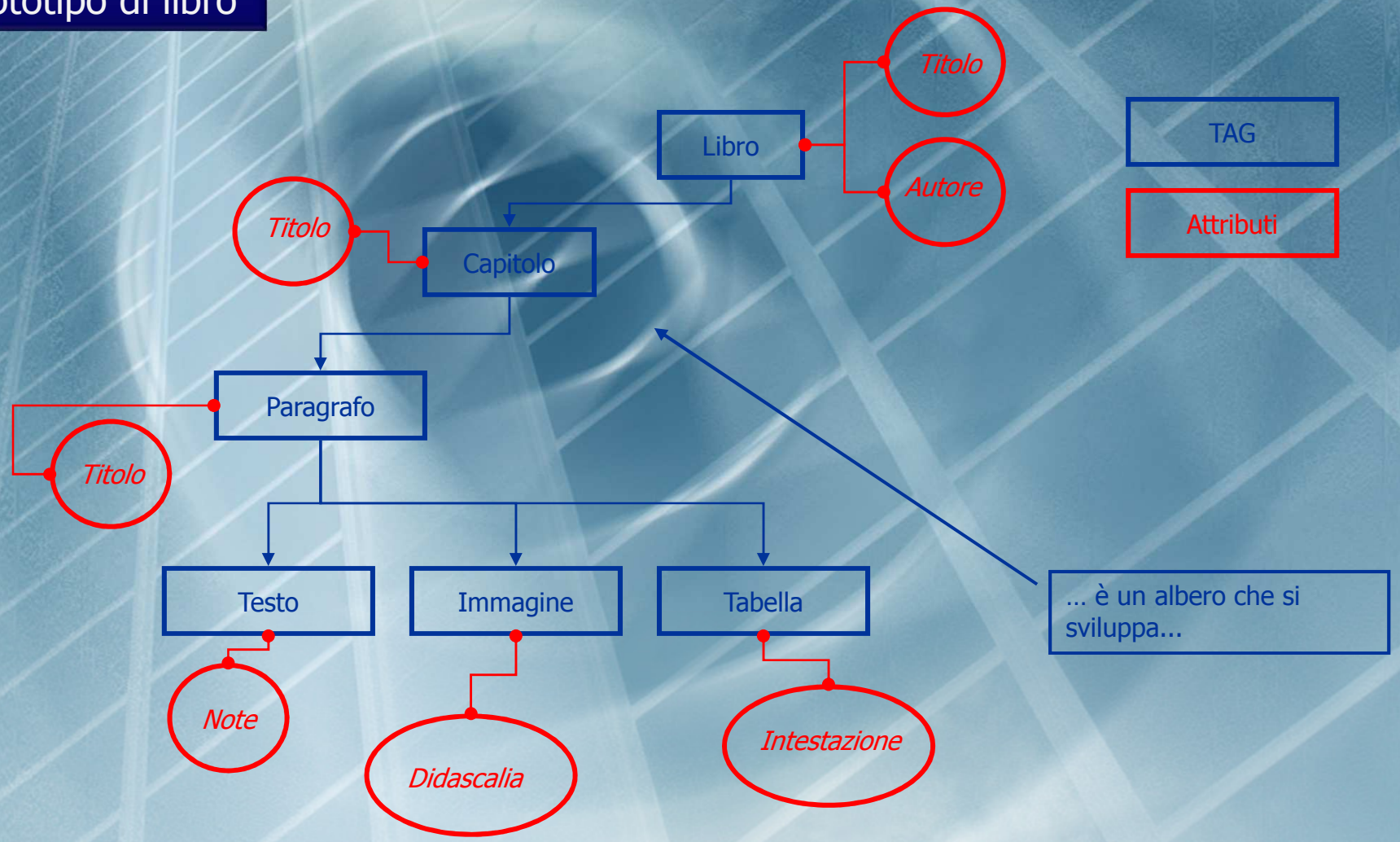
Tecniche Multimediali

Corso di Laurea in «Informatica» - aa 2010-2011

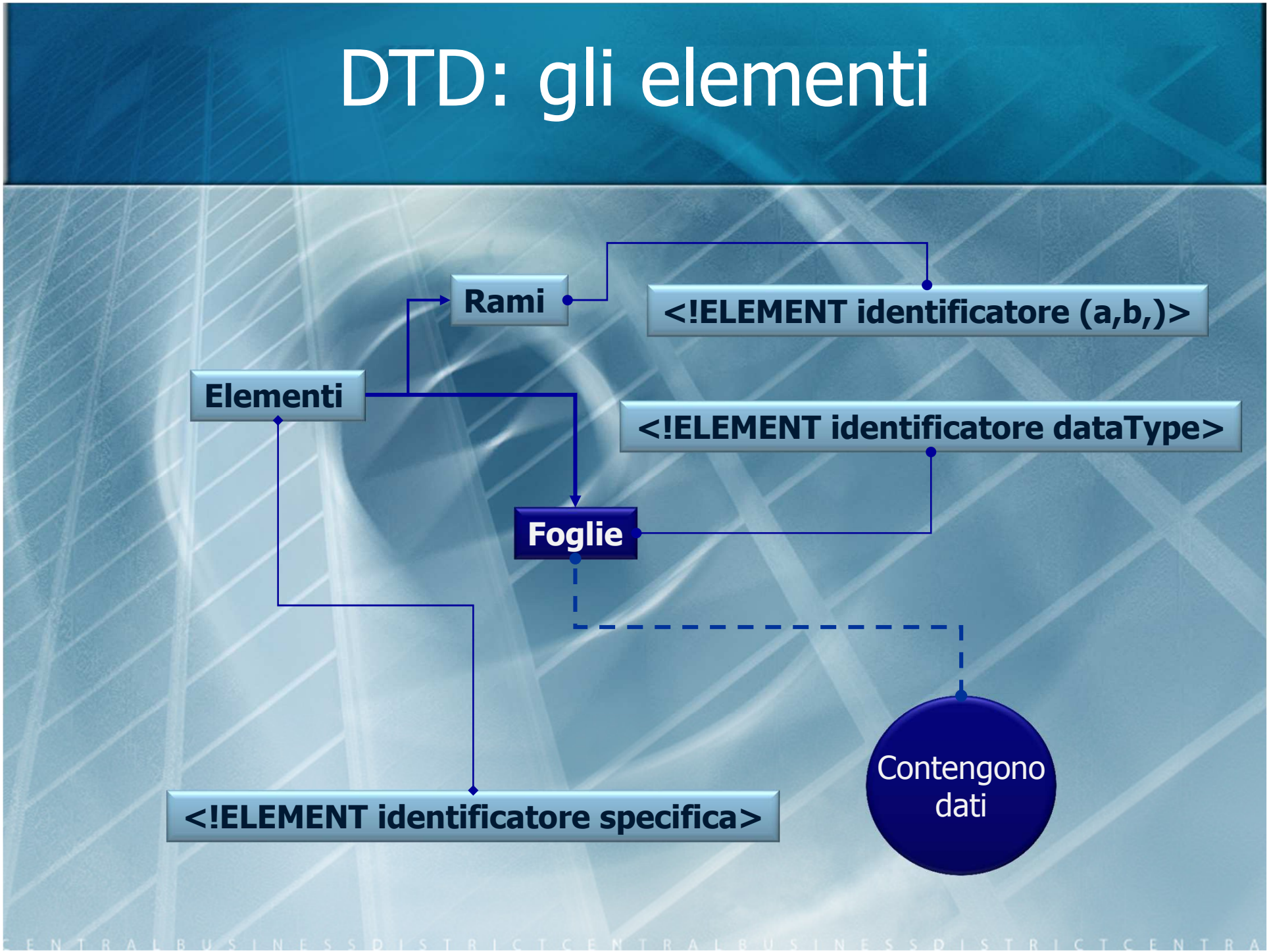
Prof. Giorgio Poletti – giorgio.poletti@unife.it

XML struttura di un documento

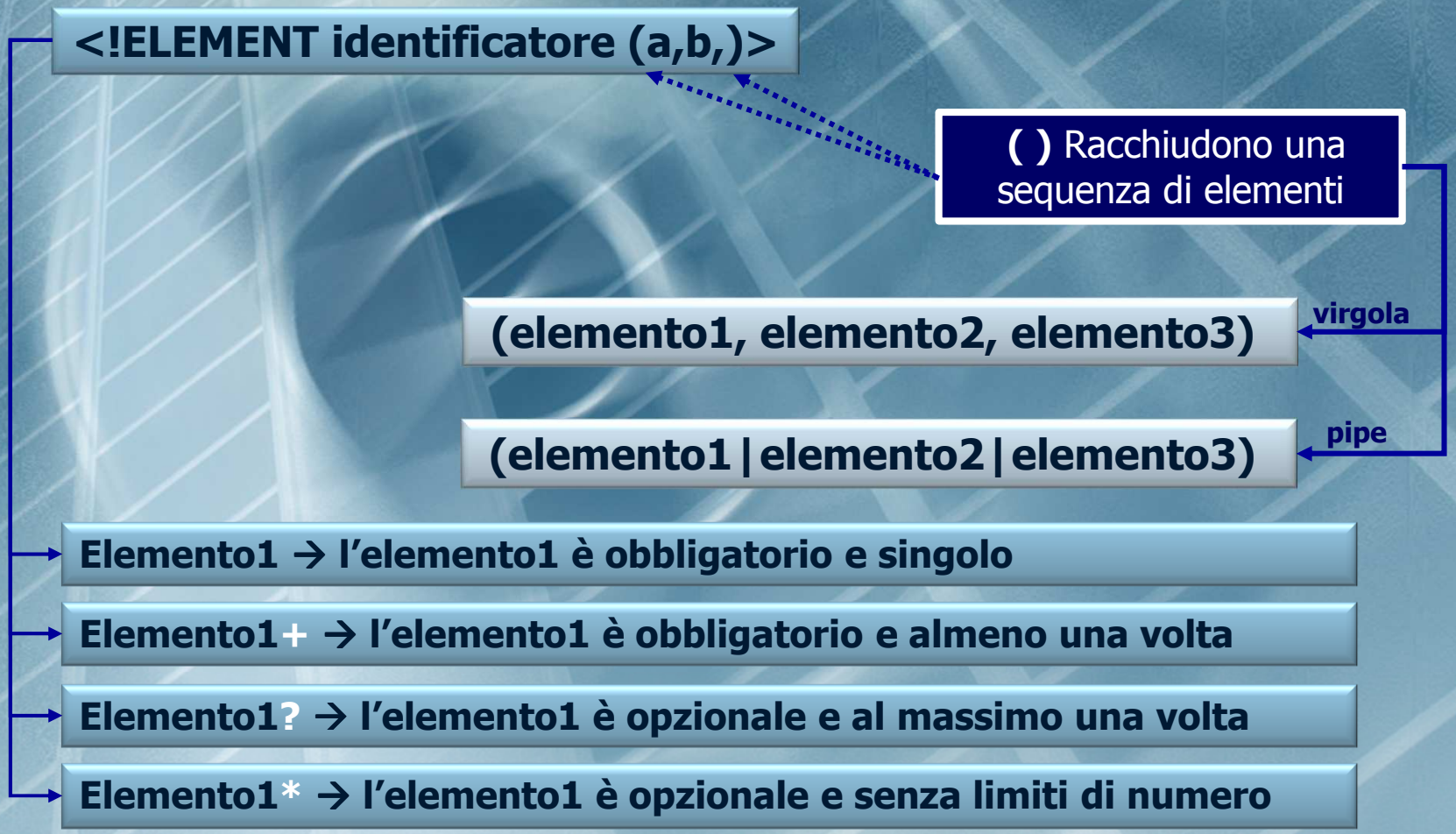
Prototipo di libro



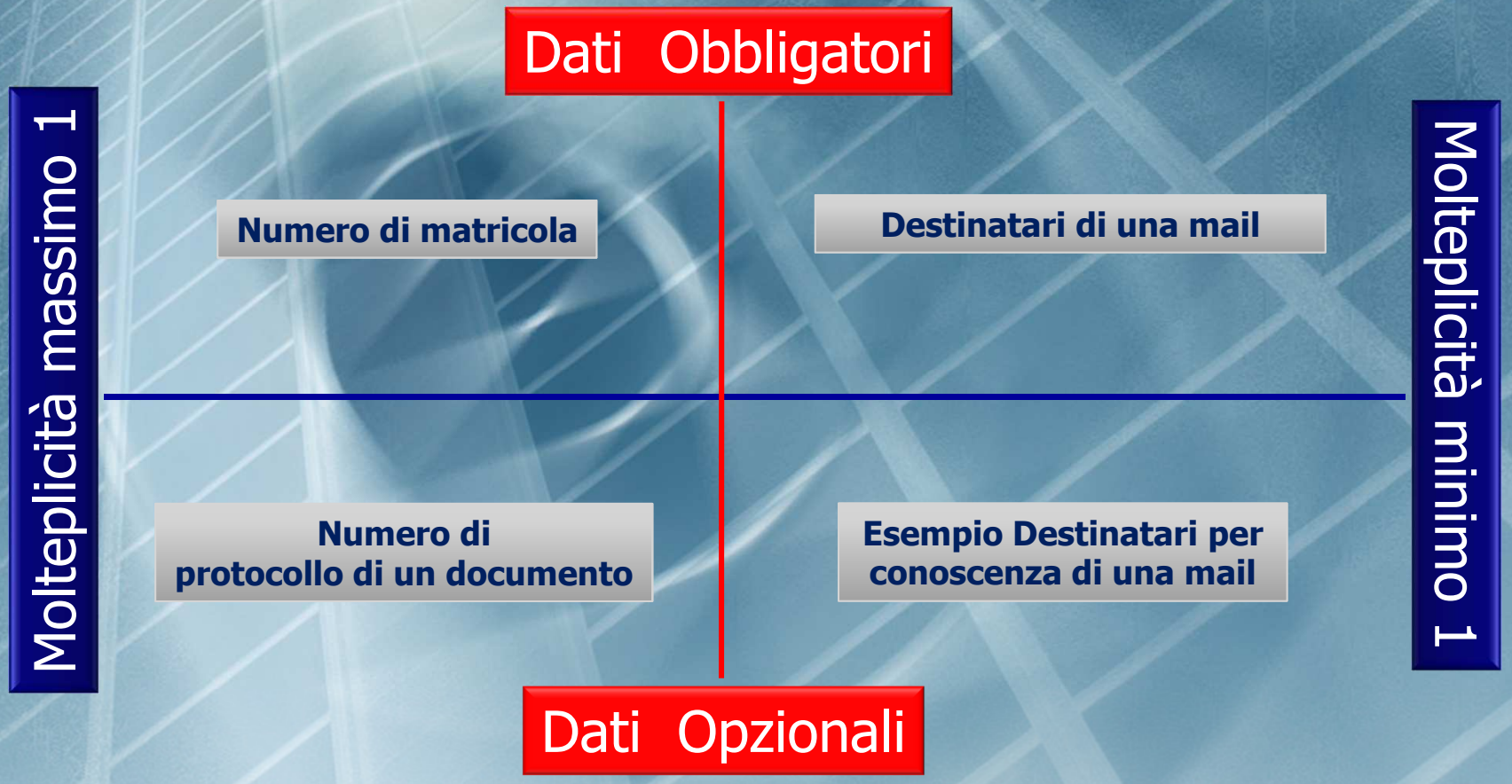
DTD: gli elementi



DTD: gli elementi e la sintassi



DTD: elementi e molteplicità



DTD: gli elementi (tabella)

SUFFISSI

Null	Obbligatotio
?	Opzionale (0 o una volta)
+	Uno o più volte
*	Zero o più volte

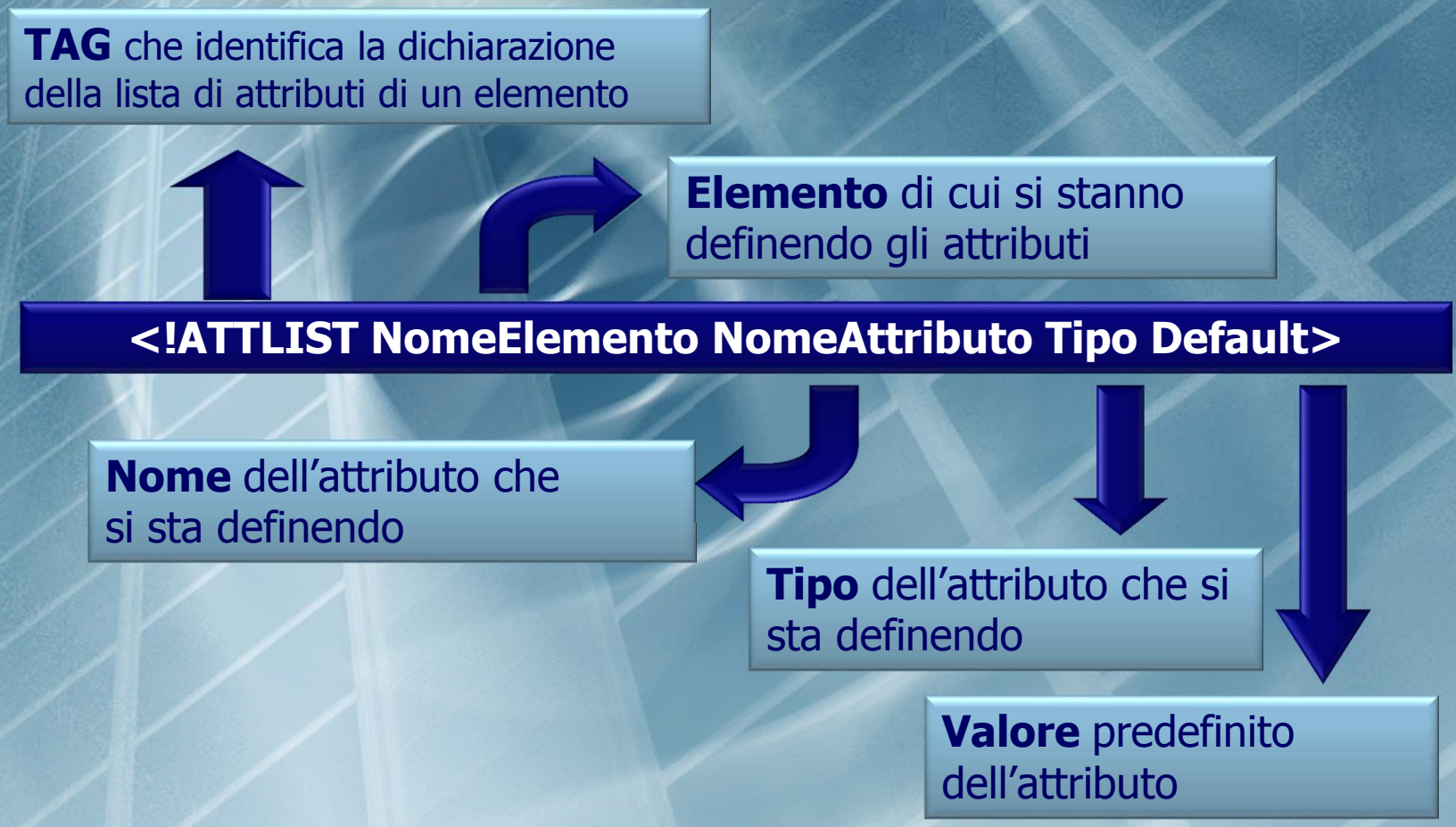
SEPARATORI

, (*virgola*)	tutti nell'ordine dell'elenco
| (*pipe*)	scelta (operatore OR esclusivo → XOR)

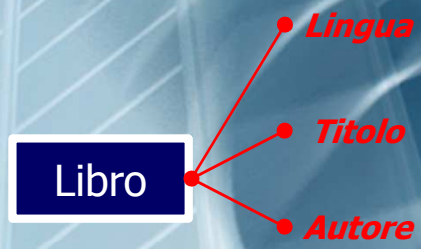
RAGGRUPPAMENTO

( )	elenco di elementi

DTD: attributi

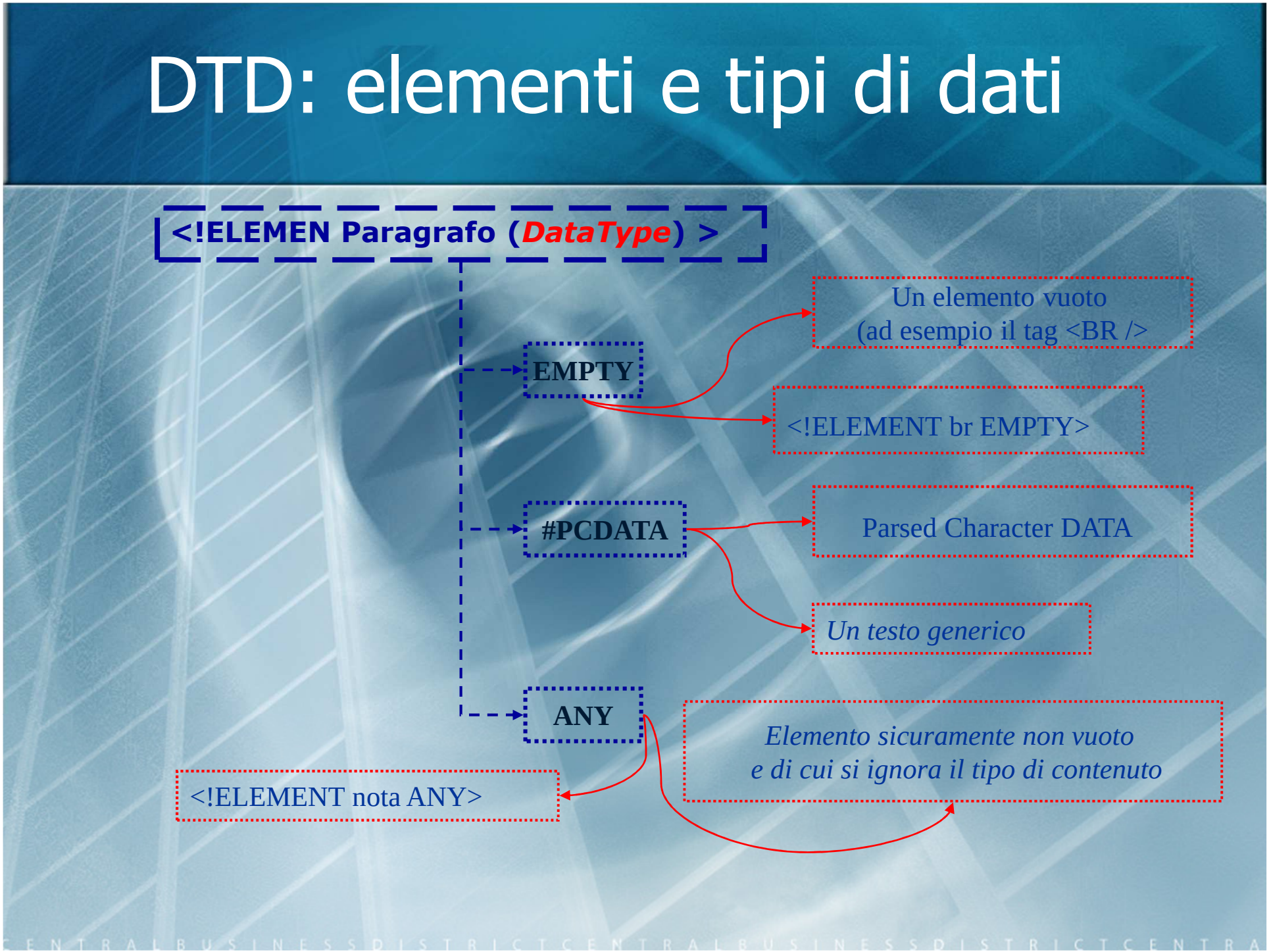


DTD: attributi, un esempio

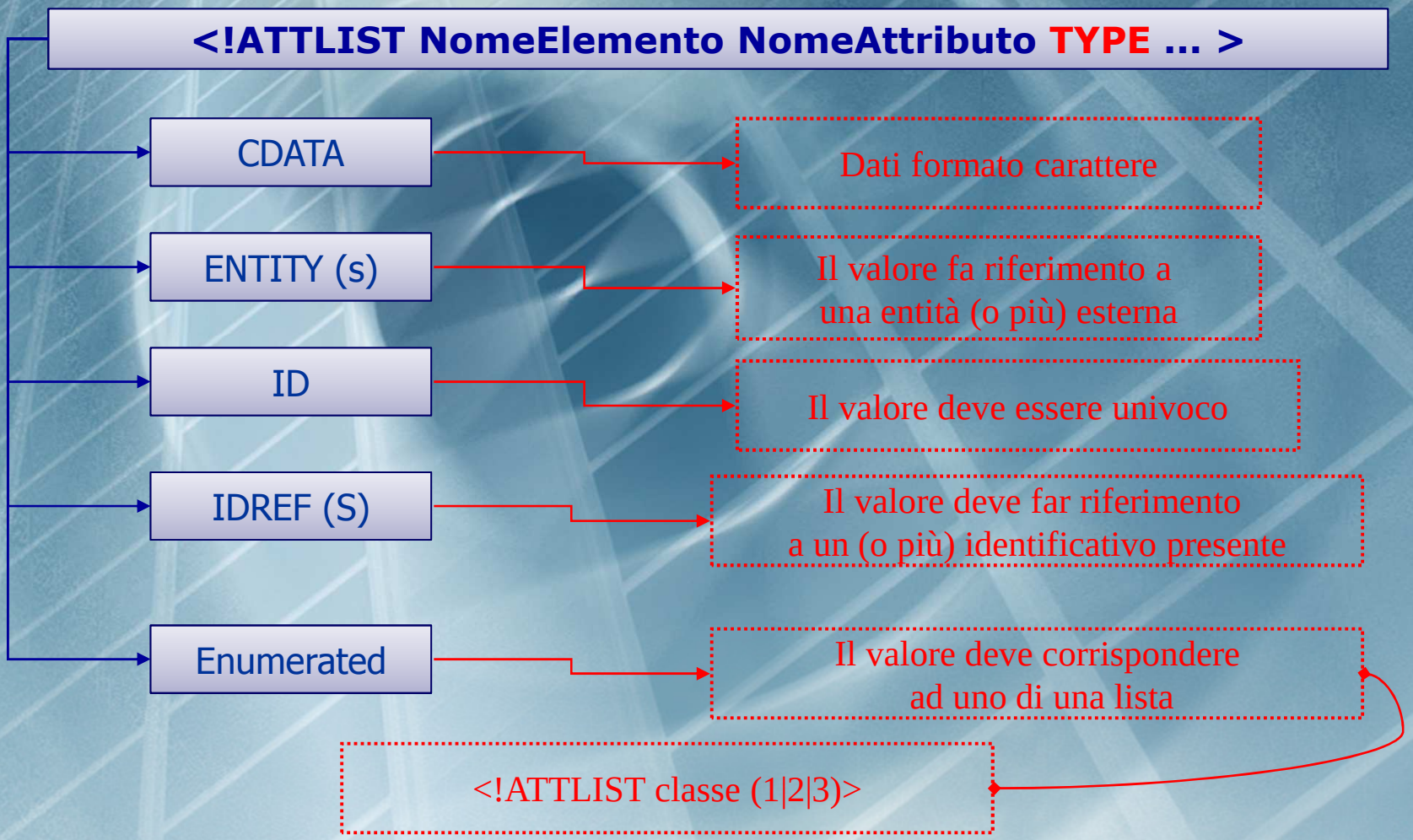



```
<!ELEMEN Libro (.....) >
   <!ATTLIST Libro Titolo CDATA #REQUIRED
                   Autore CDATA #REQUIRED
                   Lingua (Italiano|Inglese) 'Italiano' >
```

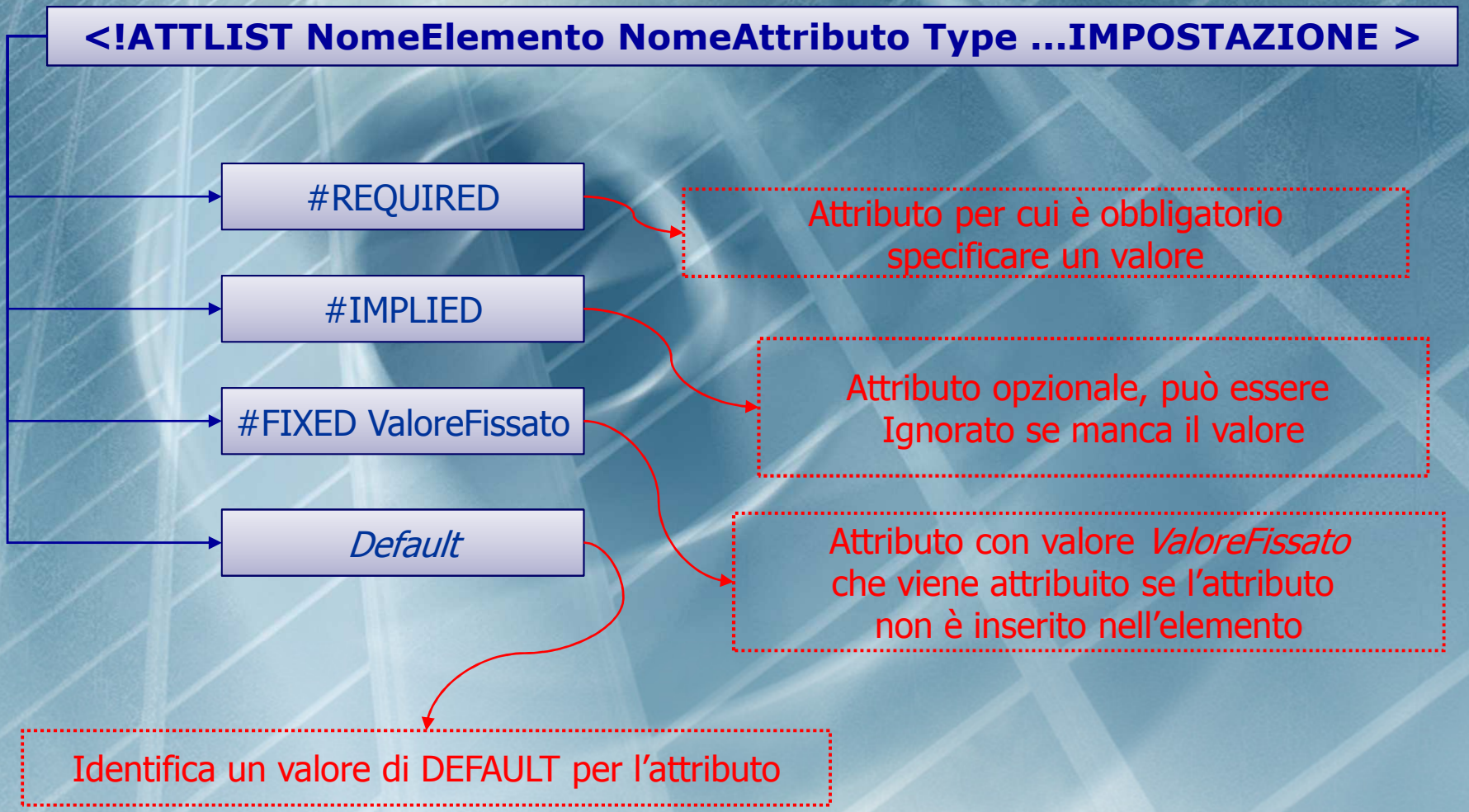
DTD: elementi e tipi di dati



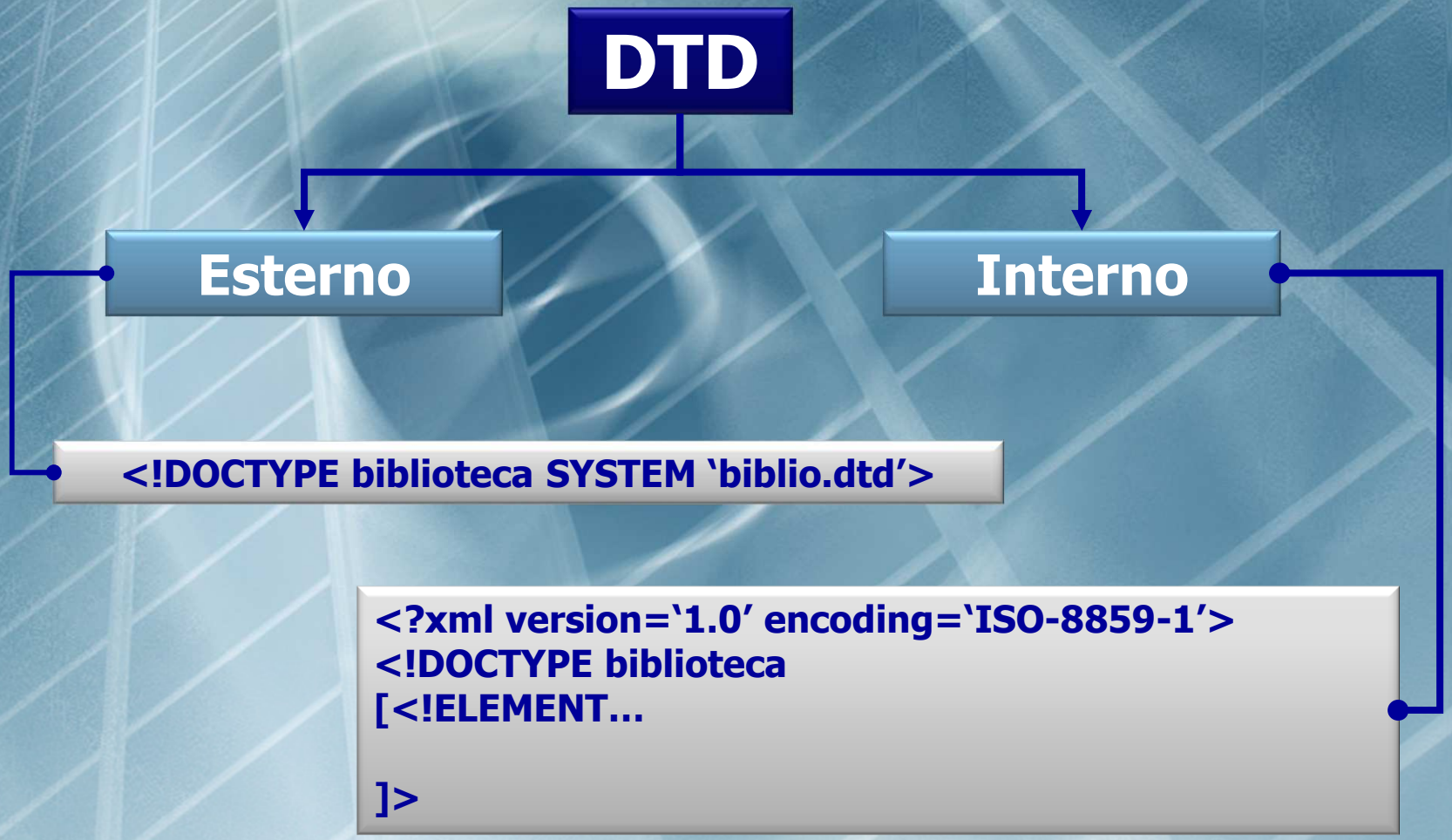
DTD: attributi e tipi di dati



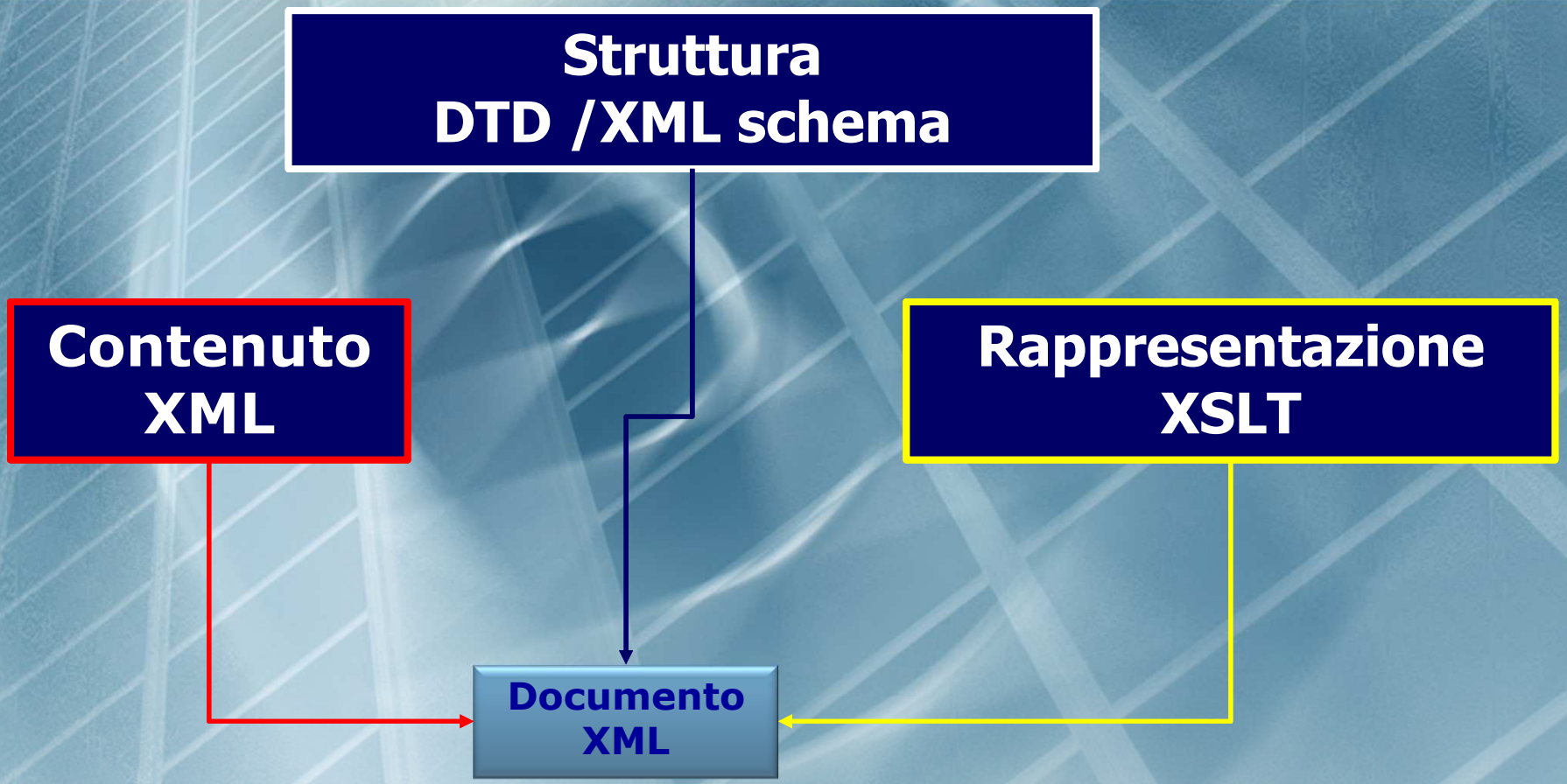
DTD: attributi e tipi di dati



DTD e file XML

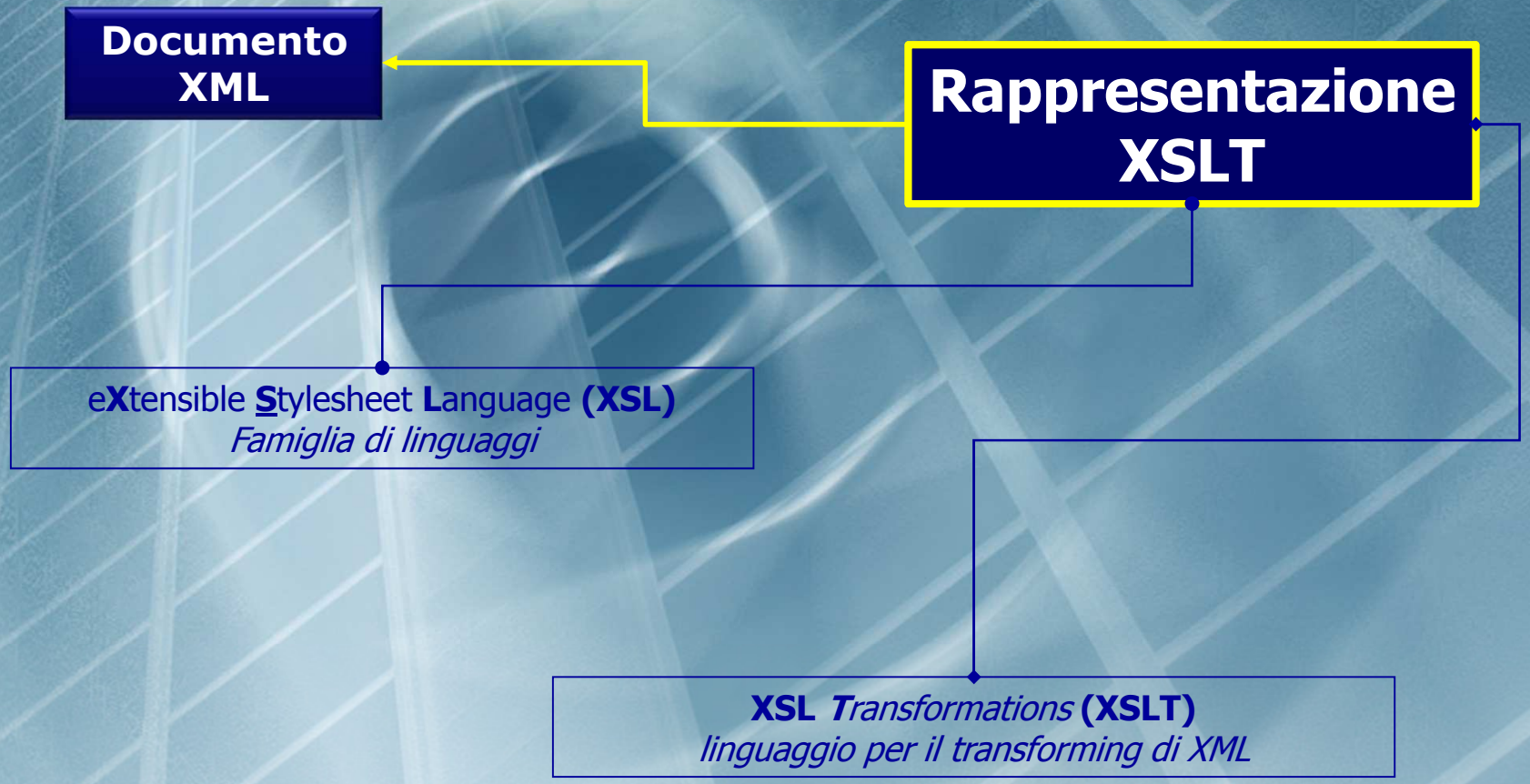


Documento XML :
Dati+Struttura+Rappresentazione



XSLT e XML: la rappresentazione

XSLT e XML: la rappresentazione



XSLT e XML: la rappresentazione

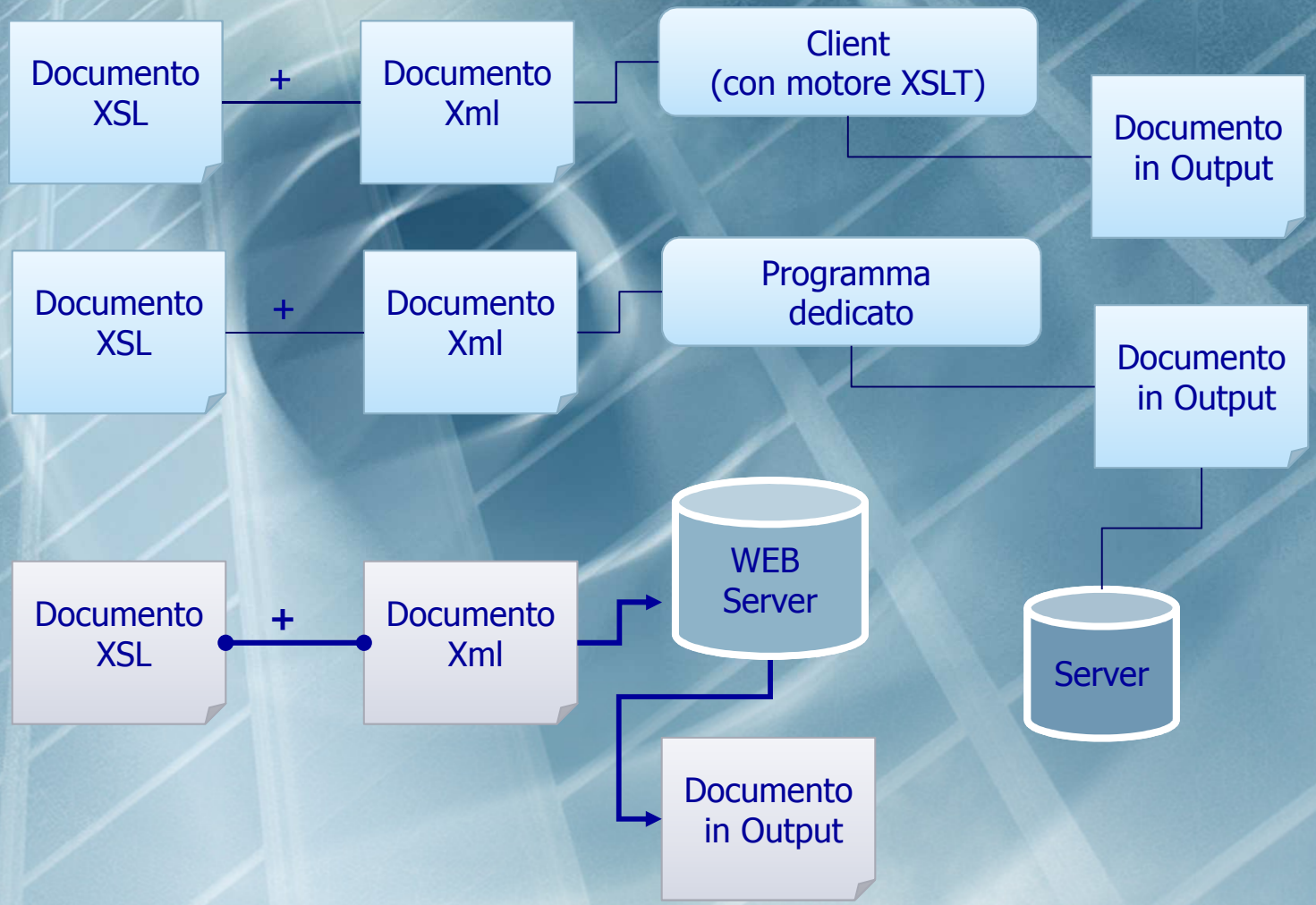
XSL Transformations (XSLT) è un linguaggio per il trasformazione

*Trasforma un documento scritto in XML in un documento rappresentato in un altro linguaggio (*ad esempio HTML*)*

XSLT e XML: il parser

Il **parsing** o **analisi sintattica** è il processo atto ad analizzare uno stream (flusso) continuo in input (letto per esempio da un file o una tastiera) in modo da determinare la sua struttura grammaticale grazie ad una data grammatica formale. Un **parser** è un programma che esegue questo compito. (*definizione da WIKIPEDIA*)

XST: il parsing



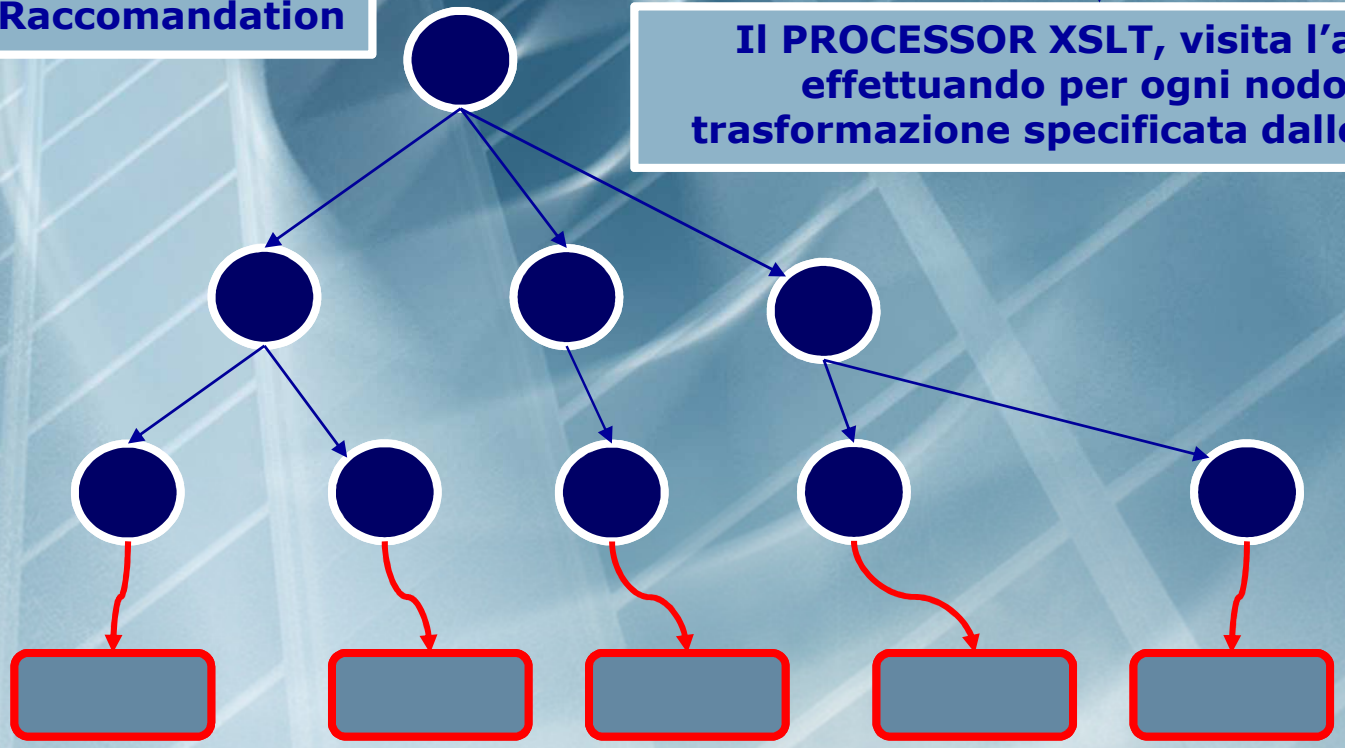
XSLT

XSLT

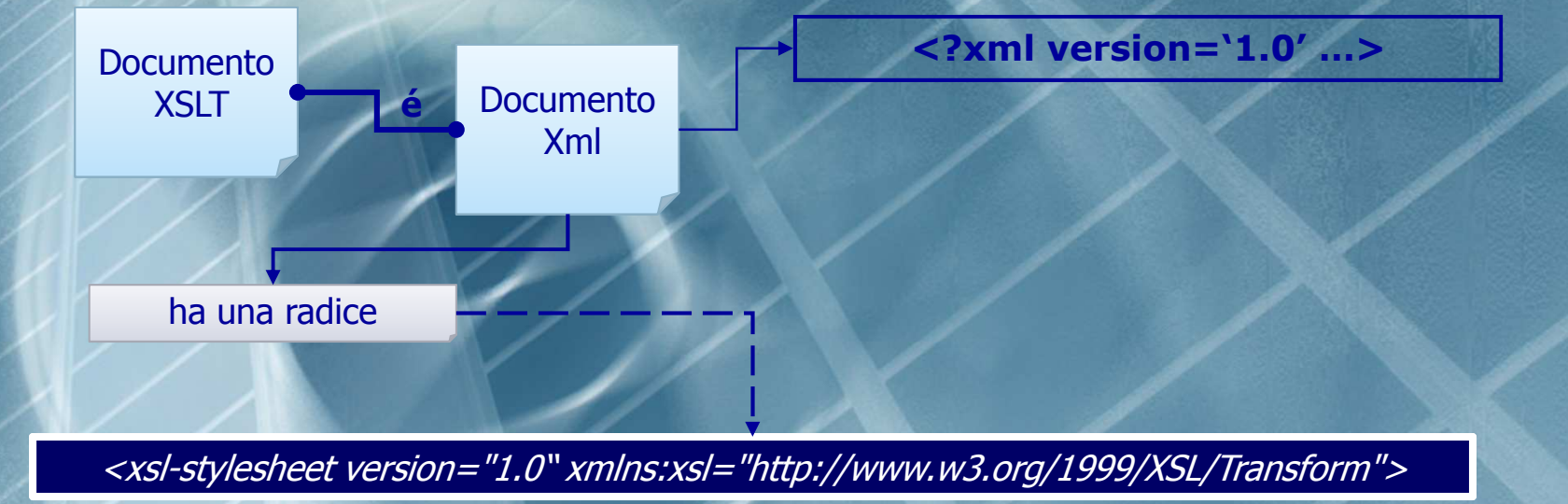
Specificare (in XML) REGOLE per la trasformazione XML→ DocumentType

1999
W3C Raccomandation

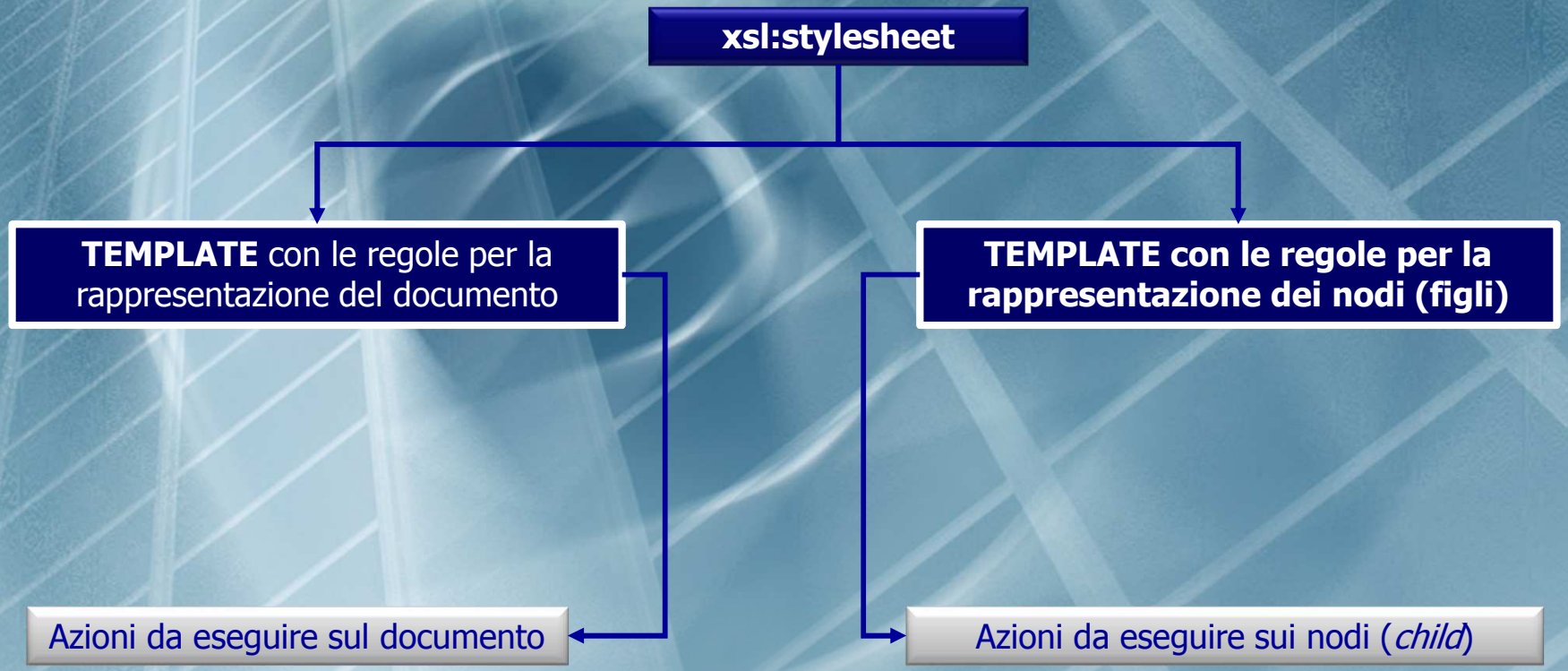
Il PROCESSOR XSLT, visita l'albero effettuando per ogni nodo la trasformazione specificata dalle regole



XSLT struttura di un documento



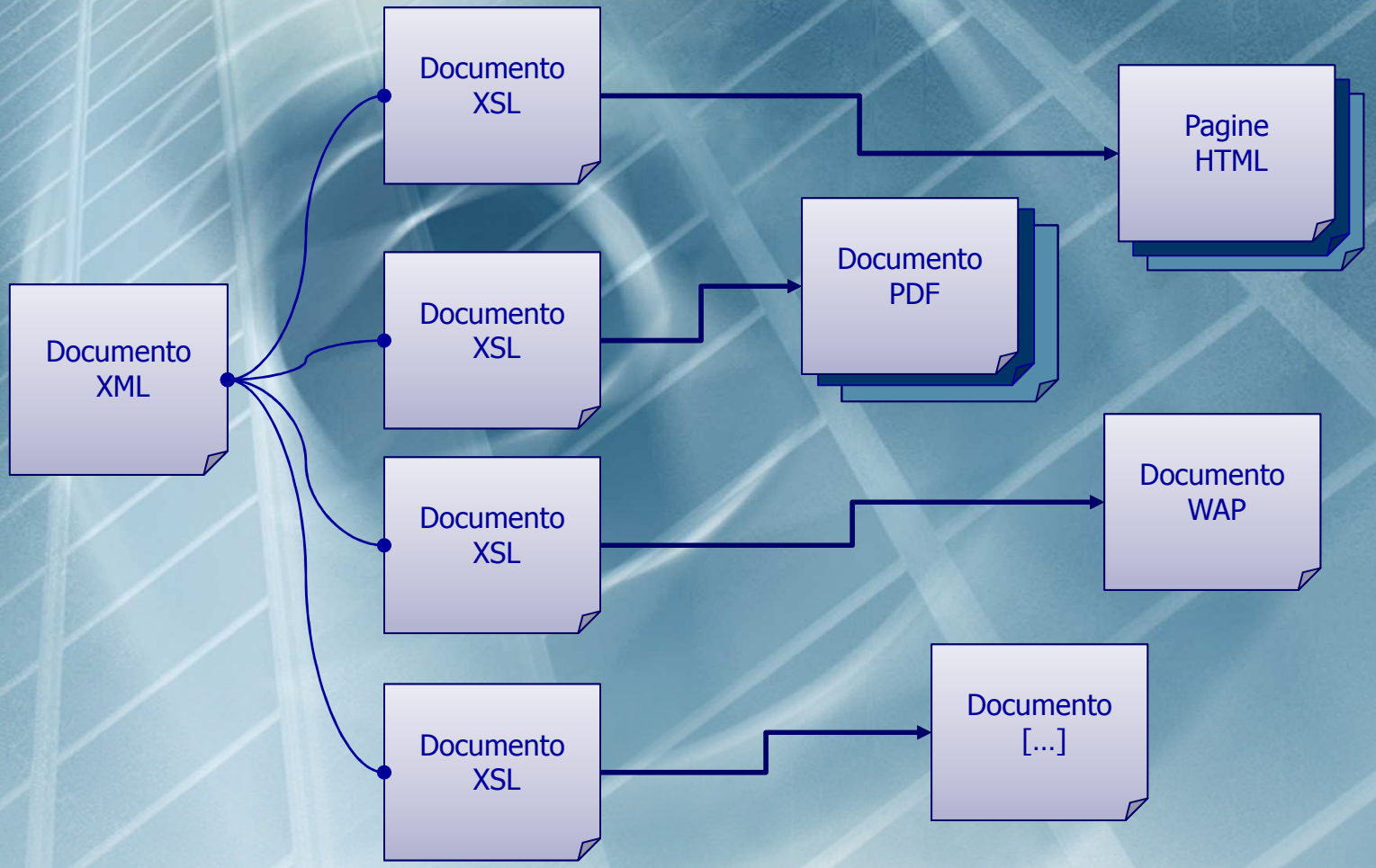
XSLT struttura di un documento



XSLT: un primo macro esempio

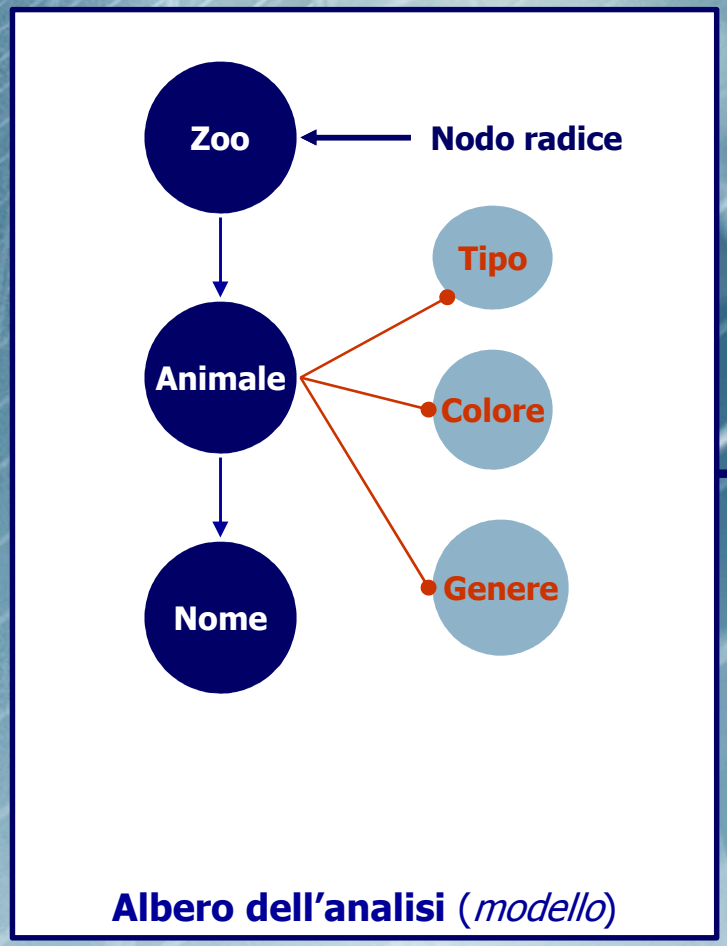
```
<?xml version="1.0"?>
<xsl:stylesheet>
        <xsl:template match="/">
                        [action]
        </xsl:template>
        <xsl:template match="BookCatalogue">
                        [action]
        </xsl:template>
        <xsl:template match="Book">
                        [action]
        </xsl:template>
        ...
</xsl:stylesheet>
```

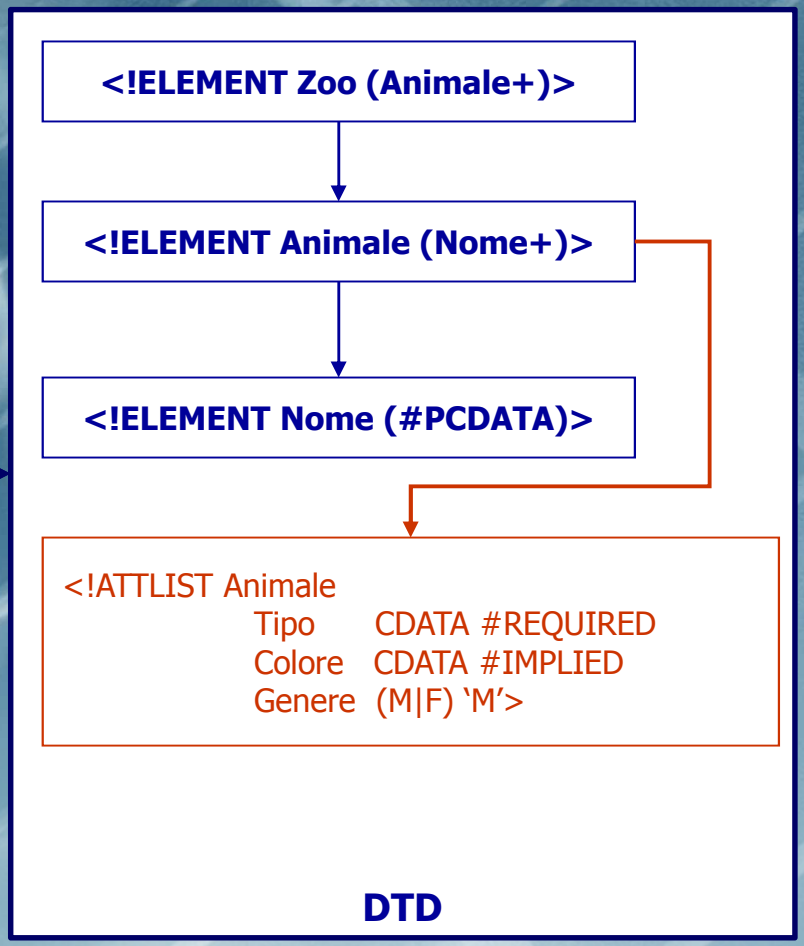
XSLT: schema riassuntivo



XSLT: esempio di trasformazione

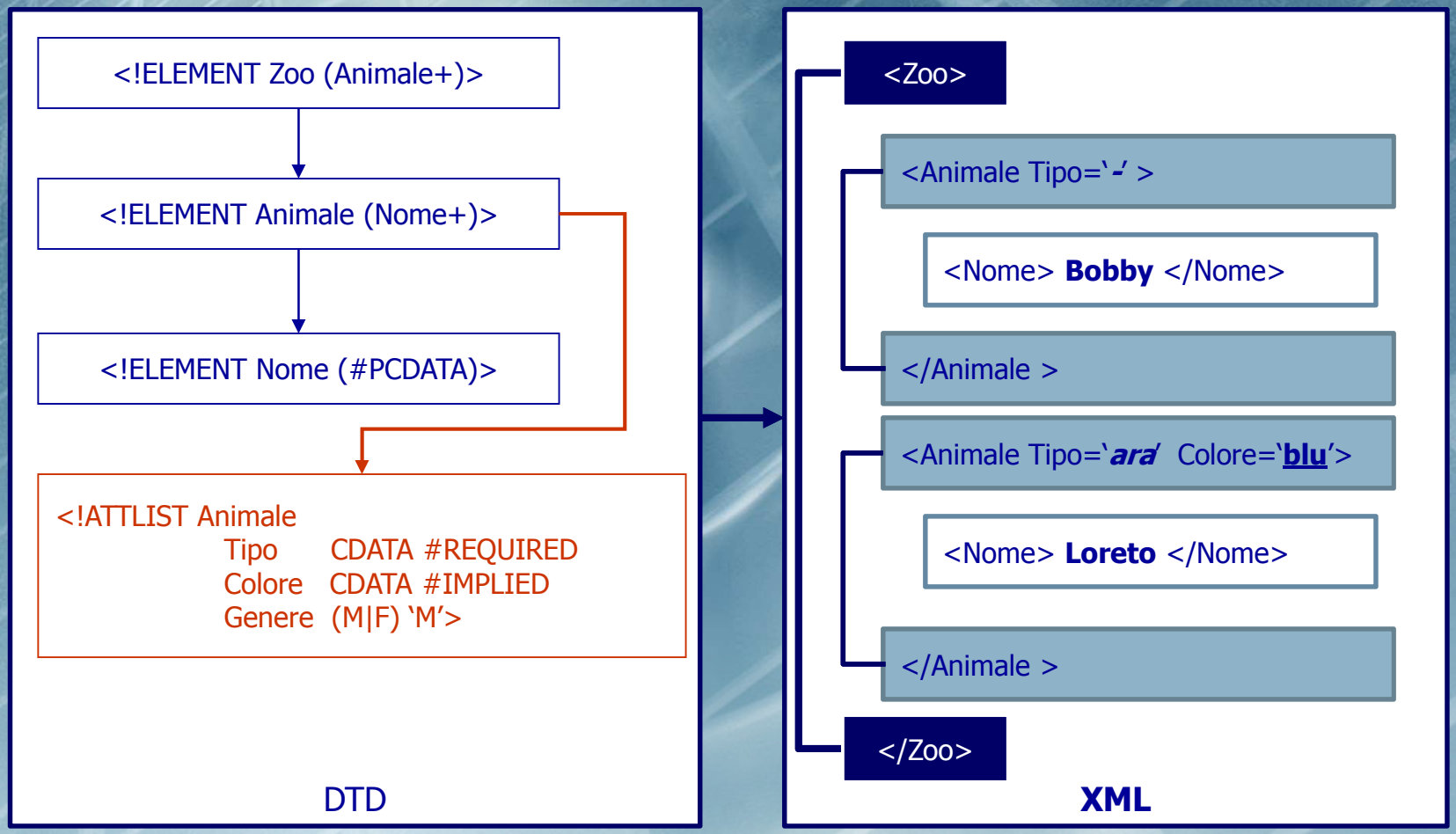
modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo





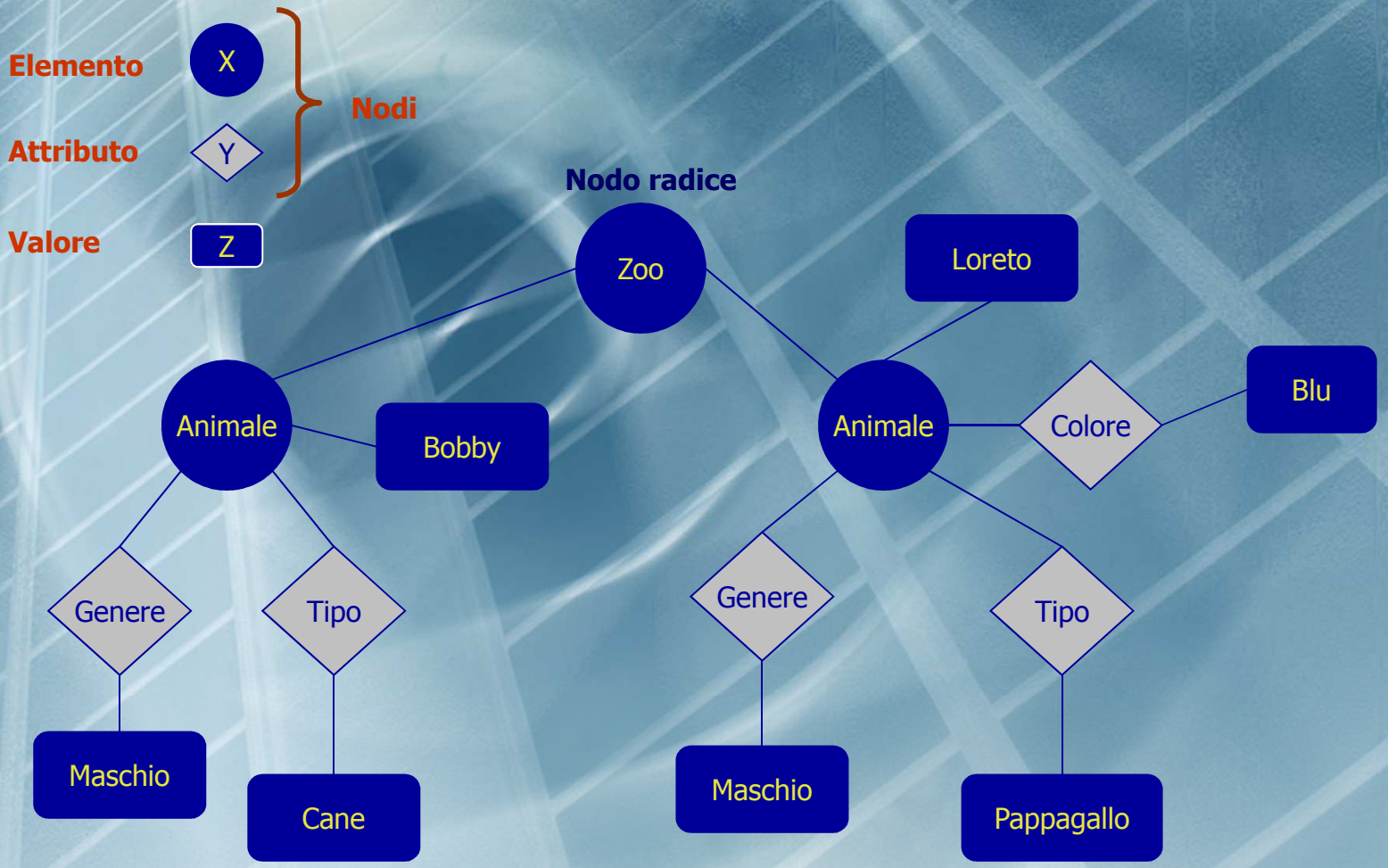
XSLT: esempio di trasformazione

modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo



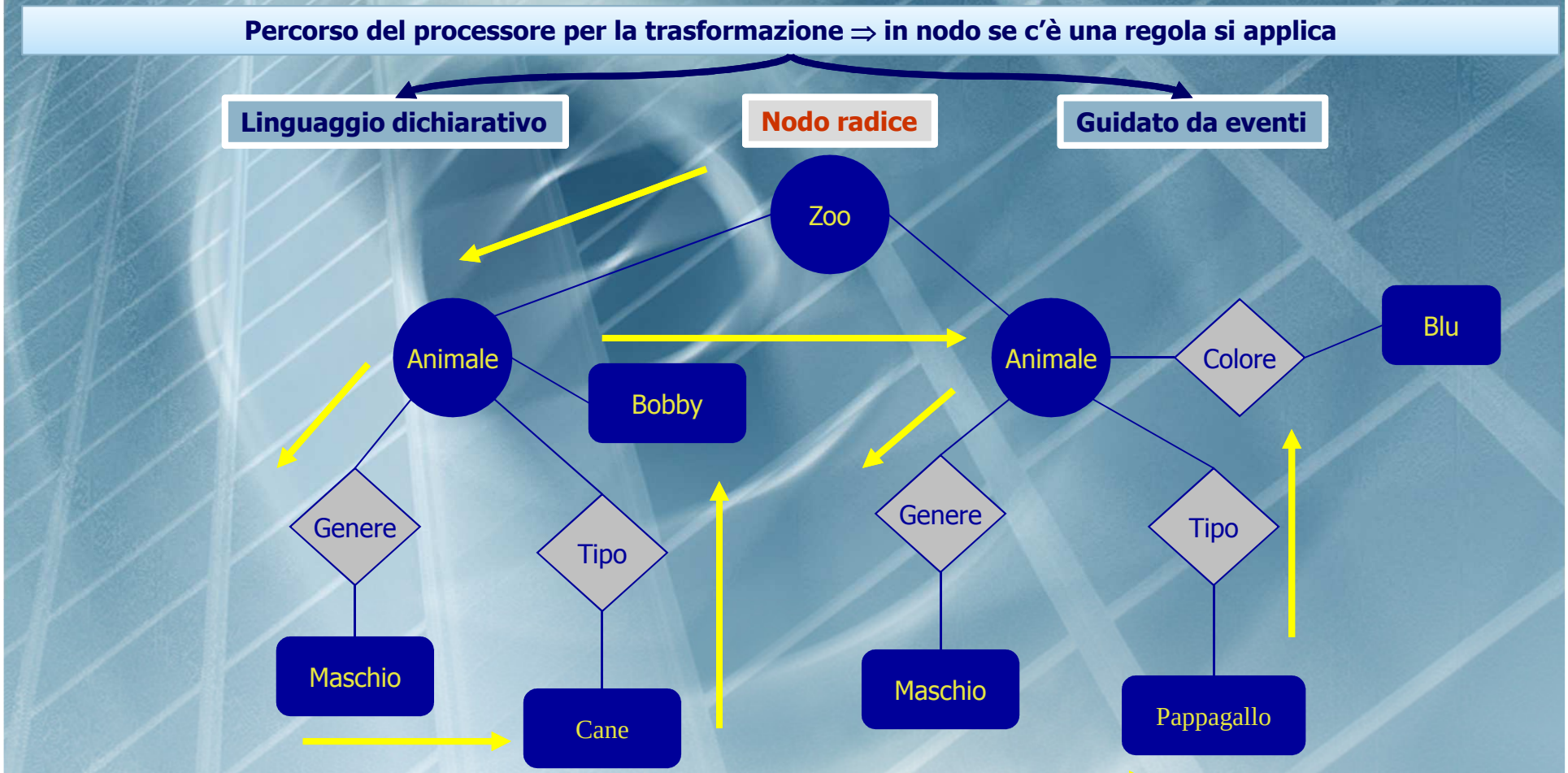
XSLT: esempio di trasformazione

modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo



XSLT: esempio di trasformazione

modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo



XSLT: esempio di trasformazione

modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<xsl:stylesheet version='1.0' xmlns:xsl='http://www.w3.org/1999/XSL/Transform'>
<xsl:output method='text' encoding='ISO-8859-1' omit-xml-declaration='yes' />

    <xsl:template match='/'>
            <xsl:applay-templates />
     </xsl:template>

    <xsl:template match='Animale'>
            <xsl:applay-templates />
    </xsl:template >

    <xsl:template match='Animale'>
       Allo zoo ho visto <xsl:value-of select='.'/> che è un <xsl:value-of select='@Tipo'/>
    <xsl:apply-templates select='@Colore' />
   </xsl:template>

<xsl:template match='@Colore'>
 che è di colore <xsl:value-of select='.'/>
</xsl:template>

</xsl:stylesheet>
```

XSLT: esempio di trasformazione

modello ⇒ DTD ⇒ XML ⇒ XSLT ⇒ Testo

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<xsl:stylesheet version='1.0' xmlns:xsl='http://www.w3.org/1999/XSL/Transform'>
<xsl:output method='html' encoding='ISO-8859-1' omit-xml-declaration='yes' />
        <xsl:template match='/'>
                <html>
                        <head>
                          <title>From XML To HTML</title>
                        </head>
                        <body>
                                <xsl:apply-templates />
                        </body>
                </html>
</xsl:template>

<xsl:template match='Animale'>
        <xsl:apply-templates />
</xsl:template >

<xsl:template match='Animale'>
        <P>Allo zoo ho visto <xsl:value-of select='.'/> che è un <xsl:value-of select='@Tipo'/></P>
<xsl:apply-templates select='@Colore' />
</xsl:template>

<xsl:template match='@Colore'>
 che è di colore <B> <xsl:value-of select='.' /> </B>
</xsl:template>

</xsl:stylesheet>
```

«Il test di un programma può essere usato per mostrare la presenza di bug, ma mai per mostrare la loro assenza.»

Edsger Wybe Dijkstra