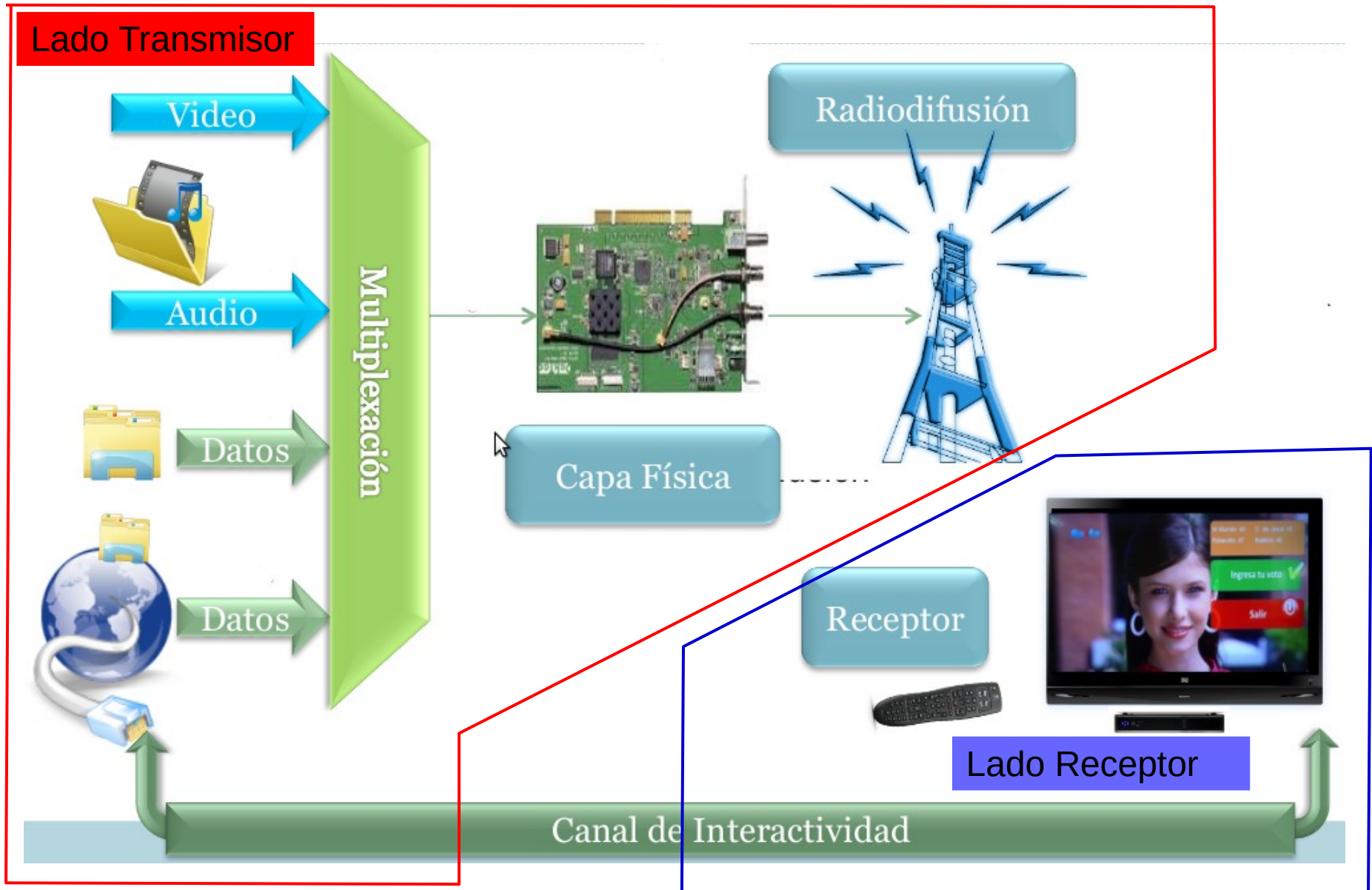


GINGA-NCL

Ginga-NCL

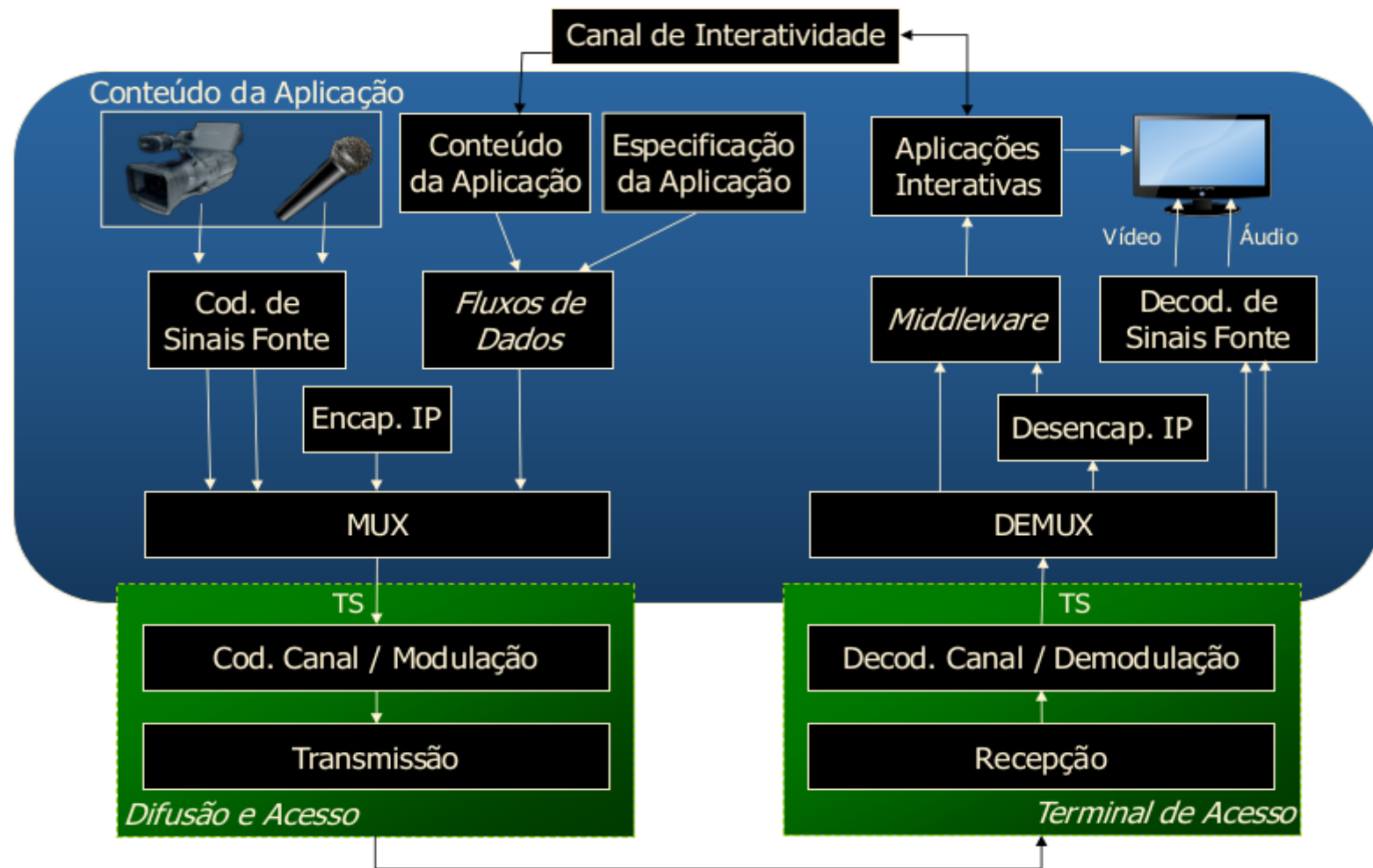
- Los contenidos digitales transmitidos en la norma ISDB-Tb incluyen programas de televisión y **datos**.
 - ISDB-Tb: Integrated Services Digital Broadcasting – Transmisión Terrestre modificación Brasileña
- Los **datos** pueden ser actualizaciones de software de receptores o sistemas de archivos (con aplicaciones y archivos).
- Ginga es el nombre que recibe en la especificación el middleware que permite ejecutar aplicaciones NCL/Lua (También Java, pero con menos soporte).
- Ginga es el nombre del middleware abierto desarrollado por Brasil.

Televisión Digital Interactiva



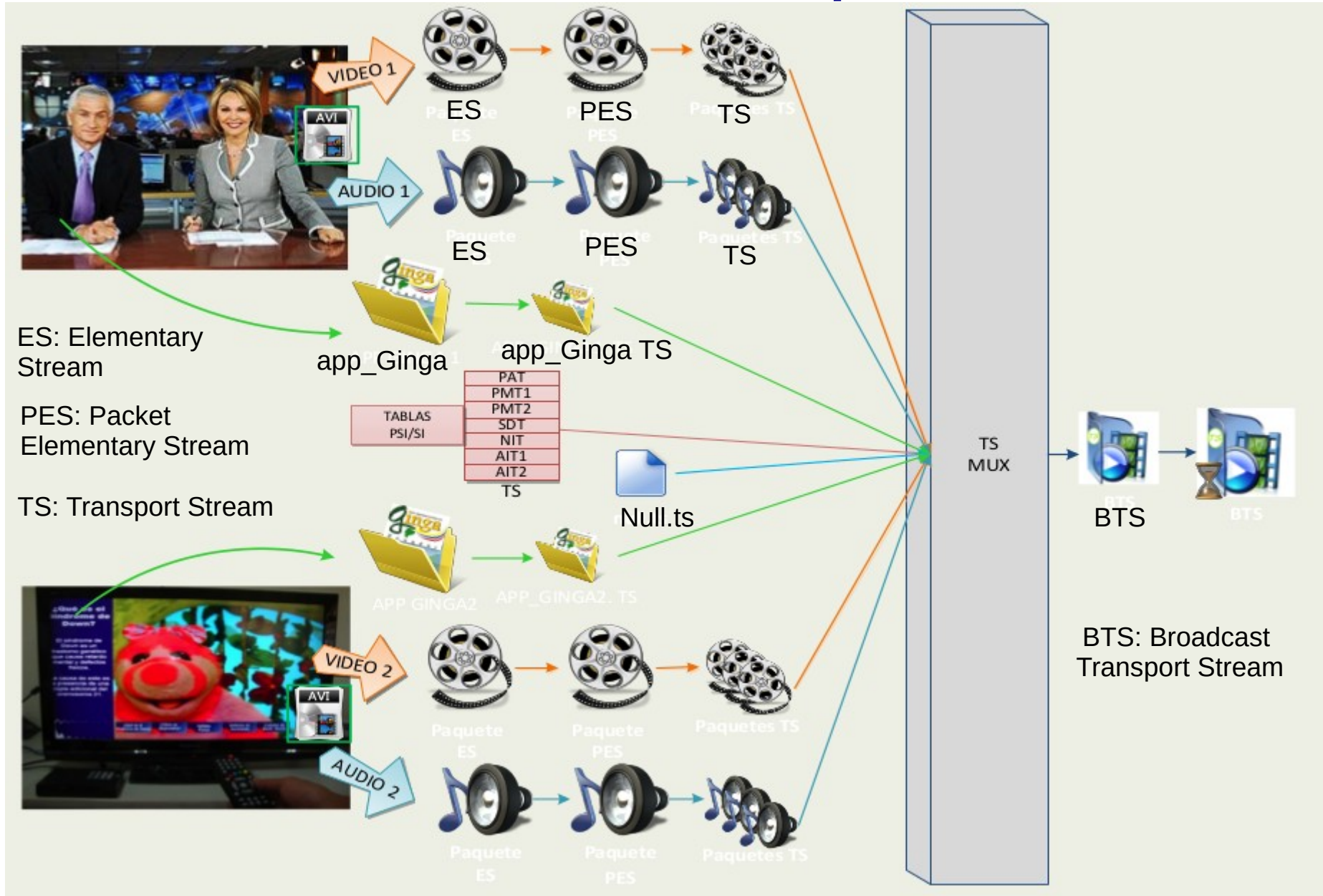
Otra vista equivalente a ésta pero más esquemática es la siguiente

Sistema de TV Digital Terrestre



- El flujo de transporte o “Transport Stream ”(TS) es una abstracción, un flujo orientado a comunicar o almacenar uno o más programas y otros datos en ambientes donde hay errores (bits o paquetes perdidos).

Lado Transmisor: Generación de **T**ransport **S**tream



Nivel de encapsulamiento para un medio

ES Level

(Elementary Stream)



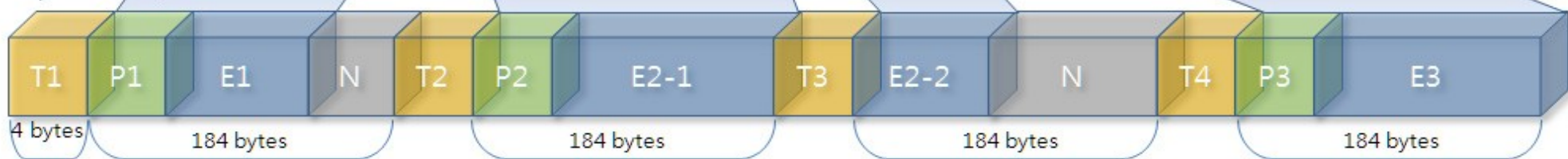
PES Level

(Packetized Elementary Stream)



TS Level

(Transport Stream)



TS header



PES header

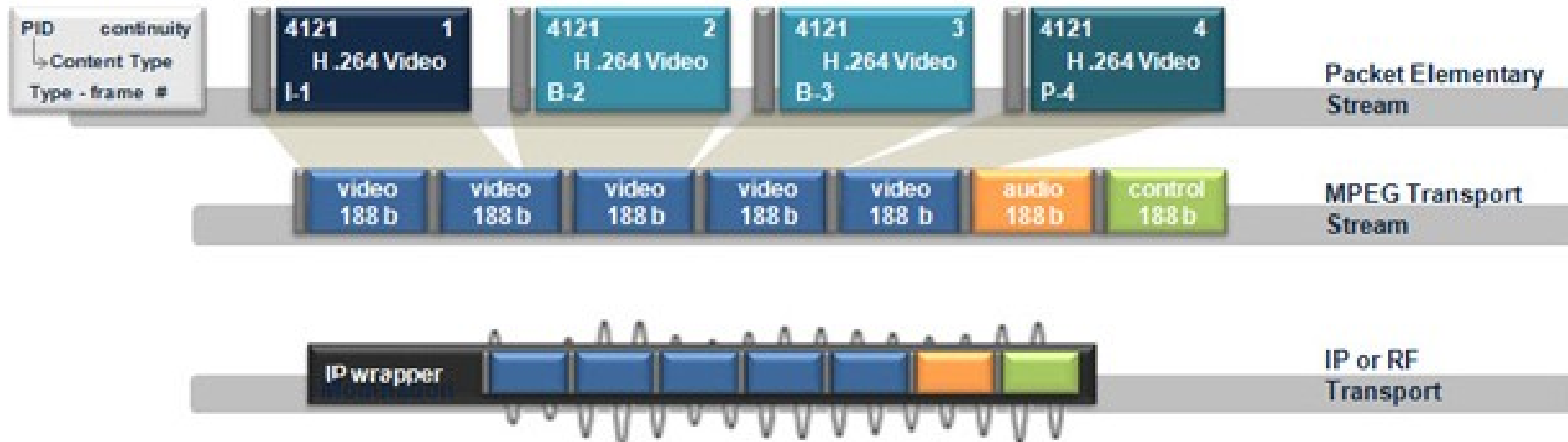
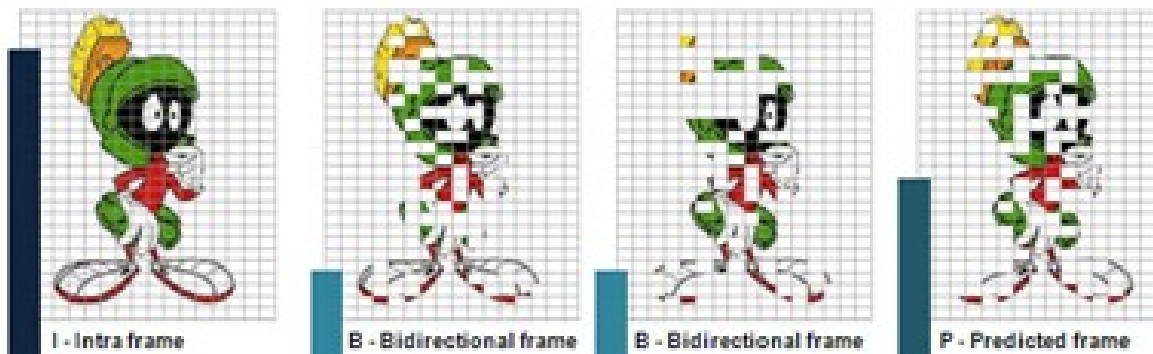


ES

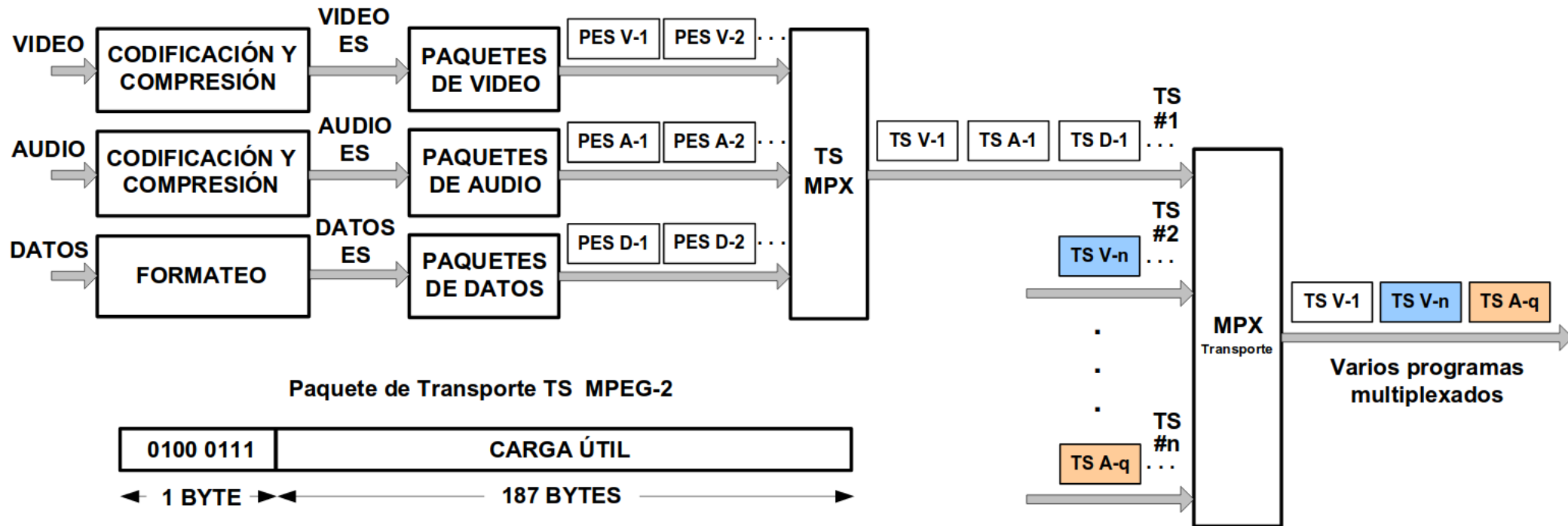


Null data

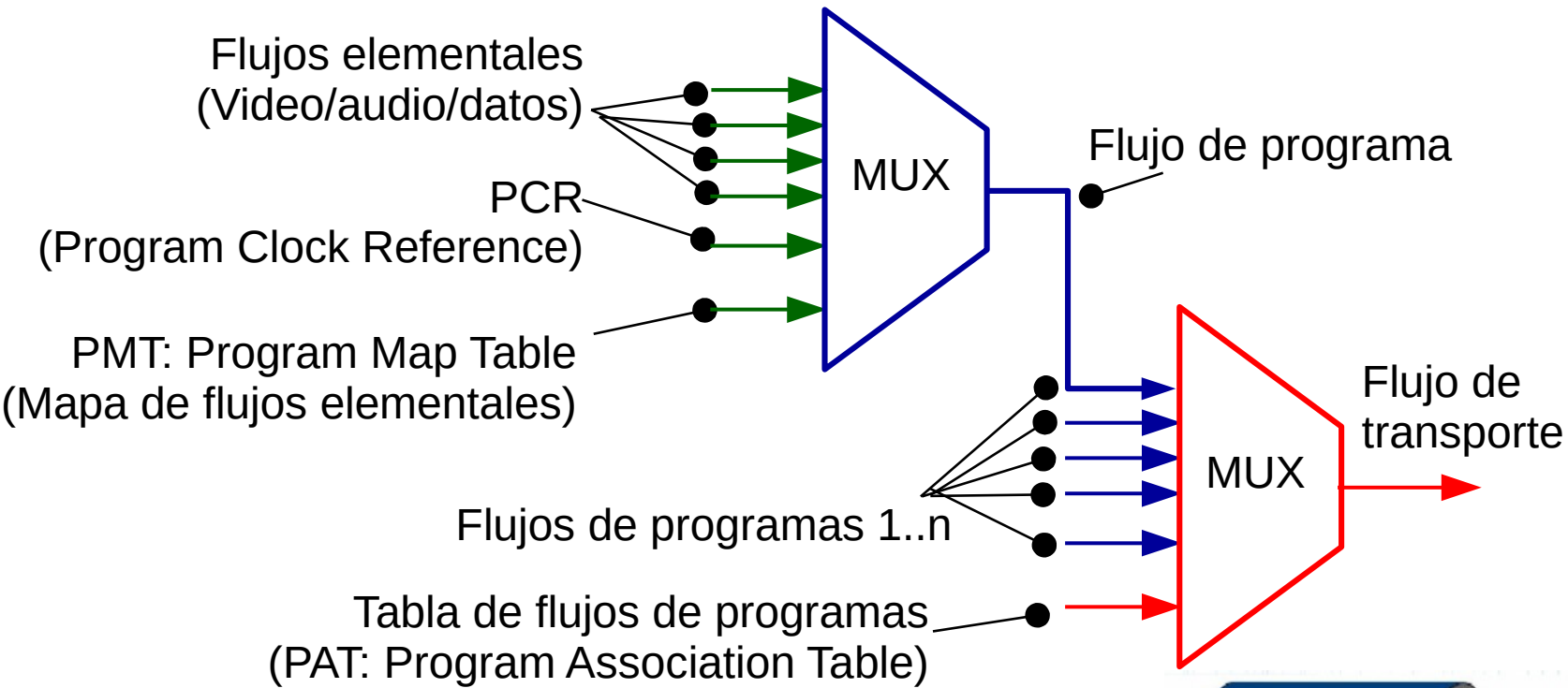
Caso Video MPEG2



Paquetes de transporte TS y multiplexación

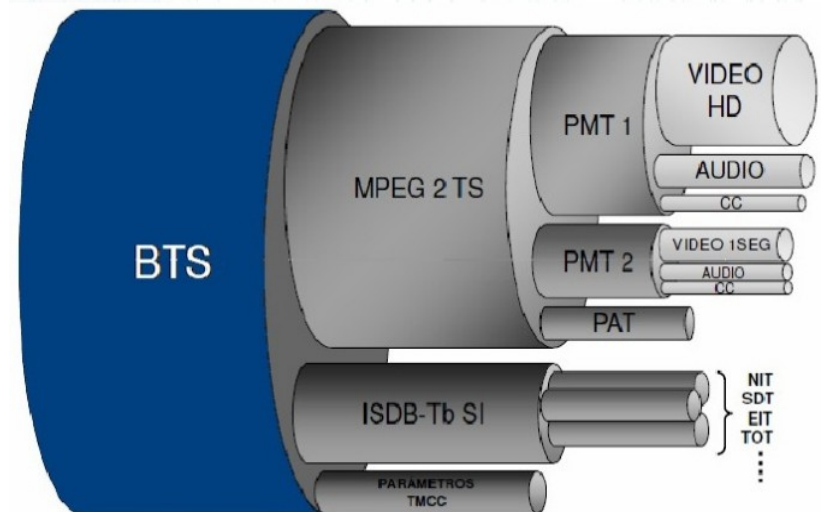


Generación de Flujo de transporte (Transport Stream) MPEG-2

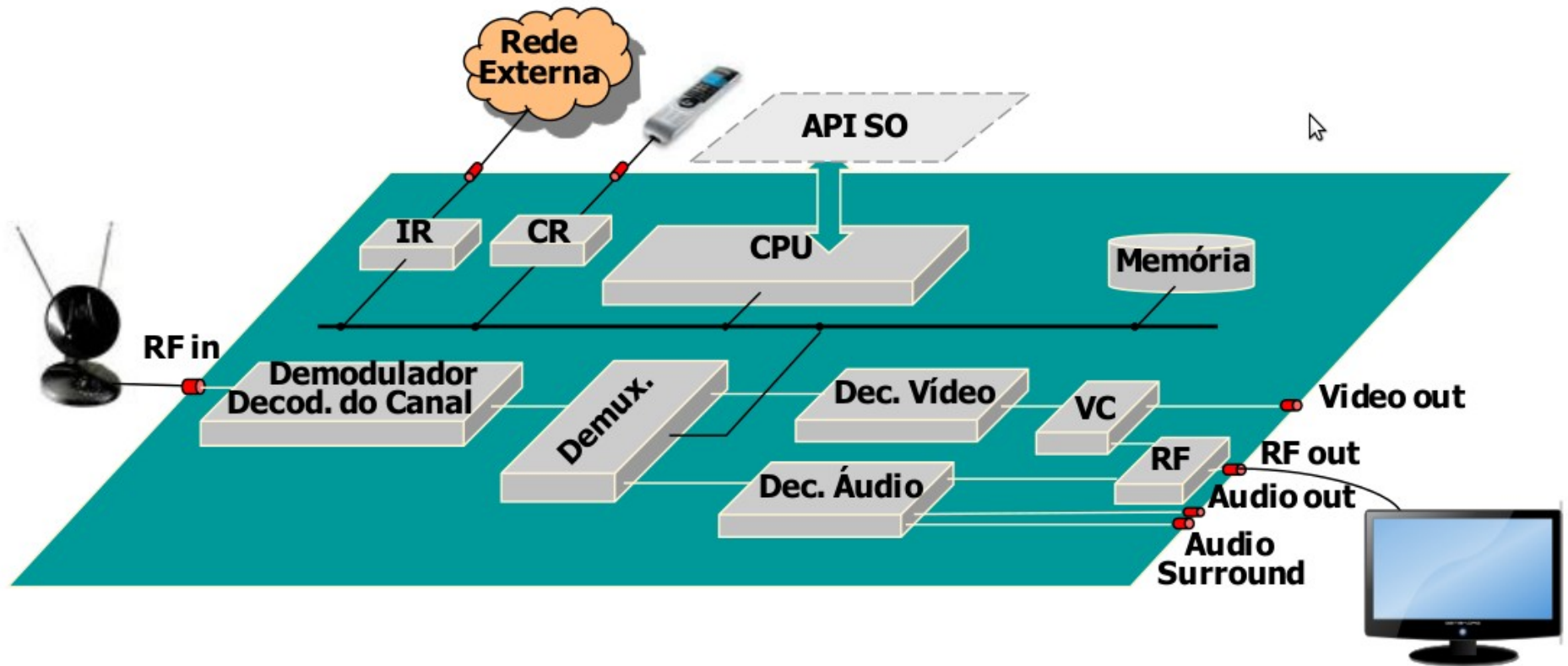


SI: Service Information, de varios tipos:

- NIT: Network Information Table
- SDT: Service Description Table
- EIT: Event Information Table

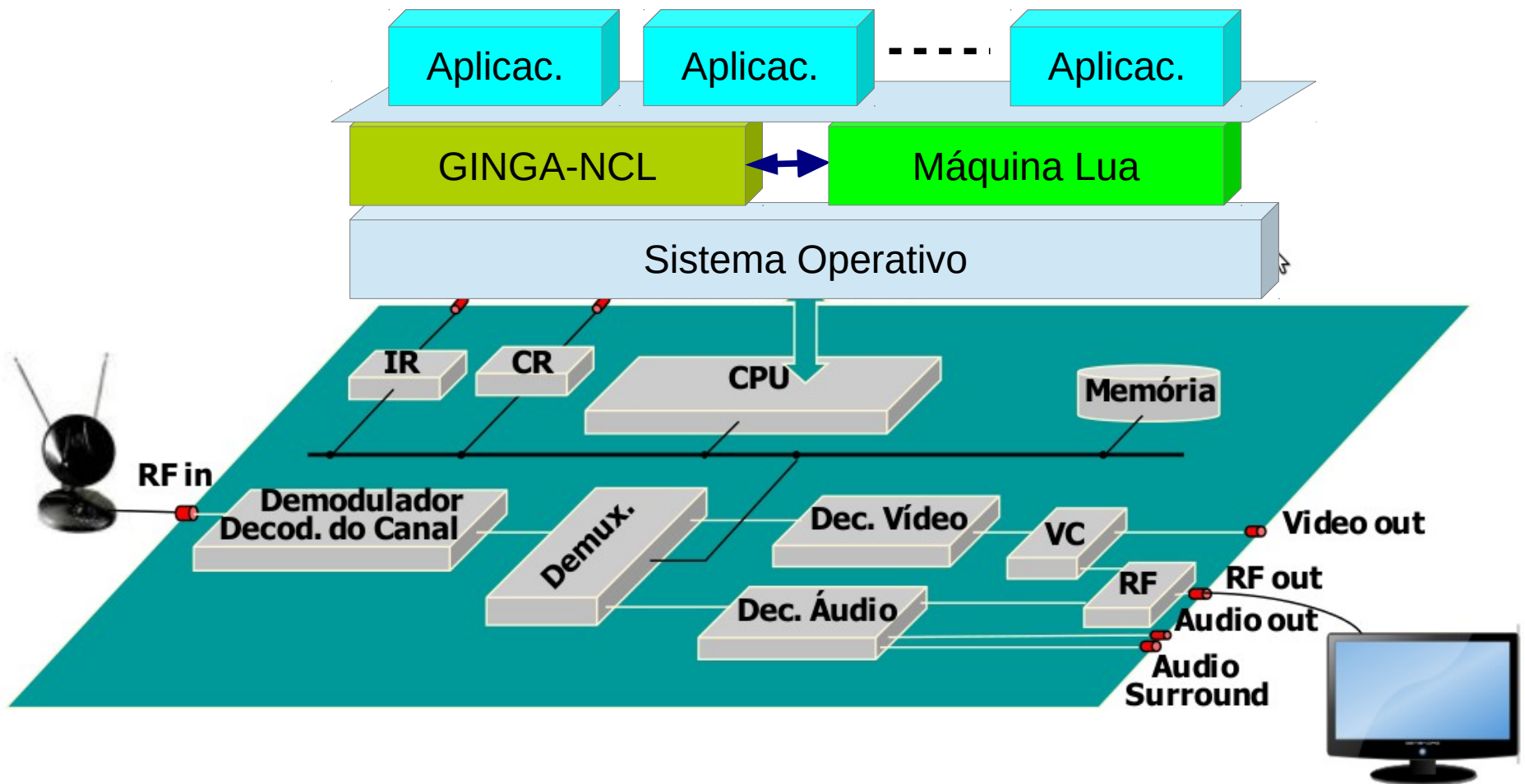


Lado Receptor: Receptor de TV Digital o Set-top-box



Lado Receptor: Receptor de TV Digital o Set-top-box

Arquitectura completa



NCL (Nested Context Language)

- NCL es un lenguaje declarativo. Los lenguajes declarativos se focalizan en qué deseamos hacer más que en cómo debe hacerse.
- Permite construir aplicaciones interactivas manipulando medios.
- Un medio pueden ser un: video, audio, imagen, una aplicación ncl (por eso se dice anidado), una página html, un programa Lua, texto, ...
- La construcción de un programa NCL se reduce a definir lo siguiente: **Qué medio mostrar, dónde va, cómo va y cuándo va.**

Ejemplo

- Qué medio? → Una imagen
- Dónde? → En el centro de la pantalla
- Cómo? → Que aparezca gradualmente
- Cuándo? → después de 2 segundos
- El lenguaje ofrece mecanismos para expresar cada una de estas cosas.
- La estructura general de todo programa ncl tiene los siguientes elementos

Estructura general

```
<ncl>
```

```
  <head>
```

```
    :
```

```
  </head>
```

```
  <body>
```

```
    :
```

```
  </body>
```

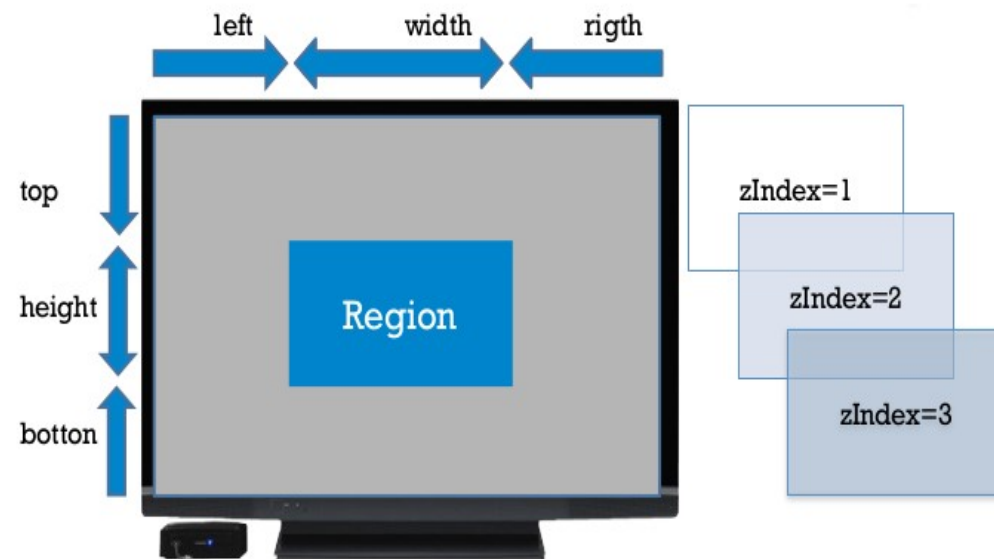
```
</ncl>
```

Por más detalles, ver aquí:

<http://www2.elo.utfsm.cl/~elo323/2013/ncl.html#ncl>

Dónde mostrar un medio

- Para esto se define las **regiones**:
- Las regiones se definen en la sección head.
- Atributos:
 - Id: identifica la región, es su nombre.
 - Height
 - width
 - Left, right, top, bottom
 - Zindex: para especificar profundidad



Ejemplo de región:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="region_pantalla" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
<!-- Crea región en pantalla al 100%-->
<!--CABECERA-->
<head>
  <!--REGIÓN-->
<regionBase>
  <region id="rg1" width="100%" height="100%" zIndex="1"/>
</regionBase>
</head>

<!--CUERPO-->
<body>
  <!--MULTIMEDIA-->
</body>
</ncl>
```


Qué mostrar

- Elemento multimedia, **<media>** se incluyen en la sección body.
- Atributos:
 - Id : nombre para el medio
 - Descriptor: nombre del descriptor que especifica dónde, cómo.
 - Scr: “source”, path donde se ubica el elemento multimedia
 - Type: define el tipo, es opcional.

Ejemplo de región con medio

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="region_pantalla" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
<!--Presenta video en pantalla al 100%-->
<head>
  <!--REGIÓN-->
  <regionBase>
    <region id="rg1" width="100%" height="100%" zIndex="1"/>
  </regionBase>
  <!--DESCRIPTOR-->
  <descriptorBase>
    <descriptor id="desc1" region="rg1"/>
  </descriptorBase>
</head>

<body>
  <!--MULTIMEDIA-->
  <media id="video" src="../media/video1.mp4" descriptor="desc1"/>
</body>
</ncl>
```

Cómo mostrar

- Se usan **descriptores**, éstos van en el encabezado.
- Atributos:
- Id: como en casos previos
- Region: se debe poner un identificador de región existente
- ExplicitDur: define la duración del objeto multimedia

Cómo parte el programa

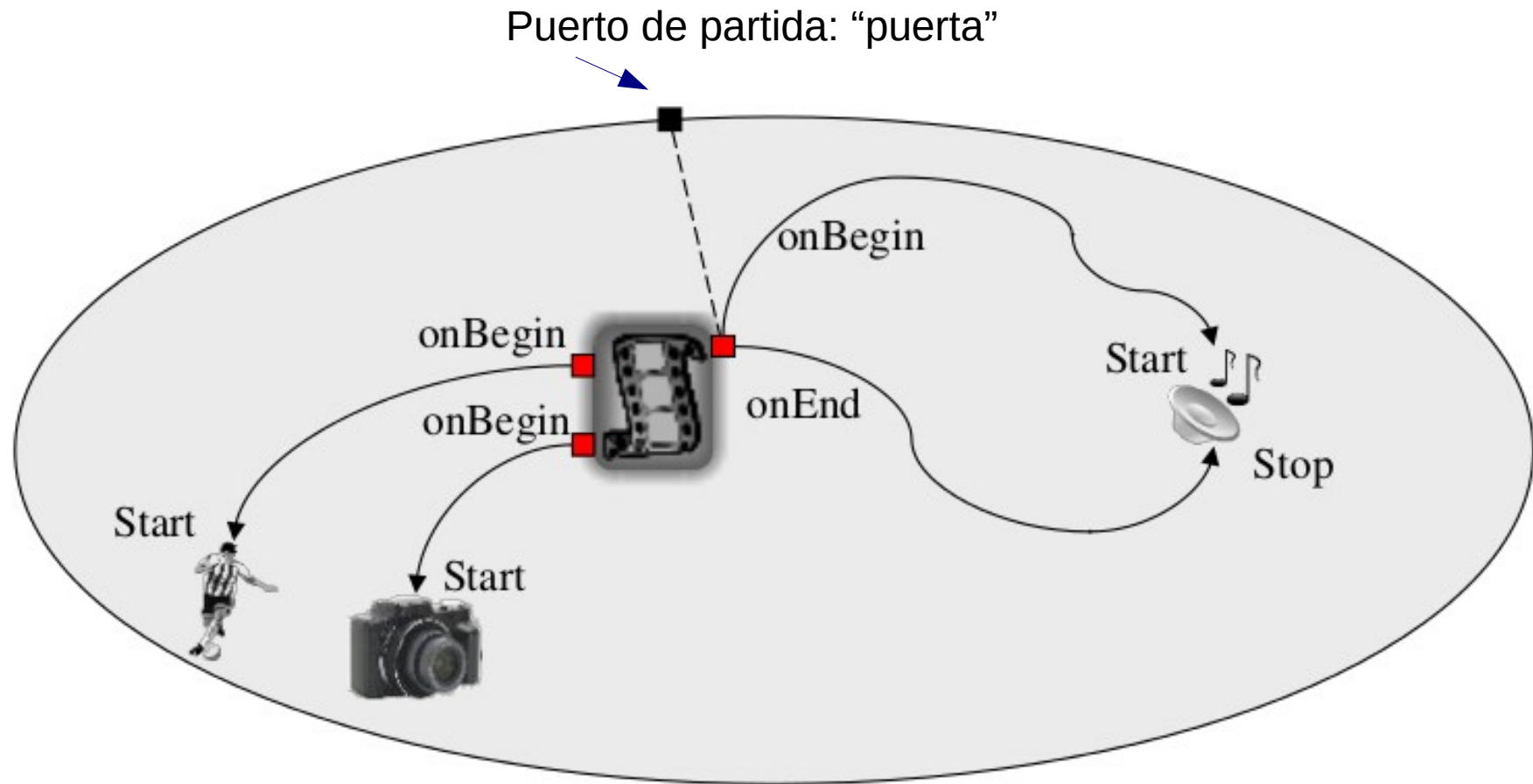
- Para definir el punto de partida, se usan las puertas <port>
- Atributos:
- Id: como antes
- Component: identificador de un elemento multimedia

1er: Programa completo ejecutable

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="region_pantalla" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
<!--CABECERA-->
<head>
  <!--REGÓN ¿Dónde? -->
  <regionBase>
    <region id="rg1" width="100%" height="100%" zIndex="1"/>
  </regionBase>
  <!--DESCRIPTOR ¿Cómo? En este caso no somos muy descriptivos -->
  <descriptorBase>
    <descriptor id="desc1" region="rg1"/>
  </descriptorBase>
</head>

<body>
  <!--MULTIMEDIA-->
  <media id="video" src="../../media/video1.mp4" descriptor="desc1"/>
  <!--PUERTA-->
  <port id="penrada" component="video"/>
</body>
</ncl>
```

Visión estructural de una aplicación



Sólo como referencia, no corresponde al programa previo.

Revisaremos otros programas usando Eclipse