composite state type definition>
- | <composite state type reference>
- | <exception definition>
- | <select definition>
- | <macro definition>
- | <interface reference>
- | <association>

5.3.3 Definición referenciada

La regla de producción de la Rec. UIT-T Z.100, <referenced definition> (véase [1], 7.3), se sustituye con la siguiente producción:

<referenced definition>: :=
 <definition>

<definition>: :=
 <package definition>
 | <agent definition>
 | <agent type definition>
 | <composite state>
 | <composite state type definition>
 | <textual procedure definition>
 | <procedure definition>
 | <textual operation definition>
 | <operation definition>
 | <macro definition>

5.4 Conceptos estructurales

5.4.1 Tipos, ejemplares y puertas

5.4.1.1 Definiciones de tipo estructural

5.4.1.1.1 Tipos de agente

<agent type definition>: :=
 <system type definition> | <block type definition> | <process type definition>

5.4.1.1.2 Tipo de sistema

<system type definition>: :=
 <package use clause>*
 <system type heading> <end> <agent structure>
 endsystem type [[<qualifier>] <system type name>] <end>

5.4.1.1.3 Tipo de bloque

<block type definition>: :=
 <package use clause>*
 <block type heading> <end> <agent structure>
 endblock type [[<qualifier>] <block type name>] <end>

5.4.1.1.4 Tipo de proceso

<process type definition>: :=
 <package use clause>*
 <process type heading> <end> <agent structure>
 endprocess type [[<qualifier>] <process type name>] <end>

5.4.1.2 Tipo de estado compuesto

<composite state type definition>: :=
 {<package use clause>}*
 { <composite state type heading> | <state aggregation type heading> } <end>
 <composite state structure>
 endsubstructure state type [[<qualifier>] <composite state type name>] <end>

5.4.1.2.1 Definiciones basadas en tipos

<textual typebased agent definition>: :=
 <textual typebased system definition>
 | <textual typebased block definition>
 | <textual typebased process definition>

El tipo de agente indicado por <base type> en la expresión de tipo de una <textual typebased agent definition> debe contener una transición de arranque no etiquetada en su máquina de estados.

5.4.1.2.2 Definición de sistema basada en tipo de sistema

<textual typebased system definition>: :=
 <typebased system heading> <end>

5.4.1.2.3 Definición de bloque basada en tipo de bloque

<textual typebased block definition>: :=
 block <typebased block heading> <end>

5.4.1.2.4 Definición de proceso basada en tipo de proceso

<textual typebased process definition>: :=
 process <typebased process heading> <end>

5.4.1.2.5 Definición de estado compuesto basada en tipo de estado compuesto

<textual typebased state partition definition>: :=
 state aggregation <typebased state partition heading> <end>

5.4.1.3 Referencias de tipos

La regla de producción de la Rec. UIT-T Z.100, <type referenced properties> (véase [1], 8.1.5), se sustituye con la siguiente producción:

<type reference properties>: :=
 [**with** { <attribute property> <end> }+]
 [**with** { <behaviour property> <end> }+]
 referenced <end>

5.4.1.4 Puerta

<gate in definition>: :=
 <textual gate definition> | <textual interface gate definition>

<textual gate definition>: :=

gate <gate> [**adding**] <gate constraint> [<gate constraint>] <end>

<textual interface gate definition>: :=

gate { **in** | **out** } **with** <interface identifier> <end>

adding sólo se puede especificar en una definición de subtipo y sólo para una puertitas definida en el supertipo. Cuando se especifica **adding** para una <gate>, cualesquiera <textual endpoint constraint> y <signal list> son adiciones a la <gate constraint> de la puertita en el supertipo.

5.4.2 Asociaciones

<association>: :=

association [<association name>] <association kind> <end>

<association kind>::=

| <association not bound kind>
| <association end bound kind>
| <association two ends bound kind>
| <composition not bound kind>
| <composition part end bound kind>
| <composition composite end bound kind>
| <composition two ends bound kind>
| <aggregation not bound kind>
| <aggregation part end bound kind>
| <aggregation aggregate end bound kind>
| <aggregation two ends bound kind>

<association not bound kind>: :=

from <association end> **from** <association end>

<association end bound kind>::=

from <association end> **to** <association end>

<association two ends bound kind>::=

to <association end> **to** <association end>

<composition not bound kind>::=

from <association end> **composition** <association end>

<composition part end bound kind>::=

to <association end> **composition** <association end>

<composition composite end bound kind>: :=

from <association end> **to composition** <association end>

<composition two ends bound kind>::=

to <association end> **to composition** <association end>

<aggregation not bound kind>::=

from <association end> **aggregation** <association end>

<aggregation part end bound kind>::=

to <association end> **aggregation** <association end>

<aggregation aggregate end bound kind>::=

from <association end> **to aggregation** <association end>

<aggregation two ends bound kind>::=

to <association end> **to aggregation** <association end>

<association end>: :=
 [<visibility>] [**as** <role name>] [**size** <multiplicity>] [**ordered**]
 { <agent type reference> | <interface reference> | <data type reference> }

Si un <association end> identifica a un tipo de agente o una interfaz, no se puede usar visibilidad **protected** en el otro <association end> de <association>.

Si dos <association> diferentes identifican al mismo tipo, en el <association end> opuesto a este tipo común los <role name> (si los hubiere) deben ser diferentes.

No debe haber un conjunto de <association> con composición de modo que un tipo esté vinculado por composición consigo mismo, directa o indirectamente.

Si un <association end> está precedido por la palabra clave **composition** e identifica a un tipo de datos o interfaz, el otro <association end> de <association> debe identificar un tipo de datos o interfaz, respectivamente.

5.5 Agentes

<agent definition>: :=
 <system definition> | <block definition> | <process definition>

<agent structure>: :=
 [<valid input signal set>]
 { { <entity in agent>
 | <channel to channel connection>
 | <channel definition>
 | <gate in definition>
 | <agent definition>
 | <agent reference>
 | <textual typebased agent definition> }*
 [<state partition>]
 | { <entity in agent>
 | <gate in definition> }*
 <agent body> }

La <state partition> debe tener el mismo nombre que el agente contenedor. Define la máquina de estados del agente. Si <agent structure> puede ser interpretada como <state partition> y <agent body>, es interpretada como <agent body>.

Una <channel to channel connection> no debe estar contenida dentro de una <agent type definition>.

<entity in agent>: :=
 <signal definition>
 | <signal reference>
 | <signal list definition>
 | <variable definition>
 | <remote procedure definition>
 | <remote variable definition>
 | <textual data definition>
 | <data definition>
 | <data type reference>
 | <timer definition>
 | <interface reference>
 | <macro definition>
 | <exception definition>
 | <procedure reference>

- | <textual procedure definition>
- | <procedure definition>
- | <composite state>
- | <composite state type definition>
- | <composite state type reference>
- | <select definition>
- | <agent type definition>
- | <agent type reference>
- | <association>

<agent body>: :=
 [[<on exception>] <start>]
 { <state> | <exception handler> | <free action> } *

5.5.1 Sistema

<system definition>: :=
 {<package use clause>}*
 <system heading> <end> <agent structure>
endsystem [<system name>] <end>

5.5.2 Bloque

<block definition>: :=
 {<package use clause>}*
 <block heading> <end> <agent structure>
endblock [[<qualifier>] <block name>] <end>

5.5.3 Proceso

<process definition>: :=
 {<package use clause>}*
 <process heading> <end> <agent structure>
endprocess [[<qualifier>] <process name>] <end>

5.5.4 Procedimiento

<textual procedure definition>: :=
 {<package use clause>}*
 <procedure heading> <end>
 <entity in procedure>*
 [<procedure body>]
endprocedure [[<qualifier>] <procedure name>] <end>

<entity in textual procedure>: :=
 <variable definition>
 | <textual data definition>
 | <data definition>
 | <data type reference>
 | <procedure reference>
 | <textual procedure definition>
 | <procedure definition>
 | <composite state type definition>
 | <composite state type reference>
 | <exception definition>
 | <select definition>
 | <macro definition>

<textual procedure body>: :=
[<on exception>] [<start>] { <state> | <exception handler> | <free action> }*

Un **endprocedure** dentro de una <textual procedure definition> interna en un <entity in textual procedure> de una <textual procedure definition> externa pertenece a la <textual procedure definition> interna.

5.6 Comunicación

5.6.1 Canal

<channel definition>: :=
channel [<channel name>] [**nodelay**]
 <channel path> [<channel path>]
endchannel [<channel name>] <end>

<channel path>: :=
from <channel endpoint>
to <channel endpoint> [**with** <signal list>] <end>

<channel endpoint>: :=
{ <agent identifier> | <state identifier> | **env** | **this** } [<via gate>]

<via gate>: :=
via <gate>

El <channel name> final solo puede ser especificado si se especifica el <channel name> inicial- Si no se especifica el <channel name> inicial, no se puede hacer referencia al canal por su nombre.

El <channel endpoint> **this** indica la máquina de estados del agente que encierra directamente la definición de canal.

Se debe especificar <gate> si:

- a) <channel endpoint> indica una conexión a una <textual typebased agent definition> o a una <textual typebased state partition definition> en cuyo caso <gate> debe ser definida directamente en el tipo de agente o tipo de datos para ese agente o estado, respectivamente, o
- b) se especifica **env** y el canal está definido en un tipo de agente, en cuyo caso la <gate> debe estar definida en este tipo de agente respectivamente.

Si se especifica <gate>, el canal está conectado a esa puerta. La puerta y el canal deben tener por lo menos un elemento común en su lista de señales en el mismo sentido. Si no se especifica <gate>, se aplica lo siguiente: si el punto extremo de canal es un agente o máquina de estado y ese agente/estado contiene una <channel to channel connection> para el canal, el canal se conecta a la puerta implícita presentada por <channel to channel connection>. En los demás casos, si el punto extremo de canal es un estado, el canal se conecta a la puerta implícita de esa máquina de estados, el canal introduce una puerta implícita en el agente o esta mencionado en <channel endpoint>. Esta puerta obtiene la <signal list> del respectivo <channel path> como su correspondiente constricción de puerta. El canal se conecta a esa puerta.

Si una <channel definition> contiene dos <channel path>, se deben satisfacer las siguientes condiciones:

- a) El <channel endpoint> **from** el primer <channel path> debe ser igual que el <channel endpoint> siguiente **to** en el segundo <channel path>.
- b) El <channel endpoint> **to** que sigue en el primer <channel path> debe ser igual que el <channel endpoint> siguiente **from** en el segundo <channel path>.

5.6.2 Conexión

<channel to channel connection>: :=
 connect <external channel identifiers>
 and <channel identifiers> <end>

<channel identifiers>: :=
 <channel identifier> {, <channel identifier>}*

Ningún canal puede ser mencionado después de la palabra clave **and** en más de una <channel to channel connection> de una unidad de alcance dada.

Para cualquier par de <channel to channel connection> de una unidad de alcance dada, los <external channel identifiers> mencionarán el mismo conjunto de canales, o no tendrán canales en común. Si dos o más <channel to channel connection> de una unidad de alcance dada tienen el mismo conjunto de canales externos, ésta es sintaxis derivada para una <channel to channel connection> que tiene uno de los <external channel identifiers> como su <external channel identifiers>, y su <channel identifiers> formado por la enumeración de todos los <channel identifier> internos.

NOTA – Debido a la construcción **connect**, un canal (externos) que puede ser anónimo en la versión gráfica de una especificación, puede necesitar un nombre en la versión textual correspondiente. Esto es completamente análogo al caso de <merge area> en gráficos de proceso o procedimientos. Una herramienta que convierte la versión gráfica de una especificación en una versión textual debe por tanto ser capaz de generar nombres de canales implícitos.

Modelo

Cada <channel to channel connection> diferente en una unidad de alcance dada define una puerta implícita en la unidad de alcance. Todos los canales en la <channel to channel connection> están conectados a esa puerta en sus respectivas unidades de alcance. Las constricciones de puerta de la puerta implícita se derivan de los canales conectados a la puerta.

El nombre de la puerta es un nombre derivado único e inequívoco. En la unidad de alcance vecina la <channel definition> que es identificada por el <channel identifier> es ampliada con una parte de <via gate>. La parte <via gate> se añade al <channel endpoint> que hace referencia a la unidad de alcance vigente y menciona la puerta implícita. Dentro de la unidad de alcance, los canales que están asociados con el canal externo por medio de <channel to channel connection> son modificados, ampliando el <channel endpoint> que menciona **env** con una parte <via gate> para la puerta implícita.

5.7 Comportamiento

5.7.1 Arranque

<start>: :=
 start [<virtuality>] [<state entry point name>] <end> [<on exception>] <transition>

Si se da <state entry point name> en un <start>, el <start> debe ser el <start> de un <composite state>.

5.7.2 Estado

`<state>`: :=
 state `<state list>` `<end>` [`<on exception>`]
 { `<input part>`
 | `<priority input>`
 | `<save part>`
 | `<spontaneous transition>`
 | `<continuous signal>`
 | `<connect part>` }
 [**endstate** [`<state name>`] `<end>`]

El `<state name>` opcional que termina un `<state>` puede ser especificado solamente si la `<state list>` en el `<state>` consta de un solo `<state name>`, en cuyo caso debe ser ese `<state name>`.

La `<connect part>` se permite solamente para un `<state>` con `<state list>` que contiene un `<composite state item>`.

5.7.3 Entrada

`<input part>`: :=
 input [`<virtuality>`] `<input list>` `<end>`
 [`<on exception>`] [`<enabling condition>`] `<transition>`

5.7.4 Entrada prioritaria

`<priority input>`: :=
 priority input [`<virtuality>`]
 `<priority input list>` `<end>` [`<on exception>`] `<transition>`

5.7.5 Señal continua

`<continuous signal>`: :=
 provided [`<virtuality>`]
 `<continuous expression>` `<end>`
 [**priority** `<priority name>` `<end>`] [`<on exception>`] `<transition>`

5.7.6 Condición habilitadora

`<enabling condition>`: :=
 provided `<provided expression>` `<end>`

5.7.7 Conservar

`<save part>`: :=
 save [`<virtuality>`] `<save list>` `<end>`

5.7.8 Transición espontánea

`<spontaneous transition>`: :=
 input [`<virtuality>`] `<spontaneous designator>` `<end>`
 [`<on exception>`] [`<enabling condition>`] `<transition>`

5.7.9 Etiqueta

`<label>`: :=
 `<connector name>`:

NOTA – En la gramática abstracta, solo las acciones libres tienen etiquetas; las etiquetas dentro de una transición son transformadas en acciones libres separadas.

<free action>: :=
connection
 <transition>
 [**endconnection** [<connector name>] <end>]

Si la <transition string> de la <transition> en <free action> no está vacía, la primera <action> debe tener una <label> o en los demás casos el <terminator> debe tener una <label>. Si está presente, el <connector name> que termina la <free action> debe ser igual que el <connector name> en esta etiqueta <label>.

Modelo

Si una <label> no es la primera etiqueta de una <transition string>, la <transition string> se divide en dos partes. Todas las <action> que preceden a la <label> son preservadas en la transición original, que es terminada con <join> a la <label>. Todas las declaraciones de acciones que siguen a <label> son copiadas en una nueva <free action>, que comienza con la <label>.

5.7.10 Máquina de estados y estado compuesto

<composite state>: :=
 <composite state graph> | <state aggregation>

5.7.10.1 Gráfico de estado compuesto

<composite state graph>: :=
 {<package use clause>}*
 <composite state heading> <end> <composite state structure>
endsubstructure [[<qualifier>] <composite state name>] <end>

<composite state structure>: :=
 [<valid input signal set>]
 {<gate in definition>}*
 <state connection points>*
 <entity in composite state>*
 { <composite state body> | <state aggregation body> }

Una <composite state structure> contendrá un <state aggregation body> solamente si está directamente contenida en una <state aggregation> o una <composite state type definition> con un <state aggregation type heading>, en los demás casos contiene un <composite state body>. Una <composite state structure> que contiene un <state aggregation body> no tendrá un <valid input signal set>.

<entity in composite state>: :=
 <variable definition>
 | <textual data definition>
 | <data definition>
 | <select definition>
 | <data type reference>
 | <macro definition>
 | <textual procedure definition>
 | <procedure definition>
 | <procedure reference>
 | <exception definition>
 | <composite state>
 | <composite state type definition>
 | <composite state type reference>

<composite state body>: :=
[<on exception>] <start>* { <state> | <exception handler> | <free action> }*

5.7.10.2 Agregación de estados

<state aggregation>: :=
{<package use clause>}*
<state aggregation heading> <end> **<composite state structure>**
endsubstructure [[<qualifier>] <composite state name>] <end>

<state aggregation body>: :=
{ <state partition>
| <state partition connection> }*

<state partition>: :=
<textual typebased state partition definition>
| <composite state reference>
| <composite state>

<composite state reference>: :=
state substructure <composite state name> **referenced** <end>

<state partition connection>: :=
connect <outer entry point> **and** <inner entry point> <end>

<outer entry point>: :=
<composite state identifier> **via** <point>

<inner entry point>: :=
<composite state identifier> **via** <point>

5.7.10.3 Punto de conexión de estados

<state connection points>: :=
{ **in** <state entry points> | **out** <state exit points> } <end>

5.7.10.4 Conectar

<connect part>: :=
connect [<virtuality>] [<connect list>] <end>
[<on exception>] <exit transition>

<exit transition>: :=
<transition>

5.7.11 Transición

5.7.11.1 Cuerpo de transición

<transition>: :=
{<transition string> [<terminator>] }
| <terminator>

<transition string>: :=
{<action>}+

<action>: :=
[<label>]
{
| <task>
| <output>
| <create request>
| <decision>

```

| <set>
| <reset>
| <export>
| <procedure call>
| <remote procedure call>
| <transition option> } <end>

```

```

<terminator>: :=
  [<label>]
  {
  | <return>
  | <raise>
  | <nextstate>
  | <join>
  | <stop> } <end>

```

Si el <terminator> de una <transition> se omite, la última acción en la <transition> debe contener una <decision> terminadora (véase 5.7.12.2) o una <transition option> terminadora, salvo cuando una <transition> está contenida en una <decision> o <transition option>.

5.7.11.2 Terminador de transición

5.7.11.2.1 Estado siguiente

```

<nextstate>: :=
  nextstate <nextstate body>

```

5.7.11.2.2 Unir

```

<join>: :=
  join <connector name>

```

A <join> corresponds to an <out connector area> in SDL-GR.

Debe haber exactamente un <connector name> correspondiente a <join> dentro del mismo cuerpo. La regla para <agent body> en una definición de tipo se indica en [1], 8.3.1.

5.7.11.2.3 Parar

```

<stop>: :=
  stop

```

5.7.11.2.4 Retornar

```

<return>: :=
  return [<return body>] [ <end> <on exception> ]

```

5.7.11.2.5 Resaltar

```

<raise>: :=
  raise <raise body>

```

5.7.12 Acción

5.7.12.1 Tarea

```

<task>: :=
  task [ <comment body> ] <left curly bracket> <task body> <right curly bracket>
  [ <end> <on exception> ]

```

5.7.12.2 Crear

```

<create request>: :=

```

create <create body> [<end> <on exception>]

5.7.12.3 Llamada de procedimiento

<procedure call>: :=
call <procedure call body>

5.7.12.4 Salida

<output>: :=
output <output body> [<end> <on exception>]

5.7.12.5 Decisión

<decision>: :=
decision <question> <end> [<on exception>]
 <textual decision body>
enddecision

<textual decision body>: :=
 <textual answer part>+ [<textual else part>]

<textual answer part>: :=
 ([<answer>]) <colon> [<transition>]

<textual else part>: :=
 else <colon> [<transition>]

Una <textual answer part> o <textual else part> en una decisión o una opción de transición es una <textual answer part> o <textual else part> terminadora respectivamente si contiene una <transition> donde se especifica un <terminator> o contiene una <transition string> cuya última <action> contiene una decisión u opción terminadora. Una <decision> o <transition option> es una decisión terminadora y una opción de transición terminadora, respectivamente, si cada <textual answer part> y <textual else part> en su <textual decision body> es una <textual answer part> o <textual else part> terminadora, respectivamente.

Se debe omitir <answer> de <textual answer part> solamente si la <question> consiste en la palabra clave **any**. En este caso, no puede estar presente <textual else part>.

Si una <decision> no es terminadora, es sintaxis derivada para una <decision> donde todas las <textual answer part> y <textual else part> no terminadoras (si no hay terminación) han insertado al final de su <transition> un <join> a la primera <action> que sigue a la decisión o (si la decisión es la última <action> en una <transition string>) al siguiente <terminator>.

5.7.13 Temporizador

<reset>: :=
reset <reset body> [<end> <on exception>]

<set>: :=
set <set body> [<end> <on exception>]

A <reset> represents a *Reset-node*; a <set> represents a *Set-node*.

5.7.14 Excepción

5.7.14.1 Manejador de excepciones

<exception handler>: :=
exceptionhandler <exception handler list> <end>
 [<on exception>]
 <handle>*

[**endexceptionhandler** [<exception handler name>] <end>]

5.7.14.2 Excepción activa

<on exception>: :=
 onexception <exception handler name>

5.7.14.3 Manejar

<handle>: :=
 handle [<virtuality>] <exception stimulus list> <end>
 [<on exception>] <transition>

5.8 Data

5.8.1 Data definitions

<textual data definition>: :=
 <textual data type definition>
 | <textual interface definition>
 | <textual syntype definition>

5.8.1.1 Definición de tipos de datos

<textual data type definition>: :=
 {<package use clause>}*
 <type preamble> <data type heading> [<data type specialization>]
 <end> [<data type definition body> <data type closing> <end>]

<data type closing>: :=
 { **endvalue** | **endobject** } **type** [<data type name>]

La palabra clave **value** u **object** y <data type name> en <data type closing> debe ser concordada por las palabras clave **endvalue** o **endobject**, respectivamente, y nombrada en el correspondiente <data type heading>, si está presente.

5.8.1.2 Definición de interfaces

<textual interface definition>: :=
 {<package use clause>}*
 [<virtuality>] <interface heading>
 [<interface specialization>]
 <end> <entity in interface>* [<interface use list>]
 <interface closing>
 | {<package use clause>}*
 [<virtuality>] <interface heading>
 [<interface specialization>] <end>

<interface closing>: :=
 endinterface [<interface name>] <end>

5.8.1.3 Comportamiento de las operaciones

<textual operation definition>: :=
 {<package use clause>}*
 <operation heading> <end>
 <entity in textual operation>*
 <operation body>
 { **endoperator** | **endmethod** } [[<qualifier>] <operation name>] <end>

<entity in textual operation>: :=


```

    <textual data definition>
  | <data definition>
  | <variable definition>
  | <exception definition>
  | <select definition>
  | <macro definition>

```

```

<operation body>: :=
    [<on exception>] <start> { <free action> | <exception handler> }*

```

5.8.1.4 Sintipos

```

<textual syntype definition>: :=
    {<package use clause>}*
    syntype <syntype name> <equals sign> <parent sort identifier>
        { { <default initialization> [ [<end>] [<constraint> ] | <constraint> } <end>]
          <syntype closing>
        | [<syntype closing>] } <end>
  | {<package use clause>}*
    <type preamble> <data type heading> [<data type specialization>] <end>
    <data type definition body> <constraint> <end> <data type closing> <end>

```

```

<syntype closing>: :=
    endsyntype [<syntype name>]

```

5.9 Definición genérica de sistemas

5.9.1 Definición opcional

```

<select definition>: :=
    select if ( <Boolean simple expression> ) <end>
        {
          <agent type definition>
        | <agent type reference>
        | <agent definition>
        | <agent reference>
        | <channel definition>
        | <signal definition>
        | <signal list definition>
        | <signal reference>
        | <remote variable definition>
        | <remote procedure definition>
        | <textual data definition>
        | <data definition>
        | <data type reference>
        | <interface reference>
        | <composite state type definition>
        | <composite state type reference>
        | <state partition>
        | <timer definition>
        | <variable definition>
        | <textual procedure definition>
        | <procedure definition>
        | <procedure reference>
        | <channel to channel connection>
        | <select definition>

```

```

    | <macro definition>
    | <exception definition> }+
endselect <end>

```

5.9.2 Cadena de transición opcional

```

<transition option>: :=
    alternative <alternative question> <end>
    <textual decision body>
endalternative

```

6 Nivel 1 de CIF (CIF-PR)

6.1 Principios generales

El nivel 1 de CIF es una relajación de la sintaxis SDL-PR: no aporta ninguna información adicional acerca de la representación gráfica, y solamente mejora la idoneidad de SDL-PR como formato de intercambio.

Supera el inconveniente principal de SDL-PR como formato de intercambio, que sólo describe sintácticamente descripciones SDL completas, siendo así que es necesario intercambiar descripciones que son parciales o que no se han obtenido todavía. Sin embargo estas descripciones deben ser sintácticamente correctas para poder ser intercambiadas.

SDL-CIF reutiliza grandes partes de la sintaxis SDL-PR. Las reglas de producción de SDL-PR que son reutilizadas en SDL-CIF después son exactamente referenciadas por sus nombres, y no se describen de nuevo en esta Recomendación.

Las reglas Z.100 para transformar SDL-PR en la gramática abstracta se aplican también a las partes de SDL-CIF compartidas con SDL-PR, en la mayor medida posible de acuerdo con el grado de completación de las descripciones SDL-CIF.

6.2 Unidades transferibles de las especificaciones SDL

6.3 Sintaxis CIF-PR

6.3.1 Fichero CIF

La regla de producción de arranque de SDL-PR, <sdl specification> (véase 5.3.1), se sustituye por la producción siguiente:

```

<cif level 1 file>: :=
    { <textual system specification>
      | <definition>
    } *

```

6.3.2 Llamada a (de) macro

En SDL las llamadas a (de) macro pueden "aparecer en cualquier lugar en que esté autorizada una <lexical unit>" ([1], 6.2.3).

En SDL/CIF las llamadas a (de) macro sólo pueden aparecer en el lugar de una tarea.

Se sustituye la regla de producción <task> (véase 5.7.12.1) por :

```

<task>: :=
    task <textual task body> [ <end> <on exception> ]
    | <macro call>

```

7 Nivel 2 de CIF (CIF-GR)

7.1 Principios generales

El nivel 2 de CIF es una ampliación del nivel 1 de CIF con las denominadas "directrices CIF" que describen las principales características de la presentación gráfica de objetos.

Las directrices CIF se colocan antes del objeto asociado con ellas. Al definir esta Recomendación se ha utilizado el siguiente principio de diseño: todas las construcciones SDL-PR que contienen información que el convertidor CIF tiene que extraer, deben tener un comentario CIF asociado colocado antes de la construcción SDL-PR. Esto permite a una herramienta de lectura CIF explorar el siguiente comentario CIF, extraer la información necesaria del SDL-PR siguiente y a continuación iniciar la búsqueda del siguiente comentario CIF.

Varias directrices CIF pueden estar asociadas con el mismo objeto.

La primera directriz CIF para un objeto describe normalmente la presentación de la parte principal del objeto, mientras que las directrices CIF siguientes para el mismo objeto describen la presentación de subpartes del objeto.

El nivel 2 de CIF no describe todos los detalles de la presentación, ya que esto restringiría demasiado el número de herramientas capaces de sustentar SDL-CIF: algunos editores SDL prefieren la presentación manual (usuario) de símbolos, mientras que otros prefieren la presentación automática. El tratamiento de ambas presentaciones manual y automática es un problema complejo que es difícil de resolver cuando se elabora una herramienta SDL.

Para tratar este asunto, las directrices CIF se clasifican en tres categorías: directrices obligatorias, opcionales y específicas de la herramienta.

Las directrices obligatorias describen características gráficas que no pueden ser computadas automáticamente, o cuya computación automática estaría ciertamente demasiado alejada de las expectativas del usuario, por ejemplo, la presentación de símbolos y líneas en diagramas de interconexión.

Estas características gráficas son controladas generalmente por el usuario en las principales herramientas de edición SDL.

Las directrices opcionales describen las características gráficas que pueden ser computadas automáticamente, por ejemplo la presentación de texto dentro de símbolos. Para cualquier información opcional que no está dada, las herramientas deben computar automáticamente una presentación.

Las directrices específicas de herramienta describen características (gráficas o no) que no están cubiertas por las directrices obligatorias u opcionales. Ellas permiten a los fabricantes de herramientas añadir nuevas directrices CIF en el formato de almacenamiento que serán analizadas por sus propias herramientas solamente.

7.2 Principios generales de la información gráfica

7.2.1 El sistema de coordenadas

La unidad utilizada es 1/10 mm. El origen es la esquina superior izquierda de la página. El eje positivo de la x está a la derecha del origen. El eje positivo de la y está debajo del origen.

7.2.2 Páginas

Los diagramas se pueden dividir en páginas, como se describe en 6.6 [1].

Sin embargo, los ficheros SDL-CIF no están estructurados en páginas, sino de acuerdo con la sintaxis SDL-PR. Las páginas se describen mediante comentarios CIF insertados entre algunas

unidades de sintaxis. Una página puede estar formada por información procedente de varias unidades de sintaxis no adyacentes.

Las páginas son áreas de dibujo independientes. Cada coordenada debe interpretarse de acuerdo con el nombre de la página vigente.

7.2.3 Clasificación de la información

La información gráfica CIF se puede clasificar en cuatro clases:

- información sobre símbolos que parecen líneas gráficas, normalmente llamadas "líneas", es decir, canales, líneas de flujo en transiciones, y líneas de asociación;
- información sobre símbolos que no parecen líneas, normalmente llamados "símbolos", por ejemplo, símbolos de proceso, símbolos de salida;
- información sobre texto;
- otra información, por ejemplo, información sobre división de páginas.

7.2.4 Representación de símbolos

Toda información sobre posiciones y tamaños de símbolos es información obligatoria.

La posición de un símbolo viene normalmente dada por las coordenadas del ángulo superior izquierdo de su caja limitadora. Hay algunas excepciones en las que en vez del ángulo izquierdo se utiliza el ángulo superior derecho (para símbolos de ampliación de textos invertidos y símbolos de comentarios invertidos).

El tamaño de un símbolo viene dado por la anchura y la altura de su caja limitadora.

A veces se añade información específica del símbolo. Tal es el caso de símbolos que se encuentran en las dos versiones derecha e izquierda, y puede estar presente una palabra clave "Left" o "Right".

7.2.5 Representación de texto

Las posiciones y tamaños del texto es información opcional. Esta información se refiere a la caja limitadora del texto y no al texto en sí.

7.2.6 Acerca de las posiciones de texto opcionales

La especificación de clases diferentes de posiciones de texto es opcional en CIF. Esto significa que una herramienta no tiene que especificar una posición de texto cuando se escribe un fichero CIF. Significa también que una herramienta que lee un fichero CIF con una posición de texto especificada no tiene que utilizar esa posición de texto. A continuación se dan algunas directrices:

Cuando se lee un fichero CIF, una herramienta debe intentar utilizar las posiciones de texto encontradas en el fichero CIF. Si esto no es posible, utilizará en su lugar autopresentación.

Si una posición de texto no está dada en el fichero CIF, se debe usar autopresentación.

Algunas posiciones de texto son más importantes que otras para la presentación SDL-GR original de un diagrama en CIF. Un fabricante de herramientas debe concentrar su atención en la realización de soporte para estas primeras posiciones de texto. A continuación se enumeran agrupadas las posiciones de texto. Las posiciones de texto que es más importantes preservar se encuentran en el primer grupo.

- Grupo 1: nombre de canal, lista de señales, nombre de puerta, seleccionar.
- Grupo 2: conectar, línea de flujo de respuesta, referencia de puerta, retornar.
- Grupo 3: encabezamiento de núcleo de diagrama, nombre de página, símbolo de sistema, símbolo de bloque, símbolo de proceso, símbolo de procedimiento, símbolo de operador, estado, conservación, tarea, fijar, reiniciar, crear, llamada a procedimiento, arranque de procedimiento, decisión, señal continua, condición habilitadora, opción de transición,

unión, etiqueta, llamada a (de) macro, acceso de salida de macro, entrada, entrada prioritaria, salida, texto, referencia de paquete.

7.2.7 Representación de línea

La presentación de líneas se da mediante una lista de coordenadas: una para el punto de arranque de la línea, una para cada punto de corte en la línea y una para el punto de final de la línea.

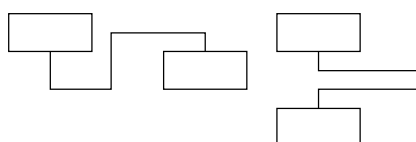
El arranque y los puntos extremos de una línea están conectados normalmente a otros símbolos.

Si un punto de arranque o de final se conecta a un símbolo que no tiene forma de rectángulo, el punto se sitúa en la caja limitadora del símbolo en lugar de hacerlo sobre el perfil del símbolo real, para simplificar los cálculos de geometría.

La presentación de líneas de canal es una información obligatoria, puesto que resulta imposible adivinar qué es lo que desea ver el usuario.

7.2.8 Acerca de las líneas de flujo opcionales

Las líneas de flujo que no siguen inmediatamente después de un símbolo de decisión, son opcionales en el mismo sentido que las posiciones de texto. Algunas orientaciones:



Es más importante dar información acerca de líneas complicadas como las dos líneas de flujo anteriores.



Es menos importante dar información acerca de líneas de flujo sencillas como las dos líneas de flujo anteriores.

Z.106_F2

Figura 2/Z.106

7.2.9 Información gráfica no cubierta por CIF

Las directrices de CIF se refieren principalmente a tamaños y posiciones gráficas, porque es una información universal que puede ser intercambiada sin problemas de realización, y porque es una información que haría muy difícil dibujar de nuevo los diagramas en caso de pérdida.

Algunas otras clases de información se han considerado menos importantes, y no son cubiertas por SDL-CIF. Son informaciones sobre tipos de caracteres de texto, tamaño de caracteres, color y espesor de línea.

Pueden ser dadas por medio de directrices específicas de herramienta.

7.2.10 Acerca de los diagramas anidados

CIF admite los SDL anidados (diagramas dentro de diagramas). Las herramientas que no admiten SDL anidados pueden convertir un diagrama dentro de un diagrama a un símbolo de referencia y un diagrama independiente:

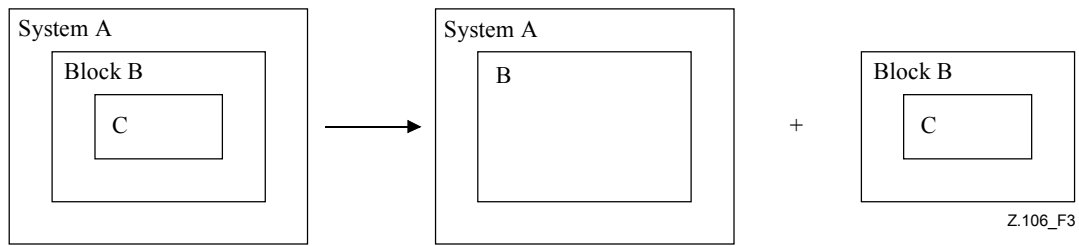
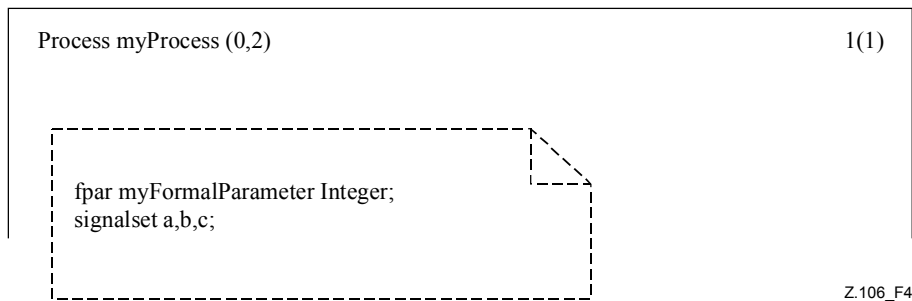


Figura 3/Z.106

7.2.11 Acerca de los encabezamientos de núcleo y adicional

CIF no admite encabezamientos adicionales. Por ello las herramientas que soportan encabezamientos adicionales han de hacer la partición del texto de encabezamiento en un encabezamiento de núcleo y un encabezamiento adicional sin soporte por CIF. Obsérvese que SDL-GR en el ejemplo anterior no es correcto, puesto que de acuerdo con la Rec. UIT-T Z.100, no debería haber un símbolo alrededor del texto de encabezamiento adicional.



Texto de encabezamiento dividido en encabezamiento de núcleo y encabezamiento adicional.

7.3 Reglas léxicas CIF-GR

7.3.1 Directrices CIF

Las directrices CIF son formas especiales de los comentarios <note> de Z.100 todas las cuales tienen en común la siguiente descripción:

```
<cif directive>: :=
    /* { CIF | cif } <text> */
```

Una línea fuente en el fichero analizado, que contiene una directiva CIF, no debe contener ningún otro testigo (token).

Una <note> CIF-GR no debe ser una <cif directive>.

7.3.2 Caracteres cambio de renglón y espacio

Se suele considerar que los caracteres cambio de renglón y espacio no son caracteres significativos cuando se encuentran durante el análisis del fichero CIF, y son pasados por alto.

Sin embargo, cuando dos testigos SDL adyacentes se presentan en un diagrama como partes adyacentes de un objeto de texto, los caracteres cambio de renglón y espacio entre los dos testigos no deben ser pasados por alto: deben ser utilizados como parte del objeto de texto.

Esto permite que las herramientas mantengan la preferencia del usuario para la presentación del texto.

Cuando dos testigos SDL de un objeto de texto están separados por algunos caracteres espacio y un cambio de renglón seguido de varios caracteres espacio, los caracteres espacio antes del primer carácter significativo de la segunda línea debe ser pasado por alto, porque son espacios de sangrado.

El primer carácter espacio significativo de una línea es el carácter de la misma columna y primer carácter "/" de la directiva CIF anterior.

Por ejemplo, en el siguiente fragmento SDL-CIF:

```
BLOCK b;  
  /* CIF Channel (500,400), (300,400) */  
  CHANNEL r FROM ENV TO P WITH s1 , s2  
    s3;  
ENDCHANNEL;
```

El texto que debe ser presentado para la lista de señales es:

```
's1 , s2' // NL // ' s3'
```

(si no hay caracteres espacio después de "s2").

7.3.3 Acerca de la presentación de texto

El emplazamiento de caracteres cambio de renglón en SDL-GR debe conservarse en SDL-PR, es decir, una lista de señales en SDL-GR con dos señales en dos líneas debe tener la SDL-PR:

```
FROM ENV TO P WITH KeyStroke,  
Card;
```

en lugar de:

```
FROM ENV TO P WITH KeyStroke, Card;
```

7.4 Sintaxis CIF/GR: Reglas A de CIF

Las reglas A de CIF describen las directrices CIF que están asociadas con diferentes construcciones SDL-PR. Usualmente hay una regla para cada construcción SDL-PR que corresponde a un símbolo gráfico. Sin embargo, en algunos casos un símbolo se describe con más de una regla porque no toda la información en SDL-PR se recoge en un lugar. Un ejemplo es el marco de diagrama, donde el comienzo y el final de un diagrama se describen mediante reglas separadas.

Las reglas B de CIF son reglas de utilidad referenciadas a partir de las reglas A de CIF y no corresponden directamente con símbolos en los diagramas.

Una regla normal de CIF se describe en general como sigue:

- Una sección que describe la gramática para el comentario CIF y para la información SDL-PR correspondiente. Esta sección muestra también como deben colocarse los comentarios CIF en la especificación SDL-PR.
- Una sección que explica texto y un ejemplo.

Obsérvese que esta gramática solamente da instrucciones sobre cómo insertar comentarios CIF en SDL-PR. Un fichero CIF de nivel 2 sólo es correcto si satisface la gramática CIF de nivel 2 (para la definición de comentarios CIF) y la gramática CIF de nivel 1.

7.4.1 A1 Descripción de CIF:

```
{ <diagram description:A2> | }*
```

Información adicional:

Ésta es una regla que define la estructura de un fichero CIF de nivel 2 válido. Obsérvese que esta regla no corresponde a ningún símbolo en un diagrama.

7.4.2 A2 Descripción de diagrama:

```
<diagram start:A3> { <CIF descriptor:A19> | <page switch:A21> }* <diagram end:A18>
```

Información adicional:

Esta regla, junto con las diversas reglas para el arranque de diagrama, <page switch:A21> y la regla <diagram end:A.18> especifica las tramas y las páginas de los diagramas.

7.4.3 A3 Arranque de diagrama:

```
<specification area start: A5B> | <package diagram start: A5> | <system diagram start: A6> |  
<system type diagram start: A7> | <block diagram start: A8> | <block type diagram start: A9> |  
<substructure diagram start: A9> | <process diagram start: A10> | <process type diagram start:  
A11> | <state diagram start: A12> | <state type diagram start: A13> | <state aggregation diagram  
start: A14> | <state aggregation type diagram start: A15> | <procedure diagram start: A16> |  
<operator diagram start: A17>
```

7.4.4 A4 Arranque de área de especificación:

```
/* CIF SpecificationArea */  
{ <page declaration: B2> }+
```

Información adicional:

Debe haber una <page declaration: B2> para cada página en el diagrama.

7.4.5 A5 Arranque de diagrama de paquete:

```
/* CIF PackageDiagram */  
<diagram parts:B1>  
{<package use clause>}* <package heading> <end>
```

Ejemplo:

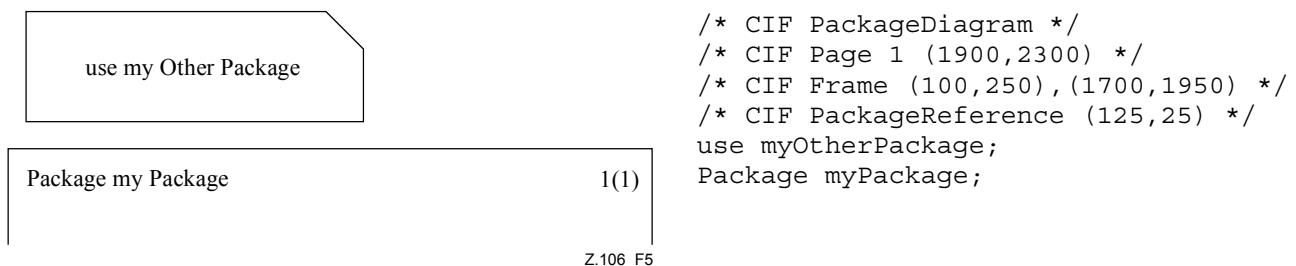


Figure 5/Z.106

7.4.6 A6 Arranque de diagrama de sistema:

```
/* CIF SystemDiagram */  
{ <diagram parts: B1> }+  
{ <package use clause> }*  
<system heading> <end>
```

7.4.7 A7 Arranque de diagrama de tipo de sistema:

```
/* CIF SystemTypeDiagram */  
<diagram parts: B1>  
<package use clause>* <system type heading> <end>
```

7.4.8 A8 Arranque de diagrama de bloque:

```
/* CIF BlockDiagram */<diagram parts: B1>  
{<package use clause>}* <block heading>
```


7.4.9 A9 Arranque de diagrama de tipo de bloque:

```
/* CIF BlockTypeDiagram */  
<diagram parts: B1>  
<package use clause>* <block type heading> <end>
```

7.4.10 A10 Arranque de diagrama de proceso:

```
/* CIF ProcessDiagram */  
<diagram parts: B1>  
{<package use clause>}* <process heading> <end>
```

7.4.11 A11 Arranque de diagrama de tipo de proceso:

```
/* CIF ProcessTypeDiagram */  
<diagram parts: B2>  
<package use clause>* <process type heading> <end>
```

7.4.12 A12 Arranque de diagrama de estado:

```
/* CIF StateDiagram */  
<diagram parts: B1>  
{<package use clause>}* <composite state heading> <end>
```

7.4.13 A13 Arranque de diagrama de tipo de estado:

```
/* CIF StateTypeDiagram */:  
<diagram parts: B1>  
{<package use clause>}* <composite state type heading> <end>
```

7.4.14 A14 Arranque de diagrama de agregación de estados:

```
/* CIF StateAggregationDiagram */  
<diagram parts: B1>  
{<package use clause>}* <state aggregation heading> <end> substructure
```

7.4.15 A15 Arranque de diagrama de tipo de agregación de estados:

```
/* CIF StateAggregationTypeDiagram */  
<diagram parts: B1>  
{<package use clause>}* <state aggregation type heading> <end> substructure
```

7.4.16 A16 Arranque de diagrama de procedimiento:

```
/* CIF ProcedureDiagram */  
<diagram parts: B1>  
{<package use clause>}* <procedure heading> <end>
```

7.4.17 A17 Arranque de diagrama de operador:

```
/* CIF OperatorDiagram */  
<diagram parts: B1>  
{<package use clause>}* <operation heading> <end>
```

7.4.18 A18 Fin de diagrama:

```
{ /* CIF End SpecificationArea */ |  
/* CIF End PackageDiagram */ ENDPACKAGE [ <package name> ] <end> |  
/* CIF End SystemDiagram */ ENDSYSTEM [ <system name> ] <end> |  
/* CIF End SystemTypeDiagram */ ENDSYSTEM TYPE [[<qualifier>] <system type name> ]  
<end> |  
/* CIF End BlockDiagram */ ENDBLOCK [[<qualifier>] <block name> ] <end> |
```

```

/* CIF End BlockTypeDiagram */ ENDBLOCK TYPE [[<qualifier>] <block type name> ]
<end> |
/* CIF End ProcessDiagram ENDPROCESS [[<qualifier>] <process name>] <end> |
/* CIF End ProcessTypeDiagram */ ENDPROCESS TYPE [[<qualifier>] <process type name>
] <end> |
/* CIF End StateDiagram */ ENDSUBSTRUCTURE [[<qualifier>] <composite state name>]
<end> |
/* CIF End StateTypeDiagram */ ENDSUBSTRUCTURE STATE TYPE [[<qualifier>]
<composite state type name>] <end> |
/* CIF End StateAggregationDiagram */ ENDSUBSTRUCTURE [[<qualifier>] <composite
state name>] <end> |
/* CIF End StateAggregationTypeDiagram */ ENDSUBSTRUCTURE STATE TYPE
[[<qualifier>] <composite state type name>] <end> |
/* CIF End ProcedureDiagram */ ENDPROCEDURE [[<qualifier>] <procedure name> ] <end>
|
/* CIF End OperatorDiagram */ ENDOPERATOR [[<qualifier>] <operator name> ] <end>

```

Ejemplo:

```

/* CIF End SystemTypeDiagram */
ENDSYSTEM TYPE mySystemType;

```

7.4.19 A19 Descriptor CIF:

<diagram description: A2> | <default size: A20> | <channel: A22> | <gate: A23> | <gate symbol reference: A24> | <connect: A25> | <state connection point: A26> | <state connection: A27> | <text extension: A28> | <comment: A29> | <create line: A30> | <flow line: A31> | <answer flow line: A32> | <block symbol: A33> | <dashed block symbol: A34> | <process symbol: A35> | <dashed process symbol: A36> | <agent reference in specification area: A37> | <package symbol: A38> | <package reference in specification area: A39> | <operator symbol: A40> | <start symbol: A41> | <stop symbol: A42> | <state symbol: A43> | <nextstate symbol: A44> | <exception handler symbol: A45> | <on exception symbol: A46> | <handle symbol: A47> | <save symbol: A48> | <task symbol: A49> | <set symbol: A50> | <reset symbol: A51> | <export symbol: A52> | <create symbol: A53> | <procedure call symbol: A54> | <procedure start symbol: A55> | <return symbol: A56> | <raise symbol: A57> | <decision symbol: A58> | <continuous signal symbol: A59> | <enabling condition symbol: A60> | <transition option symbol: A61> | <join symbol: A62> | <connect symbol: A63> | <label symbol: A64> | <input symbol: A65> | <priority input symbol: A66> | <output symbol: A67> | <text symbol: A68> | <select symbol: A69> | <descriptor end: A70> | <type reference: A71> | <association: A72> | <specialization line: A73> | <dependency line: A74>

Información adicional:

<diagram description: A2> se utiliza para describir diagramas anidados.

7.4.20 A20 Tamaño por defecto:

```

/* CIF DefaultSize <size point: B22> */

```

Información adicional:

Este comentario puede colocarse antes de cualquier símbolo o construcción SDL-PR de línea dentro de un diagrama.

El tamaño dado aquí se utilizará para todos los símbolos subsiguientes sin un tamaño definido hasta que se da un nuevo tamaño por defecto. El tamaño por defecto es recordado incluso después de un <diagram start: A3> nuevo. Es ilegal omitir una especificación de tamaño de símbolo antes de darse el tamaño por defecto.

Ejemplo: donde el símbolo de tarea tendrá el tamaño (200,100):

```
/* CIF DefaultSize (200,100) */  
/* CIF Task (800,550) */  
task GameP:=null;
```

7.4.21 A21 Cambio de página:

```
/* CIF CurrentPage <page name> */
```

Información adicional:

Este comentario puede colocarse antes de cualquier símbolo o construcción SDL-CIF y SDL-PR de línea dentro de un diagrama. El nombre de página debe referirse a una página declarada en el comentario CIF de arranque de diagrama anterior.

Cualquier cosa posterior a este comentario CIF en este diagrama se colocará en la página actual, hasta que se defina otra página actual. La página actual para un diagrama es establecida inicialmente por una <page declaration: B2>.

Ejemplo:

```
/* CIF CurrentPage 1 */  
/* CIF Task (800,550) */  
task GameP:=null;
```

7.4.22 A22 Canal:

```
/* CIF Channel <pointlist: B18> [ InvisibleName ] */  
[ <channel name text position: B21> ]  
[ <first signallist text position: B4> ]  
[ <second signallist text position: B5> ]  
[ <first arrow position: B6> ]  
[ <second arrow position: B7> ]  
<channel definition>
```

Información adicional:

Si **InvisibleName** es dado, el nombre de canal debe aparecer en SDL-GR. El nombre solamente es dado en SDL-PR para permitir referirse al mismo en una declaración **CONNECT**.

El primer punto en la lista de puntos está en el rectángulo circundante del símbolo correspondiente al <channel endpoint> **FROM** del primer <channel path> o en una puerta conectada a este símbolo. El último punto de la lista de punto se encuentra en el rectángulo circundante del símbolo correspondiente al <channel endpoint> **TO** del primer <channel path> o en una puerta conectada a este símbolo.

La posición del texto de la puerta en el construcción **VIA** viene especificada en <gate: A23> o en <gate reference: B15>. La siguiente figura muestra una puerta y una referencia a la puerta.

Ejemplo 1:

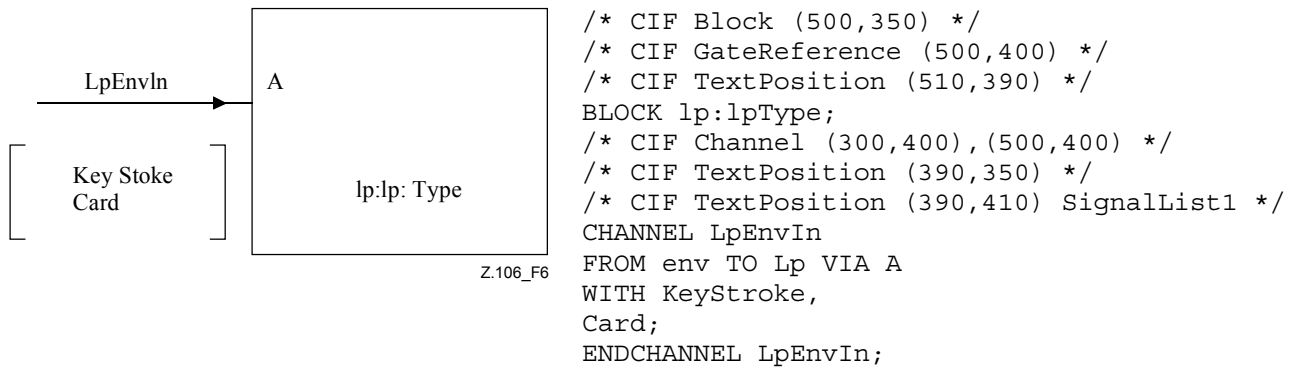


Figura 6/Z.106

7.4.23 A23 Puerta:

```

/* CIF Gate <pointlist: B18> [ Dashed ] */
[ <gate name text position: B21> ]
[ <first signallist text position: B4> ]
[ <second signallist text position: B5> ]
[ <gate constraint symbol: B3> ]
<gate definition>

```

Información adicional:

Debe haber dos puntos en la <pointlist: B18>. El primer punto en la lista de puntos está en el rectángulo circundante del símbolo de recuadro del diagrama. El segundo punto en la lista de puntos debe ser el otro punto que define la puerta. Si existe una <textual endpoint constraint> existe como parte de <gate definition>, el segundo punto estará en el rectángulo del símbolo que corresponde a la <textual endpoint constraint>.

Debe utilizarse **Dashed** si se emplea la palabra clave **ADDING** en <gate definition>.

Debe darse el <gate constraint symbol: B3> si se da una <textual endpoint constraint>.

Ejemplo 1:

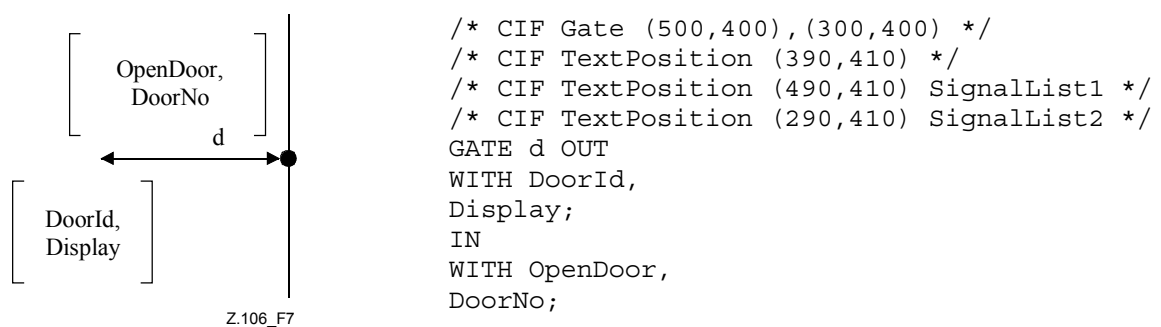


Figura 7/Z.106

Ejemplo 2:

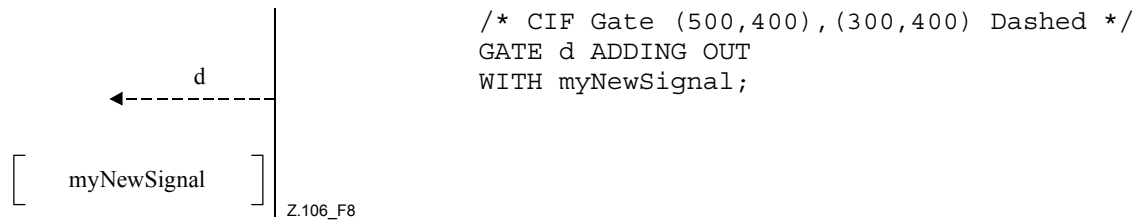


Figure 8/Z.106

Example 3:

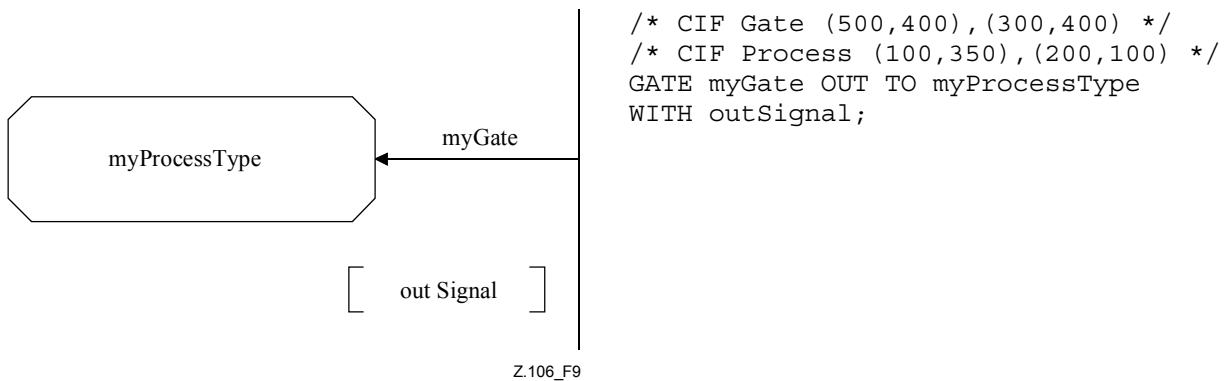


Figura 9/Z.106

7.4.24 A24 Referencia de símbolo de puerta:

```

/* CIF GateSymbolReference <name> [In [ <charstring> ] ] [Out [ <charstring> ] ] <pointlist:
B18> */
<text position: B21>
[ <in signal list position: B39> ]
[ <out signal list position: B40> ]

```

Información adicional:

El comentario CIF se utiliza para especificar puertas a referencias de agente y a referencias de tipo de agente. Obsérvese que estas puertas no tienen PR correspondiente.

La <text position: B21> define la posición del <name>.

Si se especifica **In**, hay entonces una flecha en la puerta con el sentido hacia el símbolo de referencia. La <in signal list position> define la posición de la <charstring> (que describe la lista de señales) asociada con **In**.

Si se especifica **Out**, hay entonces una flecha en la puerta con el sentido desde el símbolo de referencia. La <out signal list position> define la posición de la <charstring> (que describe la lista de señales) asociada con **Out**.

Ejemplo:

```

/* CIF GateSymbolReference g In 's1, s2' Out 's3, s4' ((800,550), (900,
550)) */
/* CIF TextPosition (850,600) */
/* CIF TextPosition (750,500) In */
/* CIF TextPosition (850,600) Out */

```

7.4.25 A25 Conectar:

```
/* CIF Connect [ <position: B22> ]*/  
[ <text position: B21> ]  
<channel to channel connection>
```

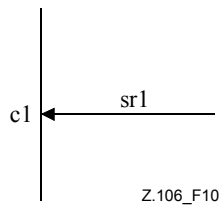
La producción para <channel to channel connection> se define en 5.6.2.

Información adicional:

<position: B22> es la posición del canal en el símbolo de recuadro.

<text position: B21> es la posición de texto de los <external channel identifiers>, es decir, el texto exterior al símbolo de recuadro.

Ejemplo:



```
/* CIF Connect */  
/* CIF TextPosition (800,50) */  
CONNECT c1 AND sr1;
```

Figure 10/Z.106

7.4.26 A26 Punto de conexión de estado:

```
/* CIF StateConnectionPoint <pointlist: B18> */  
[ <text position: B21> ]  
<state connection points>
```

Información adicional:

Debería haber dos puntos en la <pointlist: B18>. El primer punto de la lista de puntos debe estar en el rectángulo circundante del símbolo del recuadro del diagrama. El segundo punto de la lista de puntos define el otro extremo del símbolo del punto de conexión de estado. Si existe una <text position>, especifica la posición del texto en el <state connection point>.

7.4.27 A27 Conexión de estado:

```
/* CIF StateConnection <pointlist: B18> */  
[ <text position: B21> ]  
<state partition connection>
```

La producción para <state partition connection> se define en 5.7.10.2:

Información adicional:

El primer punto de la lista de puntos está en el rectángulo circundante del símbolo correspondiente al <inner entry point>. El último punto de la lista de puntos está en el rectángulo circundante del símbolo de recuadro de la página actual.

La <text position> especifica la posición del texto en <inner entry point>, es decir, el nombre o nombres del punto o puntos de entrada asociados con el símbolo de estado conectado.

7.4.28 A28 Extensión de texto:

```
/* CIF TextExtension <position and size: B20>
[ { Left | Right } ] */
[ <text position: B21> ]
[ <line: B16> ]
<text>
/* CIF End TextExtension */
```

Información adicional:

Left significa que el lado izquierdo del símbolo está abierto. **Right** significa que el lado derecho del símbolo está abierto. **Right** es el valor por defecto.

Si **Left** está dado, la posición del texto y el símbolo define la esquina superior *derecha*.

La <line: B16> es la línea que conecta el símbolo de extensión de texto con el otro símbolo. Si la línea no está dada, será autopresentada. El primer punto en la lista de puntos está en el rectángulo circundante del símbolo de extensión de texto. El último punto de la lista de puntos está en el rectángulo circundante del símbolo al cual está unido el símbolo de extensión de texto.

Un carácter de cambio de renglón antes o después de uno o de los dos comentarios de extensión de texto CIF no debe ser considerado parte del texto en los símbolos.

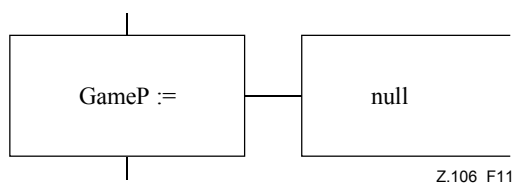
<Text extension: A28> y <Comment: A29> deben colocarse antes del <end> del siguiente modo: Las extensiones y comentarios de texto pueden unirse a cualquier regla en la gama <block symbol: A33> – <select symbol: A69>.

Cuando se une una extensión de texto al símbolo de tarea que contiene una <compound statement>, la extensión de textos se colocará antes del <right curly bracket> en la declaración de compuesto.

Ejemplo 1 (ejemplo con un símbolo de tarea informal):

```
/* CIF Task (800,550) */
TASK 'first part of task text that will be in the task symbol'
/* CIF TextExtension (1100,550) */
'last part of task text that will be in the TextExtension symbol'
/* CIF End TextExtension */
;
```

Ejemplo 2:



```
/* CIF Task (800,550) */
TASK GameP: =
/* CIF TextExtension (1100,550) */
/* CIF Line (1100,600), (1000,600) */
null
/* CIF End TextExtension */
;
```

Figura 11/Z.106

7.4.29 A29 Comentario:

```
/* CIF Comment <position and size: B20> [ Left | Right ] [ Dashed ] */
[ <text position: B21> ]
[ <dashed line: B17> ]
<comment> <end>
```

Información adicional:

Left significa que el lado izquierdo del símbolo está abierto. **Right** significa que el lado derecho del símbolo está abierto. **Right** es el valor por defecto.

Si **Left** está dado, la posición de (texto y) símbolo define la esquina superior *derecha*.

Si **Dashed** está dado, el símbolo de comentario debe dibujarse en trazos discontinuos (como un <comment symbol2> en SDL96). Si **Dashed** no está dado, el símbolo de comentario debe dibujarse en trazo no discontinuo (como un <comment symbol> en SDL92).

La <dashed line: B17> es la línea que conecta el símbolo de comentario con el otro símbolo. Si la línea no está dada, será autopresentada. El primer punto de la lista de puntos está en el rectángulo circundante del símbolo de comentarios. El último punto de la lista de puntos está en el rectángulo circundante del símbolo al que está unido el comentario.

En <text extension: A28> se explica como se utiliza esta construcción CIF.

Ejemplo 1:

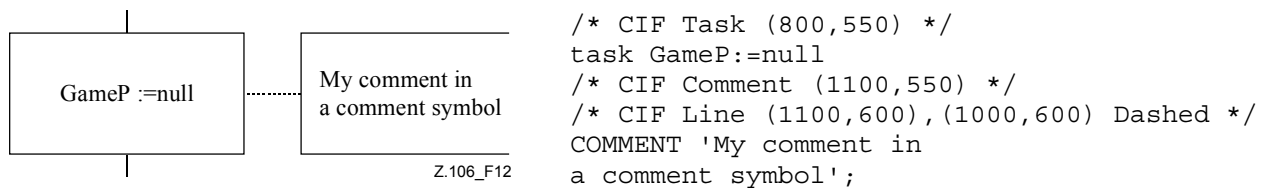


Figura 12/Z.106

Ejemplo conexo:

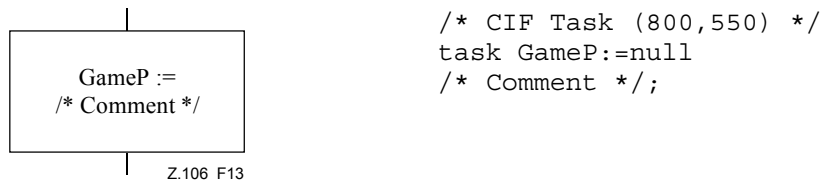


Figura 13/Z.106

Ejemplo 2:

```

/* CIF Task (800,550) */
task GameP: =
/* CIF TextExtension (1100,550) */
/* CIF Line (1100,600),(1000,600) */
null
/* CIF End TextExtension */
/* CIF Comment (1100,750) */
/* CIF Line (1100,800),(1000,600) Dashed */
COMMENT 'My comment in a comment symbol';

```

7.4.30 A30 Crear línea:

```
/* CIF CreateLine <pointlist: B18> */
```

Información adicional:

El primer punto en la lista de puntos está en el rectángulo circundante del símbolo de proceso que crea el otro proceso. El último punto de la lista de puntos está en el rectángulo circundante del símbolo de proceso que se crea.

Ejemplo:

```

/* CIF DefaultSize (200,100) */
/* CIF Process (200,500) */
PROCESS Main(1,1) REFERENCED;
/* CIF Process (500,500) */
PROCESS Game(0,1) REFERENCED;

```



```
/* CIF CreateLine (400,550), (500,550) */
```

7.4.31 A31 Línea de flujo:

```
<line: B16>
```

Información adicional:

Este comentario CIF es opcional. Si el comentario CIF no está dado para una línea de flujo, la línea de flujo se autopresenta.

El primer punto de la lista de puntos se encuentra en el rectángulo circundante del símbolo para el que el flujo es entrante. El último punto de la lista de puntos se encuentra en el rectángulo circundante del símbolo para el que el flujo es saliente.

El comentario se puede colocar antes o después de cualquier construcción SDL-CIF y SDL-PR de línea o símbolo dentro del <process body> del diagrama.

Una línea de flujo que une otra línea de flujo debe describir la lista de puntos completa desde un punto en el rectángulo circundante del símbolo "from" a un punto en el rectángulo circundante del símbolo "to".

Las flechas en las líneas de flujo están implícitas. Las líneas de flujo después de símbolos de opción de transición y decisión deben utilizar <answer flow line: A32>, que es una regla no opcional.

Ejemplo:

```
/* CIF Start (300,100) */  
START;  
/* CIF Line (400,200), (400,250) */  
/* CIF Set (300,250) */  
Set (Now+1, T);
```

7.4.32 A32 Línea de flujo de respuesta:

```
/* CIF Answer [ { Right | Left } ] [ InvisibleBrackets ] */  
[ <line: B16> ] [ <text position: B21> ]  
{ <textual answer part> | <textual else part> }
```

Esta regla es utilizada por <CIF descriptor: A19>.

Las producciones para <textual answer part> y <textual else part> se definen en 5.7.12.2.

Información adicional:

Este comentario CIF debe utilizarse para líneas de flujo después de un símbolo de opción de transición o una decisión.

Si se da la <line: B16>:

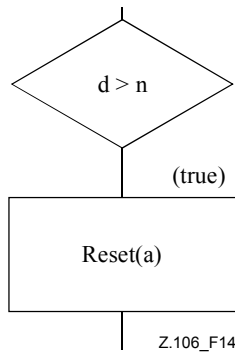
Right y **Left** no tienen significado. La lista de puntos en la línea de flujo especifica donde arranca la línea de flujo en el símbolo de decisión. Se aplican las mismas reglas que para <flow line: A31>.

Si la <line: B16> no está dada:

Right significa que la línea de flujo arranca a la derecha del símbolo de decisión (o en la esquina inferior derecha del símbolo de opción de transición). **Left** significa que la línea de flujo arranca a la izquierda del símbolo de decisión (o en la esquina inferior izquierda del símbolo de opción de transición). Por defecto, la línea de flujo arranca bajo el símbolo de decisión (o en el centro del borde inferior del símbolo de opción de transición). El resto de la línea de flujo es autopresentada.

Si se da **InvisibleBrackets**, los caracteres (y) que pueden encontrarse en SDL-PR no son visibles en SDL-GR.

Ejemplos:

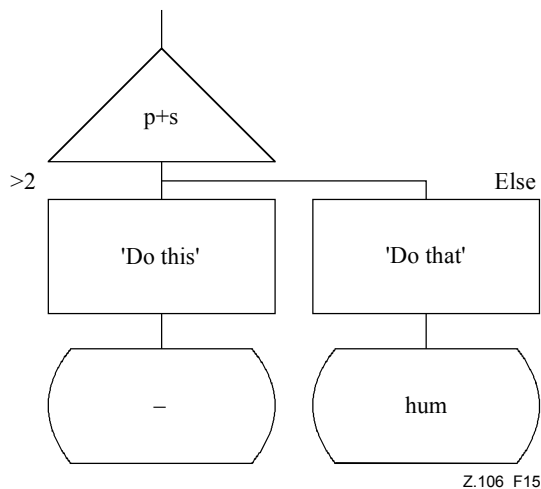


```

/* CIF Decision (800,550) */
DECISION d > n;
/* CIF Answer */
/* CIF Line (900,650), (900,750) */
/* CIF TextPosition (910,690) */
(true):
/* CIF Reset (800,750) */
reset(a);
-----

```

Figura 14/Z.106



```

/* CIF Alternative (800,550) */
ALTERNATIVE p + s;
/* CIF Answer InvisibleBrackets */
(>2):
/* CIF Task (800,750) */
TASK 'Do this';
/* CIF NextState (800,950) */
NEXTSTATE -;
/* CIF Answer */
ELSE:
/* CIF Task (1100,750) */
TASK 'Do that';
/* CIF NextState (1100,950) */
NEXTSTATE hum

```

Figura 15/Z.106

7.4.33 A33 Símbolo de bloque:

```

<block symbol rectangle: B12>
[ <text position: B21> ]
{ <gate reference: B15>* <block reference> |
  { <gate reference: B15>* <textual typebased block definition> } }

```

Esta regla es utilizada por <CIF descriptor: A19>.

Información adicional:

Las <gate reference: B15> son opcionales y se utilizan solamente para especificar posiciones de texto para referencias de puerta unidas a este símbolo de bloque. Si se omite una <gate reference: B15>, la posición de texto para la referencia de puerta se autopresentará. El nombre de la referencia de puerta está en SDL-PR mencionado junto con SDL-PR para canales o rutas de señales conectados.

Ejemplo 1:

```

/* CIF Block (800,550) */
/* CIF TextPosition (810,560) */
BLOCK myBlock REFERENCED;

```

Ejemplo 2:

```

/* CIF Block (800,550) */
/* CIF GateReference (900,550) */

```

```
/* CIF TextPosition (890,500) */  
BLOCK myBlocks (2) :myBlockType;
```

7.4.34 A34 Símbolo de bloque de trazo discontinuo:

```
/* CIF Block <position and size: B20> Dashed <block name> */  
[ <text position: B21> ]  
<gate reference: B15>*
```

Información adicional:

Este comentario puede colocarse en cualquier lugar en que una <textual block reference> está permitida. Debe haber un <dashed block symbol: A34> para cada <existing typebased block definition> en la SDL-GR. <gate reference: B15> se explica en <block symbol: A33>.

Ejemplo:

```
/* CIF Block (800,550) Dashed myBlock */
```

7.4.35 A35 Símbolo de proceso:

```
<process symbol rectangle: B13>  
[ <text position: B21> ]  
{ <gate reference: B15>* <process reference> |  
  { <gate reference: B15>* <textual typebased process definition> } }
```

Información adicional:

<gate reference: B15> se explica en <block symbol: A33>.

Ejemplo 1:

```
/* CIF Process (800,550) */  
PROCESS myProcess REFERENCED;
```

Ejemplo 2:

```
/* CIF Process (800,550) */  
PROCESS myProcess (1,1) :myProcessType;
```

7.4.36 A36 Símbolo de proceso de trazo discontinuo:

```
/* CIF Process <position and size: B20> Dashed <process name> */  
[ <text position: B21> ]  
<gate reference: B15>*
```

Información adicional:

Este comentario puede colocarse en cualquier lugar en que <textual process reference> esté permitida. Debe haber un <dashed process symbol: A36> para cada <existing typebased process definition> en SDL-GR. <gate reference: B15> se explica en <block symbol: A33>.

Ejemplo:

```
/* CIF Process (800,550) Dashed myProcess */
```

7.4.37 A37 Referencia de agente en área de especificación:

```
/* CIF Agent <name> [ : <type expression> ] <position and size: B20> { System | Block | Process  
} */  
[ <text position: B21> ]
```

Información adicional:

El símbolo sólo se puede utilizar en áreas de especificación.

Si se especifica **System**, el símbolo real será una <system reference area>.

Si se especifica **Block**, el símbolo real será una <block reference area>.

Si se especifica **Process**, el símbolo real será una <process reference area>.

La <text position> se refiere al texto correspondiente a '<name> [: <type expression>]'.

Ejemplo 1:

```
/* CIF Agent P (800,550) Process */
/* CIF TextPosition (810,560) */
```

7.4.38 A38 Símbolo de lote:

```
/* CIF Package <position and size: B20> */
[ <text position: B21> ]
<package reference>
```

Información adicional:

Este comentario CIF sólo se utilizará para referencias de lote en diagramas de lote, en los que existe una sintaxis SDL-PR para referencias de lote. Cuando se describan referencias de paquete en áreas de especificación, se utilizará la regla 'package ref in specification area'.

Ejemplo:

```
/* CIF Package (800,550) */
PACKAGE myPKG REFERENCED;
```

7.4.39 A39 Referencia de lote en área de especificación:

```
/* CIF PackageReference <name> <position and size: B20> */
[ <text position: B21> ]
```

7.4.40 A40 Símbolo de operador:

```
/* CIF Operator <identifier> <position and size: B20> */
[ <text position: B21> ]
```

Información adicional:

Este comentario puede colocarse en cualquier lugar en el que se pueda colocar <entity in package>, <entity in system>, <entity in block>, <entity in process> o <entity in procedure>. El símbolo de operador no tiene conexión gráfica directa con <textual operator reference> (que se presenta como texto en un símbolo de texto).

Ejemplo:

```
/* CIF Operator myOperator (800,550) */
```

7.4.41 A41 Símbolo de arranque:

```
/* CIF Start <position and size: B20> */
[ <text position: B21> ]
start [ <virtuality> ] [ <state entry point name> ] <end>
```

Información adicional:

Este comentario CIF se debe utilizar en diagramas de agente y de tipo de agente. Los diagramas de procedimiento y de operador deben utilizar <procedure start symbol: A55>.

Ejemplo:

```
/* CIF Start (800,550) */
START;
```

7.4.42 A42 Símbolo de parada:

```
/* CIF Stop <position and size: B20> */
<stop> <end>
```

Ejemplo:

```
/* CIF Stop (800,550) */
STOP;
```

7.4.43 A43 Símbolo de estado:

```
/* CIF State <position and size: B20> */
[ <text position: B21> ]
state <state list> <end> | state <composite state list> <end>
```

Ejemplos de este comentario CIF pueden encontrarse en el comentario CIF relacionado <nextstate symbol: A44>.

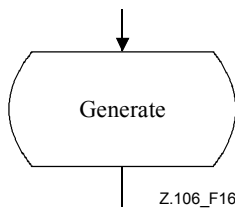
7.4.44 A44 Símbolo de estado siguiente:

```
/* CIF NextState <position and size: B20> */
[ <text position: B21> ]
<nextstate>
```

Información adicional:

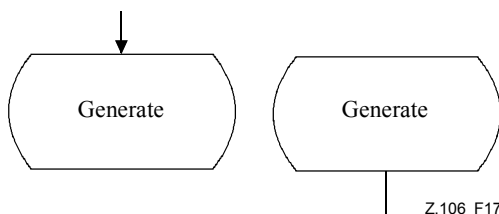
Se debe dar un comentario CIF para cada estado y cada estado siguiente en SDL-PR. Ello significa que habrá dos comentarios CIF para una símbolo SDL-GR que corresponde a un estado SDL-PR y a un estado siguiente SDL-PR. Una herramienta que lee un fichero CIF debe determinar si un estado y un estado siguiente son efectivamente un símbolo SDL-GR comparando las coordenadas de los símbolos en los dos comentarios CIF.

Ejemplos:



```
/* CIF NextState (800,550) */
NEXTSTATE Generate;
/* CIF State (800,550) */
STATE Generate;
```

Figura 16/Z.106



```
/* CIF NextState (800,550) */
NEXTSTATE Generate;
/* CIF State (1100,550) */
STATE Generate;
```

Figura 17/Z.106

7.4.45 A45 Símbolo de manejador de excepciones:

```
/* CIF Exception Handler { <position and size: B20> } */  
[ <text position: B21> ]  
exceptionhandler <exception handler list> <end>
```

Ejemplo:

```
/* CIF Exception Handler (800,550) */  
exceptionhandler myXX;
```

7.4.46 A46 Símbolo de excepción activa:

```
/* CIF OnException <position and size: B20> */  
[ <text position: B21> ]  
[ <pointlist> ]  
onexception <exception handler name> <end>
```

Información adicional:

El <pointlist> define el <solid on exception association symbol> que conecta el <exception handler symbol> con el otro símbolo. El primer punto en la lista de puntos está en el rectángulo circundante del <exception handler symbol>. El último punto de la lista de puntos está en el rectángulo circundante del símbolo al que se une la excepción.

Obsérvese que puede utilizarse el mismo <exception handler symbol> gráfico como definición de manejador de excepción y como una excepción activa. Así ocurre si <position and size> son iguales en los comentarios CIF manejador de excepción y excepción activa.

Si se omite <pointlist>, se utilizará entonces la autopresentación para el <solid on exception association symbol>.

Ejemplo:

```
/* CIF OnException (800,550) */  
onexception myXX;
```

7.4.47 A47 Símbolo de manejo:

```
/* CIF Handle <position and size: B20> [ { Left | Right } ] */  
[ <text position: B21> ]  
handle [ <virtuality> ] <exception stimulus list> <end>
```

Información adicional:

Left significa que la parte del símbolo que visualiza una flecha se encuentra a la izquierda. **Right** significa que la parte del símbolo que visualiza una flecha se encuentra a la izquierda. El valor por defecto es **Right**.

Ejemplo:

```
/* CIF Handle (800,550) */  
HANDLE myException;
```

7.4.48 A48 Símbolo de conservación:

```
/* CIF Save <position and size: B20> */  
[ <text position: B21> ]  
<save part>
```

Ejemplo:

```
/* CIF Save (800,550) */  
SAVE mySignal;
```

7.4.49 A49 Símbolo de tarea:

```
/* CIF Task <position and size: B20> */  
[ <text position: B21> ]  
<task> <end>
```

Información adicional:

Hay tres casos en los que un símbolo de tarea debe ser descrito por comentarios CIF distintos que éste. Los símbolos de tarea SDL-GR que contienen <set> deben utilizar <set symbol: A50>. Los símbolos de tareas SDL-GR que contienen <reset> deben utilizar <reset symbol: A51>. Los símbolos de tareas SDL-GR que contienen <export> deben utilizar <export symbol: A52>.

Ejemplo:

```
/* CIF Task (800,550) */  
TASK myVariable: = 0;
```

7.4.50 A50 Símbolo de fijar:

```
/* CIF Set <position and size: B20> */  
[ <text position: B21> ]  
<set> <end>
```

Información adicional:

Un símbolo fijar es un símbolo de tarea SDL-GR que contiene un solo <set statement> (véase [1], 11.14).

Ejemplo:

```
/* CIF Set (800,550) */  
SET (Now+1, myTime);
```

7.4.51 A51 Símbolo de reiniciar:

```
/* CIF Reset <position and size: B20> */  
[ <text position: B21> ]  
<reset> <end>
```

Información adicional:

Un símbolo de reiniciar es un símbolo de tarea SDL-GR que contiene una sola <reset statement>. (véase [1], 11.14).

Ejemplo:

```
/* CIF Reset (800,550) */  
RESET T;
```

7.4.52 A52 Símbolo de exportación:

```
/* CIF Export <position and size: B20> */  
[ <text position: B21> ]  
<export> <end>
```

Información adicional:

Un símbolo de exportación es un símbolo de tarea SDL-GR que contiene una sola <export statement> (véase [1], 11.14).

Ejemplo:

```
/* CIF Export (800,550) */  
Export (myVariable1, myVariable2);
```

7.4.53 A53 Símbolo de crear:

```
/* CIF Create <position and size: B20> */  
[ <text position: B21> ]  
<create request> <end>
```

Ejemplo:

```
/* CIF Create (800,550) */  
CREATE Game;
```

7.4.54 A54 Símbolo de llamada a procedimiento:

```
/* CIF ProcedureCall <position and size: B20> */  
[ <text position: B21> ]  
<procedure call> <end>
```

Ejemplo:

```
/* CIF ProcedureCall (800,550) */  
CALL myProcedure;
```

7.4.55 A55 Símbolo de arranque de procedimiento:

```
/* CIF ProcedureStart <position and size: B20> */  
[ <text position: B21> ]  
start [ <virtuality> ] [ <state entry point name> ] <end>
```

Información adicional:

Este comentario CIF se debe utilizar para diagramas de procedimiento y de operador. Los diagramas de agente, de tipo de agente, de estado y de tipo de estado deben utilizar <start symbol: A41>.

Ejemplo:

```
/* CIF ProcedureStart (800,550) */  
START;
```

7.4.56 A56 Símbolo de retornar:

```
/* CIF Return <position and size: B20> */  
[ <text position: B21> ]  
<return> <end>
```

Ejemplo:

```
/* CIF Return (800,550) */  
RETURN myReturnValue;
```

7.4.57 A57 Símbolo de realzar:

```
/* CIF Raise <position and size: B20> */  
[ <text position: B21> ]  
<raise> <end>
```

Ejemplo:

```
/* CIF Raise (800,550) */  
RAISE myException;
```

7.4.58 A58 Símbolo de decisión:

```
/* CIF Decision <position and size: B20> */  
[ <text position: B21> ]  
DECISION <question> <end>
```


Información adicional:

Una línea de flujo inmediatamente después de un símbolo de decisión debe describirse mediante una <answer flow line: A32>.

Ejemplo (sin líneas de flujo):

```
/* CIF Decision (800,550) */  
DECISION DoorIndex > NoOfDoors;
```

7.4.59 A59 Símbolo de señal continua:

```
/* CIF ContinuousSignal <position and size: B20> */  
[ <text position: B21> ]  
provided [ <virtuality> ] <continuous expression> <end> [ priority <priority name> <end> ]
```

Ejemplo:

```
/* CIF ContinuousSignal (800,550) */  
PROVIDED level > 5;
```

7.4.60 A60 Símbolo de condición habilitadora:

```
/* CIF EnablingCondition <position and size: B20> */  
[ <text position: B21> ]  
<enabling condition>
```

Ejemplo:

```
/* CIF EnablingCondition (800,550) */  
PROVIDED level > 5;
```

7.4.61 A61 Símbolo de opción de transición:

```
/* CIF TransitionOption <position and size: B20> */  
[ <text position: B21> ]  
ALTERNATIVE <alternative question> <end>
```

Información adicional:

Una línea de flujo inmediatamente después de un símbolo de opción de transición debe describirse mediante una <answer flow line: A32>.

Ejemplo:

```
/* CIF TransitionOption (800,550) */  
ALTERNATIVE level;
```

7.4.62 A62 Símbolo de unión:

```
/* CIF Join { <position and size: B20> | Invisible } */  
[ <text position: B21> ]  
<join> <end>
```

Información adicional:

Invisible significa que esta unión SDL-PR no debe ser visible como símbolo en SDL-GR, se da sólo en SDL-PR para indicar una línea de flujo que termina en un símbolo ya descrito (es decir, el símbolo que sigue a la etiqueta <join> se refiere a ella). Véase también <label symbol: A64>.

Ejemplo 1:

```
/* CIF Join Invisible */  
JOIN myInvisibleLabel;
```

Ejemplo 2:

```
/* CIF Join (800,550) */  
JOIN myLabel;
```

7.4.63 A63 Conectar:

```
/* CIF Connect */
[ <line: B16> ] [ <text position: B21> ]
connect [ <virtuality> ] [ <connect list> ] <end>
```

Información adicional:

Este comentario CIF debe utilizarse para líneas de flujo desde un símbolo de estado a una transición sin una entrada de arranque ni una señal continua.

7.4.64 A64 Símbolo de etiqueta:

```
/* CIF Label { <position and size: B20> | Invisible } */
[ <text position: B21> ]
{ <label> | CONNECTION <label> }
```

Información adicional:

Invisible significa que esta etiqueta SDL-PR no debe ser visible como símbolo en SDL-GR, dándose solamente en SDL-PR para indicar una línea de flujo que termina en un símbolo ya descrito (es decir, el símbolo que sigue a la etiqueta). Véase también <join symbol: A62>.

La primera <label> en una <free action> tiene su comentario CIF situado antes de CONNECTION; véase el tercer ejemplo a continuación.

Ejemplo 1:

```
/* CIF Label Invisible */
myInvisibleLabel:
```

Ejemplo 2:

```
/* CIF Label (800,550), (100,100) */
myVisibleLabel:
```

Ejemplo 3:

```
/* CIF Label (800,550), (100,100) */
CONNECTION myLabel:
```

7.4.65 A65 Símbolo de entrada:

```
/* CIF Input <position and size: B20> [ { Left | Right } ] */
[ <text position: B21> ]
{ input [ <virtuality> ] <input list> <end> | input [ <virtuality> ] <spontaneous designator> <end> }
```

Información adicional:

Left significa que la parte del símbolo que visualiza una flecha se encuentra a la izquierda. **Right** significa que la parte del símbolo que visualiza una flecha se encuentra a la derecha. El valor por defecto es **Right**.

Ejemplo:

```
/* CIF Input (800,550) Left */
INPUT mySignal;
```

7.4.66 A66 Símbolo de entrada prioritaria:

```
/* CIF PriorityInput <position and size: B20>
[ { Left | Right } ] */
[ <text position: B21> ]
priority input [ <virtuality> ] <priority input list> <end>
```

Información adicional:

Left significa que la parte del símbolo que visualiza una flecha se encuentra a la derecha. **Right** significa que la parte del símbolo que visualiza una flecha se encuentra a la derecha. El valor por defecto es **Right**.

Ejemplo:

```
/* CIF PriorityInput (800,550) Left */
PRIORITY INPUT mySignal;
```

7.4.67 A67 Símbolo de salida:

```
/* CIF Output <position and size: B20> [ { Left | Right } ] */
[ <text position: B21> ]
<output> <end>
```

Información adicional:

Left significa que la parte del símbolo que visualiza una flecha se encuentra a la derecha. **Right** significa que la parte del símbolo que visualiza una flecha se encuentra a la derecha. El valor por defecto es **Right**.

Ejemplo:

```
/* CIF Output (800,550) */
OUTPUT mySignal;
```

7.4.68 A68 Símbolo de texto:

```
/* CIF Text <position and size: B20> */
[ <text position: B21> ]
<text>
/* CIF End Text */
```

Información adicional:

<text> no puede contener una línea adaptada **/* CIF End Text */**

Un carácter de cambio de renglón antes o después de los dos comentarios de texto CIF no debe considerarse como parte del texto en el símbolo de texto.

Ejemplo:

```
/* CIF Text (800,550) */
Timer myTimer;
/* CIF End Text */
```

7.4.69 A69 Símbolo de seleccionar:

```
/* CIF Select <pointlist: B18> */
[ <text position: B21> ]
SELECT IF <boolean simple expression> <end>
```

Información adicional:

Los puntos en la lista de puntos describen todas las esquinas del símbolo seleccionado, en el sentido de las agujas del reloj o en sentido contrario a las agujas del reloj.

Ejemplo:

```
/* CIF Select (700,400) , (1100,400) , (1100,750) , (700,750) */
/* CIF TextPosition (725,425) */
SELECT IF (p = 3);
```

7.4.70 A70 Fin de descriptor:

```
{ /* CIF End Decision */ ENDDECISION <end> |
/* CIF End State */ ENDSTATE [ <state name> ] <end> |
/* CIF End Label */ ENDCONNECTION [ <connector name> ] <end> |
/* CIF End Select */ ENDSELECT <end> |
/* CIF End TransitionOption */ ENDALTERNATIVE <end> |
/* CIF End ExceptionHandler */ ENDEXCEPTIONHANDLER [ <exception handler name> ]
<end> }
```

Información adicional:

Esta regla se introduce para distinguir el final de la información sobre un símbolo del final de la información sobre un diagrama.

7.4.71 A71 Referencia de tipo:

```
/* CIF TypeReference <position and size: B20> [ Iconized ] */
[ <id text position: B23> ] [ <stereotype text position: B24> ] [ <icon symbol: B29> ]
[ <class symbol first line position: B27> ] [ <attribute text position: B25> ]
[ <class symbol second line position: B28> ] [ <behaviour text position: B26> ]
{ { system | block | process | object | value | state } type } | signal | procedure | interface }
<identifier> <type reference properties>
```

Información adicional:

Si **Iconized** es parte del comentario CIF, el símbolo de referencia de tipo no debe utilizar el símbolo de clase, sino los símbolos especiales para tipos o procedimientos bloque/proceso/estado. En otro caso, se utilizará el símbolo de clase.

Si aparece el <icon symbol: B29>, el símbolo de clase utilizará entonces el icono pequeño correspondiente a la modalidad del tipo. Si no aparece <icon symbol: B29>, se utilizará el nombre de estereotipo.

Los símbolos de clase se dividen en tres compartimentos que utilizan dos líneas. El compartimento superior del símbolo de clase contiene el <identifier> y sea el estereotipo (uno de **system**, **block**, **process**, **object**, **value**, **state**, **signal**, **procedure**, **interface**) o un icono pequeño que represente el estereotipo. El compartimento medio contiene la parte <attribute property>. El compartimento inferior contiene la parte <behaviour property>.

Todas las posiciones de texto y de líneas son opcionales en esta directriz.

Ejemplo:

```
/*CIF TypeReference (500,400) */
/* CIF Icon */
/* CIF Line1 (500,440) */
/* CIF Line2 (500,480) */
process type P
with integer;
with signal s (Integer);
referenced;
```

7.4.72 A72 Asociación:

```
/* CIF Association <pointlist: B18> */
[<association name text position: B30>]
[<first association end role name position: B31>]
[<first association end visibility position: B32>]
[<first association end multiplicity position: B33>]
[<first association end ordering position: B34>]
[<second association end role name position: B35>]
[<second association end visibility position: B36>]
[<second association end multiplicity position: B37>]
[<second association end ordering position: B38>]
<association definition>
```

Información adicional:

Todas las posiciones (excepto la <pointlist: B18>) son opcionales.

El primero y último puntos de la lista de puntos deben estar en el rectángulo circundante de los símbolos enlazados.

Ejemplo:

```
/* CIF Association (400,300), (500,700) */
/* TextPosition (500,500) Association */
association ExampleAssociation from Pt1 from Pt2;
```

7.4.73 A73 Línea de especialización:

```
/* CIF SpecializationLine <pointlist: B18> */
```

Información adicional:

El primer punto de la lista de puntos está en el rectángulo circundante del supertipo. El último punto de la lista de puntos está en el rectángulo circundante del subtipo.

7.4.74 A74 Línea de dependencia:

```
/* CIF DependencyLine <pointlist: B18> */
```

Información adicional:

El primer punto de la lista de puntos está en el rectángulo circundante del cliente (la entidad que "utiliza"). El último punto está en la lista de puntos del rectángulo circundante del suministrador (el lote que "es utilizado").

7.5 Sintaxis CIF-GR – Reglas B de CIF:

Las reglas B de CIF son reglas de utilidad referenciadas a partir de las reglas A y no corresponden por sí mismas a símbolos autónomos.

7.5.1 B1 Partes de diagrama:

```
{ <page declaration: B2>+ | <nested frame: B10> }
```

Información adicional:

Si el diagrama está insertado en el diagrama envolvente, debe utilizarse un <nested frame: B10>.

Si el diagrama no está insertado en el diagrama envolvente, debe haber una <page declaration: B2> para cada página en el diagrama.

7.5.2 B2 Declaración de página:

```
/* CIF Page <page name> <size point: B22> */  
[ <frame declaration: B11> ]  
[ <diagram heading text position: B21> ]  
[ <page text position: B19> ]  
[ <package use symbol: B14> ]
```

Información adicional:

Si una <frame declaration: B11> no está dada, el recuadro tiene la posición (0,0) y el mismo tamaño que la página. Esta página viene a ser la página actual y permanece como página actual hasta que se encuentra una nueva <page declaration: B2> o una <page switch: A21>.

La <page text position: B19> define la esquina superior *derecha* del rectángulo envolvente del texto.

El <package use symbol: B14> debe darse solamente si el símbolo de referencia de paquete es visible en SDL-GR.

Ejemplo:

```
/* CIF Page 1 (1900,2300) */  
/* CIF Frame (100,250), (1700,1950) */
```

7.5.3 B3 Símbolo de limitación de puerta:

```
<block symbol rectangle: B12> | <process symbol rectangle: B13>
```

7.5.4 B4 Posición de texto de la primera lista de señales:

```
/* CIF TextPosition <point: B22> SignalList1 */
```

Información adicional:

Esta posición de texto debe estar vinculada a la primera lista de señales mencionada en la especificación SDL-PR.

Ejemplo:

```
/* CIF TextPosition (800,550) SignalList1 */
```

7.5.5 B5 Posición de texto de la segunda lista de señales:

```
/* CIF TextPosition <point: B22> SignalList2 */
```

Información adicional:

Esta posición de texto debe estar atada a la segunda lista de señales mencionada en la especificación SDL-PR.

7.5.6 B6 Posición de la primera flecha:

```
/* CIF Arrow1Position <point: B22> */
```

Información adicional:

Esta posición de flecha se utiliza para el primer trayecto de canal mencionado en SDL-PR para <channel: A22>.

Ejemplo:

```
/* CIF Arrow1Position (800,550) */
```

7.5.7 B7 Posición de la segunda flecha:

```
/* CIF Arrow2Position <point: B22> */
```

Información adicional:

Esta posición de flecha se utiliza para el segundo trayecto de canal mencionado en SDL-PR para el <channel: A22>.

7.5.8 B8 Texto de acceso de entrada:

```
/* CIF InletText <macro label> */  
[ <text position: B21> ]
```

7.5.9 B9 Texto de acceso de salida:

```
/* CIF OutletText <macro label> */  
[ <text position: B21> ]
```

7.5.10 B10 Recuadro anidado:

```
/* CIF NestedFrame <position and size: B20> */  
[ <diagram heading text position: B21> ]
```

7.5.11 B11 Declaración de recuadro:

```
/* CIF Frame <position and size: B20> */
```

7.5.12 B12 Rectángulo de símbolo de bloque:

```
/* CIF Block <position and size: B20> */
```

7.5.13 B13 Rectángulo de símbolo de proceso:

```
/* CIF Process <position and size: B20> */
```

7.5.14 B14 Símbolo de utilización de lote:

```
/* CIF Use <position and size: B20> */  
[<text position: B21>]
```

Información adicional:

El comentario CIF se utiliza para especificar la posición del símbolo de utilización de lote. <text position> especifica la posición de la esquina superior izquierda de la casilla que delimita las cláusulas de utilización de lote.

7.5.15 B15 Referencia de puerta:

```
/* CIF GateReference [ <name> ] <connection point: B22> */  
<text position: B21>
```

Información adicional:

El comentario CIF se utiliza para especificar la posición de texto de una referencia de puerta. El punto de conexión es el punto donde la referencia de puerta conecta con canales. El nombre de la puerta es especificado en la regla o, si no está especificado, el nombre de la puerta puede encontrarse en SDL-PR para canales conectados. Véase más información sobre puertas y referencias de puertas en <channel: A22>. Véase también el ejemplo. Este comentario CIF se ha de usar para conexiones por puertas. Para conexiones directas entre canales sin puertas, utilícese <connect: A25>.

Ejemplo:

```
/* CIF GateReference (800,550) */  
/* CIF TextPosition (750,500) */
```

7.5.16 B16 Línea:

`/* CIF Line <pointlist: B18> */`

7.5.17 B17 Línea de trazo discontinuo:

`/* CIF Line <pointlist: B18> Dashed */`

7.5.18 B18 Lista de puntos:

`<point: B22> {, <point: B22> }+`

7.5.19 B19 Posición de texto de página:

`/* CIF TextPosition <point: B22> PageName */`

7.5.20 B20 Posición y tamaño:

`<position point: B22> [, <size point: B22>]`

Información adicional:

El tamaño del punto puede omitirse si se define un tamaño por defecto con `<default size: A20>`. El punto de posición es la esquina superior *izquierda* del rectángulo circundante si no se especifica otra cosa en una regla superior.

7.5.21 B21 Posición de texto:

`/* CIF TextPosition <point: B22> */`

Información adicional:

El punto define la esquina superior *izquierda* del rectángulo circundante del texto si no se especifica otra cosa en una regla superior. No se define el ancho ni el largo del texto.

Ejemplo:

```
/* CIF TextPosition (800,550) */
```

7.5.22 B22 Punto:

`(<integer>, <integer>)`

7.5.23 B23 Posición de texto de id:

`/* CIF TextPosition <point: B22> TypeRefId */`

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de la parte identificador de un símbolo de referencia de tipo.

7.5.24 B24 Posición de texto de estereotipo:

`/* CIF TextPosition <point: B22> Stereotype */`

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de la parte estereotipo de un símbolo de referencia de tipo.

7.5.25 B25 Posición de texto de atributo:

`/* CIF TextPosition <point: B22> Attribute */`

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de la parte atributo de un símbolo de referencia de tipo.

7.5.26 B26 Posición de texto de comportamiento:

```
/* CIF TextPosition <point: B22> Behavior */
```

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de la parte comportamiento de un símbolo de referencia de tipo.

7.5.27 B27 Posición de la primera línea de símbolo de clase:

```
/* CIF Line1 <point: B22> */
```

Información adicional:

Este comentario CIF se utiliza para especificar la posición de una línea que divide los compartimentos superior y medio de un símbolo de clase. El punto define el punto extremo izquierdo de la línea.

Ejemplo:

```
/* CIF Line1 (800,550) */
```

7.5.28 B28 Posición de la segunda línea de símbolo de clase:

```
/* CIF Line2 <point: B22> */
```

Información adicional:

Este comentario CIF se utiliza para especificar la posición de una línea que divide los compartimentos medio e inferior de un símbolo de clase. El punto define el punto extremo izquierdo de la línea.

Ejemplo:

```
/* CIF FirstLine (900,550) */
```

7.5.29 B29 Símbolo de icono:

```
/* CIF Icon [ <position and size: B20>] */
```

7.5.30 B30 Posición de texto de nombre de asociación:

```
/* CIF TextPosition <point: B22> Association */
```

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto del nombre de asociación.

7.5.31 B31 Posición de nombre de primer rol de fin de asociación:

```
/* CIF TextPosition <point: B22> Role1 */
```

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto del nombre de rol.

7.5.32 B32 Posición de primera visibilidad de fin de asociación:

```
/* CIF TextPosition <point: B22> Visibility1 */
```

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de visibilidad.

7.5.33 B33 Posición de primera multiplicidad de fin de asociación:

/ CIF TextPosition <point: B22> Multiplicity1 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de multiplicidad.

7.5.34 B34 Posición de primer orden de fin de asociación:

/ CIF TextPosition <point: B22> Ordering1 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de orden.

7.5.35 B35 Posición de segundo nombre de rol de fin de asociación:

/ CIF TextPosition <point: B22> Role2 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto del nombre de rol.

7.5.36 B36 Posición de segunda visibilidad de fin de asociación:

/ CIF TextPosition <point: B22> Visibility2 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de visibilidad.

7.5.37 B37 Posición de segunda multiplicidad de fin de asociación:

/ CIF TextPosition <point: B22> Multiplicity2 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de multiplicidad.

7.5.38 B38 Posición de segundo orden de fin de asociación:

/ CIF TextPosition <point: B22> Ordering2 */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto de orden.

7.5.39 B39 Posición en la lista de señales:

/ CIF TextPosition <point: B22> In */*

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto en la lista de señales asociada con un símbolo de puerta.

7.5.40 B40 Posición fuera de la lista de señales:

```
/* CIF TextPosition <point: B22> Out */
```

Información adicional:

Esta posición de texto define la esquina superior izquierda del rectángulo circundante del texto fuera de lista de señales asociada con un símbolo de puerta.

7.6 Comentarios CIF específicos de herramienta:

7.6.1 C0 Comentario CIF específico de herramienta:

```
/* CIF [ Keep ] Specific <name of tool> <tool-specific information> */
```

Información adicional:

Si se da **Keep**, este comentario CIF específico de herramienta debe ser mantenido si el objeto CIF actual es editado. Si se omite **Keep**, este comentario CIF debe ser eliminado cuando el objeto CIF vigente es editado.

Un comentario CIF específico de herramienta debe estar asociado con una regla A (igual a objeto CIF). El comentario CIF específico de herramienta debe colocarse entre los comentarios CIF y las construcciones SDL-PR asociadas con la regla A. No se define el orden entre los comentarios CIF específicos de herramienta asociados con la misma regla A.

Ejemplo informal (CIF no correcto y SDL-PR no correcto):

```
/* CIF 'The CIF comment associated with rule Ax'*/  
/* CIF Specific mySDLTool 'This information is understood by mySDLTool.  
Tools other than mySDLTool may or may not understand this information.  
This tool-specific CIF comment is associated with rule Ax'*/  
'This is the SDL-PR associated with A rule number X';
```

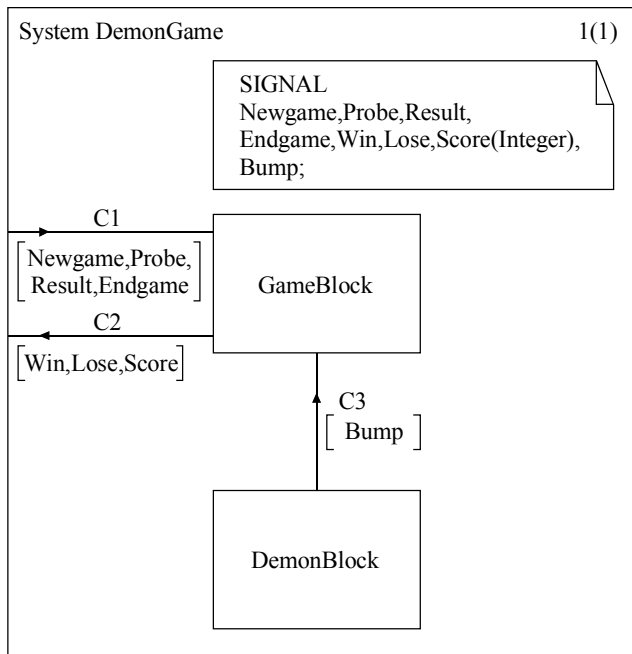
8 Ejemplos

Esta cláusula muestra algunos ejemplos SDL-GR y los correspondientes SDL-CIF. En primer lugar se presentan tres diagramas completos (aunque pequeños), y a continuación se presentan y examinan algunas construcciones SDL-GR complicadas.

Se señala que la mayoría de los ejemplos dados no son diagramas anidados, es decir, SDL-PR utiliza la palabra clave REFERENCED. Por supuesto, también está permitido en CIF expresar diagramas anidados, cuando SDL-PR no tiene la palabra clave REFERENCED.

8.1 DemonGame

8.1.1 Sistema DemonGame



Z.106_F18

```

/* CIF SystemDiagram */
/* CIF Page 1 (1150,1000) */
System DemonGame;
/* CIF Specific mySDLTool Page 1
NoGrid */
/* CIF DefaultSize (300,200) */
/* CIF Text (400,100), (600,100) */
SIGNAL
Newgame, Probe, Result,
Endgame, Win, Lose, Score(Integer), Bump;
/* CIF End Text */
/* CIF Channel (550,700), (550,500) */
/* CIF TextPosition (575,551) */
/* CIF TextPosition (575,600)
SignalList1 */
channel C3
from DemonBlock to GameBlock
with Bump;
endchannel C3;
/* CIF Channel (400,475), (0,475) */
/* CIF TextPosition (150,425) */
/* CIF TextPosition (62,500)
SignalList1 */
channel C2
from GameBlock to env
with Win, Lose, Score;
endchannel C2;
/* CIF Channel (0,325), (400,325) */
/* CIF TextPosition (212,250) */
/* CIF TextPosition (50,350)
SignalList1 */
channel C1
from env to GameBlock
with Newgame, Probe,
Result, Endgame;
endchannel C1;
/* CIF BlockSymbol (400,700) */
block DemonBlock referenced;
/* CIF BlockSymbol (400,300) */
block GameBlock referenced;
/* CIF End SystemDiagram */
endsystem DemonGame;

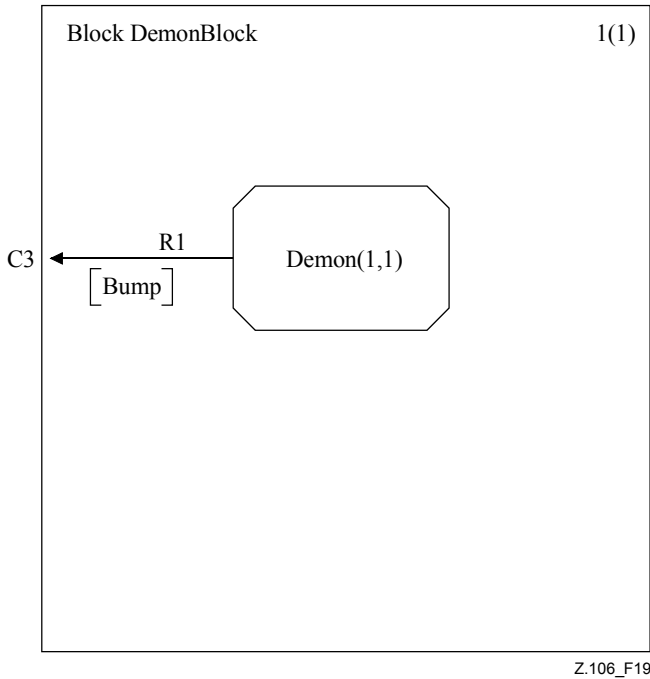
```

SDL-GR

SDL-CIF

Figura 18/Z.106 – Sistema DemonGame

8.1.2 Bloque DemonBlock



SDL-GR

```

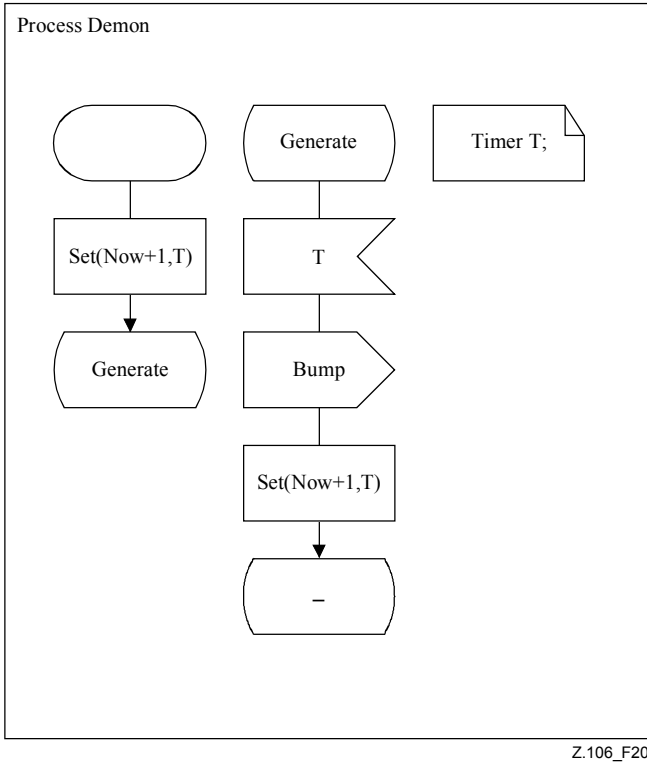
/* CIF BlockDiagram */
/* CIF Page 1 (1000,1000) */
/* CIF Frame (100,100), (800,800)
*/
Block DemonBlock;
/* CIF Specific mySDLTool Page 1
NoGrid */
/* CIF DefaultSize (300,200) */
/* CIF Connect */
/* CIF TextPosition (25,300) */
Connect C3 and R1;
/* CIF SignalRoute
(250,350), (0,350) */
/* CIF TextPosition (150,300) */
/* CIF TextPosition (75,375)
SignalList1 */
signalroute R1
from Demon to env with Bump;
/* CIF Process (250,250) */
process Demon (1,1) referenced;
/* CIF End BlockDiagram */
endblock DemonBlock;

```

SDL-CIF

Figura 19/Z.106 – Bloque DemonBlock

8.1.3 Processo Demon



SDL-GR

```

/* CIF ProcessDiagram */
/* CIF Page 1 (1400,1000) */
Process Demon;
/* CIF DefaultSize (200,100) */
/* CIF Text (800,100) */
Timer T;
/* CIF End Text */
/* CIF Start (300,100) */
start;
/* CIF Set (300,250) */
Set(Now+1, T);
/* CIF NextState (300,400) */
nextstate Generate;
/* CIF State (550,100) */
state Generate;
/* CIF Input (550,250) */
input T;
/* CIF Output (550,400) */
output Bump;
/* CIF Set (550,550) */
Set(Now+1, T);
/* CIF NextState (550,700) */
nextstate -;
/* CIF End ProcessDiagram */
endprocess Demon;

```

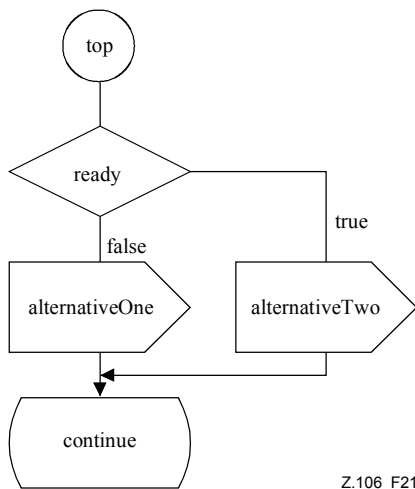
SDL-CIF

Figura 20/Z.106 – Processo Demon

8.2 Construcciones SDL intrincadas

8.2.1 Unión mediante líneas de flujo 1

Obsérvese que la lista de puntos para la línea de flujo después del símbolo de salida alternativeTwo tiene cuatro punos, dos de ellos en el borde de los símbolos.



Z.106_F21

```
/* CIF Label (50,100) ,(100,100) */
top:
/* CIF Line (100,200) ,(100,300) */
/* CIF Decision (0,300) */
decision ready;
/* CIF Answer */
/* CIF Line (100,400) ,(100,500) */
(false):
/* CIF Output (0,500) */
output alternativeOne;
/* CIF Line (100,600) ,(100,700) */
/* CIF Answer */
/* CIF Line
(200,350) ,(400,350) ,(400,500) */
(true):
/* CIF Output (300,500) */
output alternativeTwo;
/* CIF Line
(400,600) ,(400,650) ,(100,650) ,
(100,700) */
/* CIF End Decision */
enddecision;
/* CIF NextState (0,700) */
nextstate continue;
```

Figura 21/Z.106 – Unión mediante líneas de flujo

8.2.2 Unión mediante líneas de flujo 2

Este ejemplo es muy similar al anterior, de hecho es el mismo SDL-GR. Sin embargo, SDL-PR se expresa de manera diferente. Obsérvese que las dos líneas de flujo de unión están colocadas junto al símbolo del cual provienen en el siguiente SDL-PR.

```
/* CIF Label (50,100) ,(100,100) */
top:
/* CIF Line (100,200) ,(100,300) */
/* CIF Decision (0,300) */
decision ready;
/* CIF Answer */
/* CIF Line (100,400) ,(100,500) */
(false):
/* CIF Output (0,500) */
output alternativeOne;
/* CIF Line (100,600) ,(100,700) */
/* CIF Label Invisible */
grs0:
/* CIF NextState (0,700) */
nextstate continue;
/* CIF Answer */
/* CIF Line (200,350) ,(400,350) ,(400,500) */
(true):
/* CIF Output (300,500) */
output alternativeTwo;
/* CIF Line (400,600) ,(400,650) ,(100,650) ,(100,700) */
/* CIF Join Invisible */
join grs0;
```

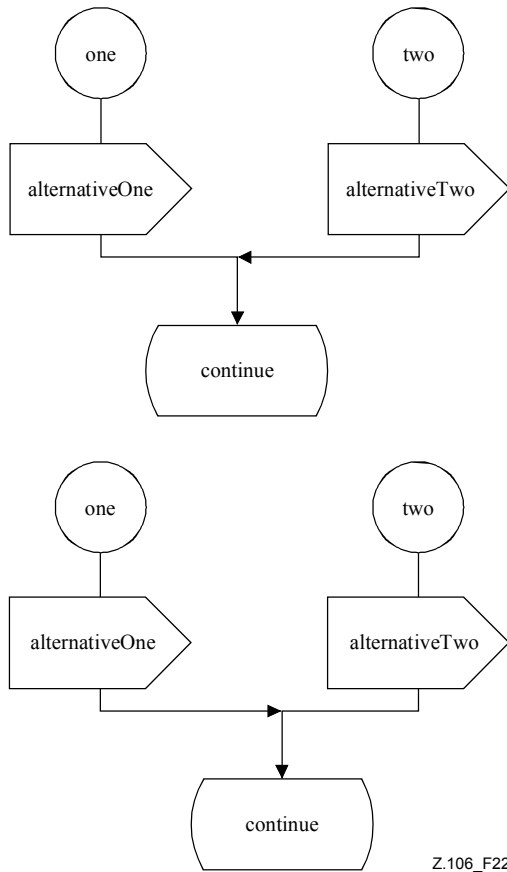
```

/* CIF End Decision */
enddecision;

```

8.2.3 Unión mediante líneas de flujo 3

CIF no define lo que une una línea de flujo. Los dos ejemplos SDL-GR siguientes pueden producir el mismo SDL-CIF. (Ejemplo SDL-PR sin líneas de flujo.)



```

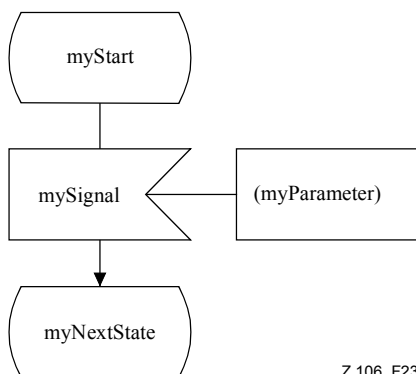
/* CIF Label (150,100), (100,100)
*/
connection
  one:
    /* CIF Output (100,300) */
    output alternativeOne;
    /* CIF Label Invisible */
    grs0:
      /* CIF NextState (250,500) */
      nextstate continue;
/* CIF End Label */
endconnection one;
/* CIF Label (450,100), (100,100)
*/
connection
  two:
    /* CIF Output (400,300) */
    output alternativeTwo;
    /* CIF Join Invisible */
    join grs0;
/* CIF End Label */
endconnection two;

```

Figura 22/Z.106 – Unión mediante líneas de flujo

8.2.4 Líneas y rectángulos envolventes

Obsérvese que la línea de extensión de texto se describe como si sus puntos extremos se encontraran en el borde de los rectángulos circundantes de los símbolos adjuntos, incluso si esto no es completamente cierto en SDL-GR. (El punto extremo en el símbolo de entrada se encuentra en el interior del rectángulo circundante.) Esta regla se aplica a todas las líneas/canales conectados a símbolos que no tienen forma rectangular. (Símbolo de parada, símbolo de proceso, etc.)



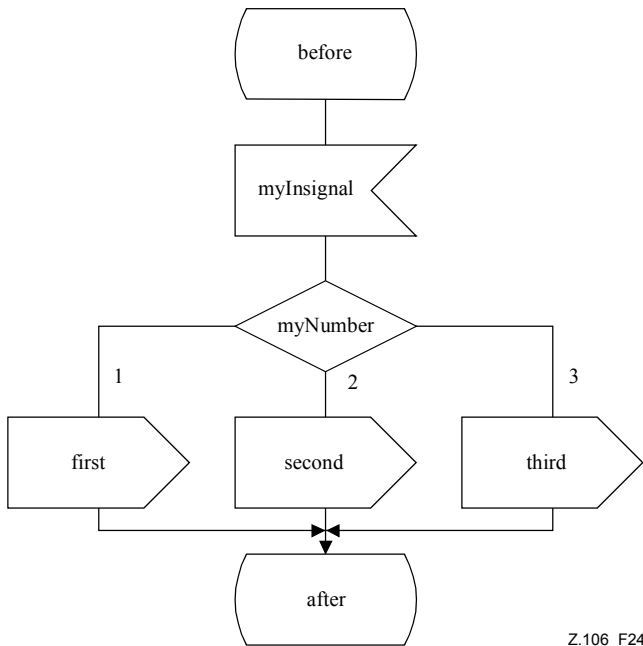
```

/* CIF State (100,100) */
state myStart;
/* Input (100,300) */
input mySignal
/* CIF TextExtension (400,300) */
/* CIF Line (400,350), (300,350) */
(myParameter)
/* CIF End TextExtension */
;
/* CIF NextState (100,500) */
nextstate myNextState;

```

Figura 23/Z.106 – Líneas y rectángulos envolventes

8.2.5 Líneas de flujo de respuesta después de decisión



Z.106_F24

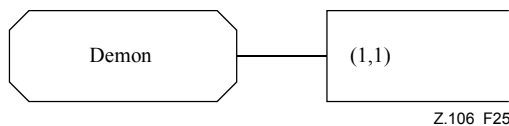
```

/* CIF State (400,100) */
state before;
/* CIF Input (400,300) */
input myInsignal;
/* CIF Decision (400,500) */
decision myNumber;
/* CIF Answer Left
InvisibleBrackets */
(1):
/* CIF Output (100,700) */
output First;
/* CIF Answer
InvisibleBrackets */
(2):
/* CIF Output (400,700) */
output second;
/* CIF Answer Right
InvisibleBrackets */
(3):
/* CIF Output (700,700) */
output third;
/* CIF End Decision */
enddecision;
/* CIF NextState (400,900) */
nextstate after;
/* CIF End Estate */
endstate;
  
```

Figura 24/Z.106 – Líneas de flujo de respuesta

8.2.6 Extensión de texto

Se permite:



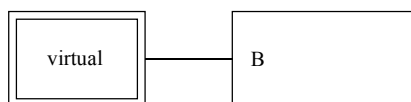
Z.106_F25

```

/* CIF Process (800,550) */
PROCESS Demon
/* CIF TextExtension (1100,550) */
(1,1)
/* CIF TextExtensionEnd */
REFERENCED;
  
```

Figura 25/Z.106 – Extensión de texto

Se permite también:



Z.106_F26

```

/* CIF BlockType (800,550) */
virtual BLOCK TYPE
/* CIF TextExtension (1100,550) */
B
/* CIF TextExtensionEnd */
REFERENCED;
  
```

Figura 26/Z.106 – Extensión de texto

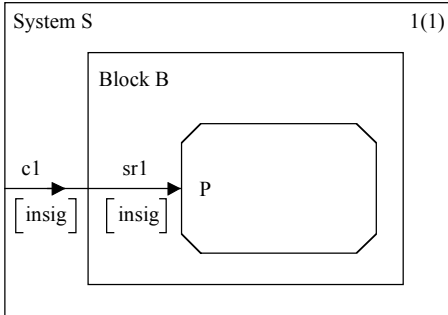
8.2.7 Diagramas anidados

Este diagrama anidado se convierte en dos diagramas con los siguientes pasos:

- 1) Reemplazar el diagrama de bloques del sistema S por un símbolo de referencia de bloque.

- 2) Crear un diagrama de bloques separado.
- 3) Insertar información acerca del canal al que se conecta la ruta de señales.
- 4) Denominar la página con un nombre apropiado.

a) *Forma anidada*

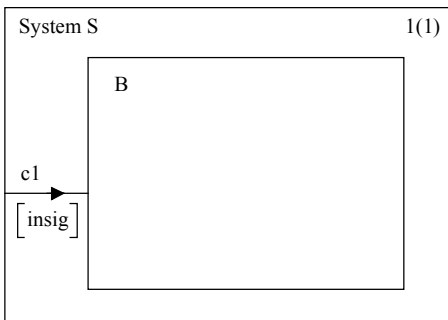


```

/* CIF SystemDiagram */
/* CIF Page 1 (600,500) */
System S;
/* CIF Channel (0,250),(100,250) */
CHANNEL c1
FROM ENV TO B WITH insig;
/* CIF BlockDiagram */
/* CIF NestedFrame (100,100),(400,300)
*/
Block B;
/* CIF Channel (100,250),(200,250) */
CHANNEL sr1
FROM ENV TO P WITH insig;
/* CIF Connect */
CONNECT c1 AND sr1;
/* CIF Process (200,200),(200,100) */
PROCESS P REFERENCED;
/* CIF End BlockDiagram */
ENDBLOCK B;
/* CIF End SystemDiagram */
ENDSYSTEM S;

```

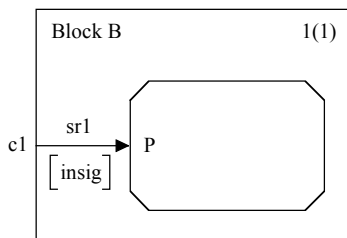
b) *Después de la conversión/sin anidado*



```

/* CIF SystemDiagram */
/* CIF Page 1 (600,500) */
System S;
/* CIF Channel (0,250),(100,250) */
CHANNEL c1
FROM ENV TO B WITH insig;
/* CIF Block (100,100),(400,300) */
BLOCK B REFERENCED;
/* CIF End SystemDiagram */
ENDSYSTEM S;

```



```

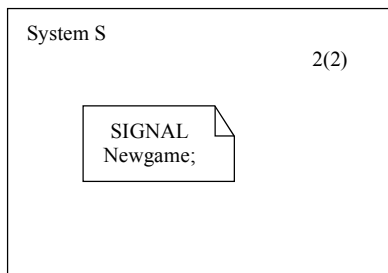
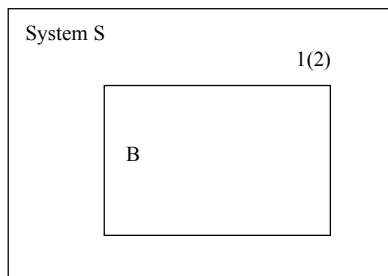
/* CIF BlockDiagram */
/* CIF Page 1 (600,500) */
/* CIF Frame (100,100),(400,300) */
Block B;
/* CIF Channel (100,250),(200,250) */
CHANNEL sr1
FROM ENV TO P WITH insig;
/* CIF Connect */
CONNECT c1 AND sr1;
/* CIF Process (200,200),(200,100) */
PROCESS P REFERENCED;
/* CIF End BlockDiagram */
ENDBLOCK B;

```

Z.106_F27

Figura 27/Z.106 – Diagramas anidados

8.2.8 Muchas páginas



F.106_F28

```
/* CIF SystemDiagram */
/* CIF Page 2 (600,300) */
/* CIF Page 1 (600,300) */
System S;
/* CIF DefaultSize (200,100) */
/* CIF Block (200,100) */
BLOCK B REFERENCED;
/* CIF CurrentPage 2 */
/* CIF Text (200,100) */
SIGNAL
NewGame;
/* CIF End Text */
/* CIF End SystemDiagram */
ENDSYSTEM S;
```

Figura 28/Z.106

8.3 Situaciones que CIF no puede tratar



Z.106_F29

Figura 29/Z.106

CIF no puede tratar esto adecuadamente. El código CIF sería:

```
/* CIF Process (800,550) Dashed myProcess */
```

El comentario en SDL-GR se pierde. En consecuencia, no se puede utilizar CIF como un formato de almacenamiento para todos los diagramas SDL-GR legales. Para tratar este ejemplo de manera apropiada, el texto completo dentro del símbolo tiene que ser almacenado dentro del comentario CIF. Esto significaría que hay que introducir caracteres escape para gestionar los testigos/caracteres de arranque de comentario, fin de comentario, arranque de texto y fin de texto dentro del texto. Sin embargo, esto se ha dejado fuera de CIF para evitar un exceso de complejidad.

Éste es un ejemplo del tipo de problemas que se plantean porque SDL-PR no se adapta exactamente a SDL-GR. También pueden darse problemas con la expresión de comentarios textuales en SDL-PR, por ejemplo, en relación con conectar, puertas y macros.

A continuación se muestra otro ejemplo de dificultades con comentarios. Esta vez se trata de un símbolo de comentario. ¿Cómo se expresaría en SDL-PR?

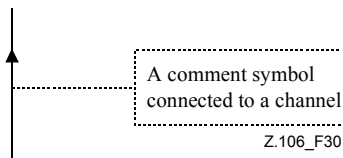


Figure 30/Z.106

Aquí se muestra un ejemplo más de SDL-GR, que puede o no ser SDL-GR legal, que en este momento ni CIF ni SDL-PR pueden gestionar. (La actuación más razonable al convertirlo a SDL-PR es duplicar el comentario/símbolo de extensión de texto. ¿Qué ocurriría al reconvertir a SDL-GR?)

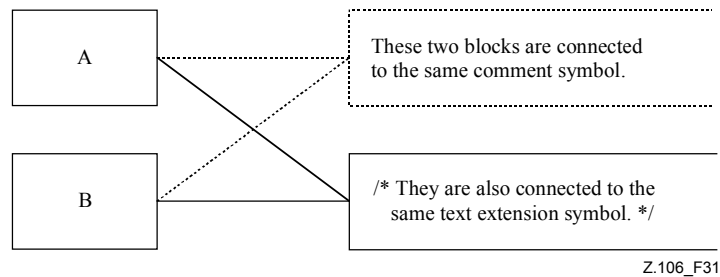


Figura 31/Z.106

Se plantea una situación similar, que CIF no puede tratar a causa de la construcción del propio CIF. Esto ocurre cuando SDL-GR contiene algo que parece CIF:

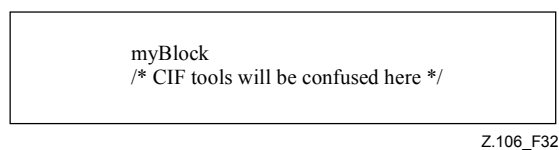


Figura 32/Z.106

9 Criterios de conformidad CIF

9.1 Acerca de herramientas lectoras de un fichero CIF

Una herramienta que lee un fichero CIF debe tratar de ser indulgente en lugar de seguir estrictamente las reglas.

Ejemplo 1: El fichero CIF dice que la página actual es X, pero no hay página X en el diagrama actual. La herramienta debe emitir un aviso y pasar por alto el comentario CIF de página vigente errónea.

Ejemplo 2: El fichero CIF describe una línea de flujo que no termina en el borde de un símbolo en ambos extremos. La herramienta debe emitir un aviso y pasar por alto el comentario CIF de línea de flujo errónea.

9.2 Presentación automática y obligada

Cuando un fichero CIF es exportado a partir de una herramienta, contiene información de posición referente a la presentación de los diagramas SDL. Cuando otra herramienta importa esta información, utilizará la información de presentación del fichero CIF. Si la herramienta no usa la información de presentación sino un mecanismo de presentación automático, entonces se ajusta al CIF nivel 1 para importación solamente. Las herramientas que sustentan el CIF deben indicar claramente si utilizan presentación obligada en la importación CIF y la forma en que sustentan la protección de información gráfica, así como los niveles CIF con los se conforman para la importación y la exportación.

9.3 Retención y uso de información específica de herramienta

El CIF se ha definido expresamente para proporcionar información específica de herramienta que puede o no ser utilizada por otra herramienta. El UIT-T recomienda que si los vendedores de herramientas usan información específica de herramienta, publiquen esta información para evitar incompatibilidad con otras herramientas. En esta Recomendación se ha proporcionado un ejemplo, y las futuras versiones del CIF pueden incluir información adicional específica de herramienta para hacer posible que otros vendedores de herramientas sustenten la importación y posible protección de esta información. No serán aplicables derechos de licencia a toda información específica de herramienta que utilice las facilidades para la información específica de herramienta suministrada por esta Recomendación.

Apéndice I

Comentarios CIF específicos de herramienta

I.1 Mantenimiento de CIF

Como es totalmente imposible prever en SDL-CIF todas las potenciales peticiones procedentes de fabricantes de herramientas en relación con directivas específicas de herramienta, la sintaxis se ha abierto hasta integrar continuamente nuevas directivas específicas de herramienta.

En primer lugar, el Grupo de Trabajo UIT-T Z.106 debe aprobar nombres de nuevas herramientas para asegurar que no hay conflicto de nombres.

Con objeto de enriquecer el SDL-CIF, los fabricantes de herramientas que han creado directrices específicas de herramientas nuevas deben proponerlas Grupo de Trabajo Z.106. De este modo, es posible que estas directrices sean introducidas en esta Recomendación, como nuevas directrices recomendadas no específicas, obligatorias u opcionales.

Tanto las realizaciones de SDL-CIF que han tenido éxito y las que no lo han tenido deben ser comunicadas al Grupo de Trabajo Z.106, con el fin de acomodarlas a la clasificación de directrices obligatorias/opcionales.

En el sitio web de la UIT se dispone de una lista actualizada de comentarios CIF específicos de herramienta utilizados por varias herramientas. Además, este apéndice proporciona una lista inicial de comentarios CIF específicos de herramienta que se conocían cuando se elaboró este apéndice.

I.2 Comentarios CIF específicos de herramienta actuales

Los comentarios CIF específicos de herramienta definidos más adelante no forman parte de la norma CIF. Constituyen los comentarios CIF específicos de herramienta actualmente conocidos.

El modo en que una herramienta debe manejar los comentarios CIF específicos de herramienta se define en la subcláusula relativa a los comentarios CIF específicos de herramienta en la presente Recomendación.

En este apéndice se utilizará una herramienta SDL imaginaria que tiene un <tool name> igual a mySDLTool y un diseñador de herramienta debe sustituir mySDLTool por un nombre apropiado que identifique inequívocamente a la herramienta.

I.2.1 Colocación de los comentarios CIF específicos de herramienta

Este tema se examina en la información adicional asociada con <tool-specific CIF comments: C0>.

I.2.2 Ejemplo

En el ejemplo a continuación, muchos comentarios CIF específicos de herramienta están asociados con la regla CIF <system diagram: A6>. Más adelante en este ejemplo figura información detallada para cada comentario CIF específico de herramienta mencionado en este apéndice.

```
/* CIF SystemDiagram */
/* CIF Page 1 (1900,2300) */
/* CIF Page 2 (1900,2300) */
/* CIF Frame (100,100),(1700,2100) */
/* CIF Specific mySDLTool Version 1.0 */
/* CIF Specific mySDLTool Page 1 Scale 200 AutoNumbered
FixedHeadingSize (200,100) */
/* CIF Specific mySDLTool Page 2 Scale 200 AutoNumbered */
/* CIF Specific mySDLTool OriginalFileName 'mysystem.ssy' */
SYSTEM mySystem;
/* CIF CurrentPage 1 */
/* CIF Text (800,550),(200,100) */
/* CIF Specific mySDLTool FixedSize (200,100) */
dcl
  MyNo Integer;
timer
  DoorTimer;
/* CIF End Text */
...
```

I.2.3 C1 Número de versión de herramienta:

```
/* CIF Specific mySDLTool Version x.y */
<diagram description: A2>
```

Asociación:

Este comentario CIF específico de herramienta está asociado con una regla A mencionada en la regla para <diagram start: A3>.

Ejemplo (en el que esta regla CIF específica de herramienta está asociada con <system diagram start: A6>):

```
/* CIF SystemDiagram */
/* CIF Page 1 (1900,2300) */
/* CIF Frame (100,100),(1700,2100) */
/* CIF Specific mySDLTool Version 1.2 */
SYSTEM mySystem;
```

I.2.4 C2 Fichero original:

```
/* CIF Specific mySDLTool OriginalFileName <file name char string literal> */
<diagram start: A3>
```

Asociación:

Este comentario CIF específico de herramienta está asociado con una regla A mencionada en la regla para <diagram start: A3>.

Información adicional:

Este comentario se utiliza para recordar el nombre del fichero binario que se convirtió a CIF. Cuando el fichero CIF se convierte a fichero binario de nuevo, se puede reutilizar el mismo nombre. Solamente se dará el nombre de fichero (a.ssy) en <file name charstring literal>, y no el trayecto completo (/home/lat/a.ssy).

Ejemplo (en el que esta regla CIF específica de herramienta está asociada con <system diagram start: A6>):

```
/* CIF SystemDiagram */
/* CIF Page 1 (1900,2300) */
/* CIF Frame (100,100),(1700,2100) */
/* CIF Specific mySDLTool OriginalFileName 'mysystem.ssy' */
SYSTEM mySystem;
```

I.2.5 C3 Especificación de detalles de página:

```
/* CIF Specific mySDLTool Page <page number> [ ShowMeFirst ] [ Scale <integer> ] [ Grid
<point: B22> ] [ AutoNumbered ] [ FixedHeadingSize [ <fixed size point: B22> ] ] */
```

Asociación:

Este comentario CIF específico de herramienta está asociado con una regla A mencionada en la regla para <diagram start: A3>. Este comentario CIF específico de herramienta aparece una vez o no aparece para cada <page declaration: B2> en el diagrama.

Información adicional:

Este comentario CIF específico de herramienta se utiliza para dar información adicional de una página ya definida:

- Si esta página debe aparecer como página por defecto cuando el diagrama se carga en un editor (se da la palabra clave ShowMeFirst) o no debe aparecer (no se da la palabra clave ShowMeFirst).
- Especificar la escala utilizada en el editor SDL cuando se presenta el diagrama. La escala se expresa como porcentaje del tamaño normal. Si no se da la escala, se utiliza 100%.
- El tamaño de la rejilla de la página. Si no se da la rejilla, ésta será (50,50), que es igual a ninguna rejilla.
- Si se debe utilizar autonumeración (se da la palabra clave AutoNumbered) o no (no se da la palabra clave AutoNumbered). AutoNumbered significa que la herramienta guarda la huella de la página numerando del siguiente modo: Inicialmente, la primera página declarada autonumerada tomará el número "1", la segunda página declarada autonumerada tomará el número "2", etc. Más adelante, cuando el usuario añade una página autonumerada antes de la página "2", la nueva página tomará el número "2" y todas las páginas autonumeradas que siguen a página "1" verán su número aumentado en una unidad. En <page declaration: B2> se debe dar un único <page name charstring literal> para la página, incluso si la página es autonumerada para poder referirse a la página desde otros comentarios CIF.

Además, si se da FixedHeadingSize, este comentario CIF específico de herramienta dice que se debe dar al símbolo de encabezamiento adicional un tamaño fijo y no ajustarse al tamaño del texto. Esto significa que el texto completo no será visible. El usuario tiene que pulsar clic dos veces en el símbolo de encabezamiento adicional para ampliar el símbolo al tamaño mayor. La herramienta SDL usa este comentario para símbolos de encabezamiento adicionales que el usuario ha "desintegrado" con un doble clic.

Ejemplo (en el que esta regla CIF específica de herramienta está asociada con <system diagram start: A6>):

```
/* CIF SystemDiagram */
/* CIF Page 1 (1900,2300) */
/* CIF Page 2 (1900,2300) */
/* CIF Frame (100,100), (1700,2100) */
/* CIF Specific mySDLTool Page 1 Scale 200 AutoNumbered */
/* CIF Specific mySDLTool Page 2 Scale 200 AutoNumbered */
SYSTEM mySystem;
/* CIF Text (800,550) */
SIGNAL Newgame;
/* CIF End Text */
```

1.2.6 C4 Tamaño fijo:

```
/* CIF Specific mySDLTool FixedSize [ <fixed size point: B22> ] */
```

Asociación:

Este comentario específico de herramienta está asociado con <text symbol: A68>.

Información adicional:

Este comentario CIF específico de herramienta dice que aunque se había dado un tamaño al símbolo que acaba ser definido en un comentario CIF (un tamaño que hace visible el texto completo), se utiliza este tamaño menor fijo para presentar el símbolo. Esto significa que el texto completo no será visible. El usuario debe pulsar dos veces sobre el símbolo para ampliarlo al tamaño mayor. La herramienta SDL utiliza este comentario para símbolos de texto que el usuario ha "desintegrado" con un doble clic.

Ejemplo:

```
/* CIF Text (800,550), (200,200) */
/* CIF Specific mySDLTool FixedSize (200,100) */
dcl
  MyNo Integer;
timer
  DoorTimer;
/* CIF End Text */
```


SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedia
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedia
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos y comunicación entre sistemas abiertos
Serie Y	Infraestructura mundial de la información y aspectos del protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación