

# IEC 61850

---

## [M3] Sesión práctica: Análisis de mensajes IEC 61850

Yasmina Galve Pastor – [ygalve@fcirce.es](mailto:ygalve@fcirce.es)  
Noviembre de 2017

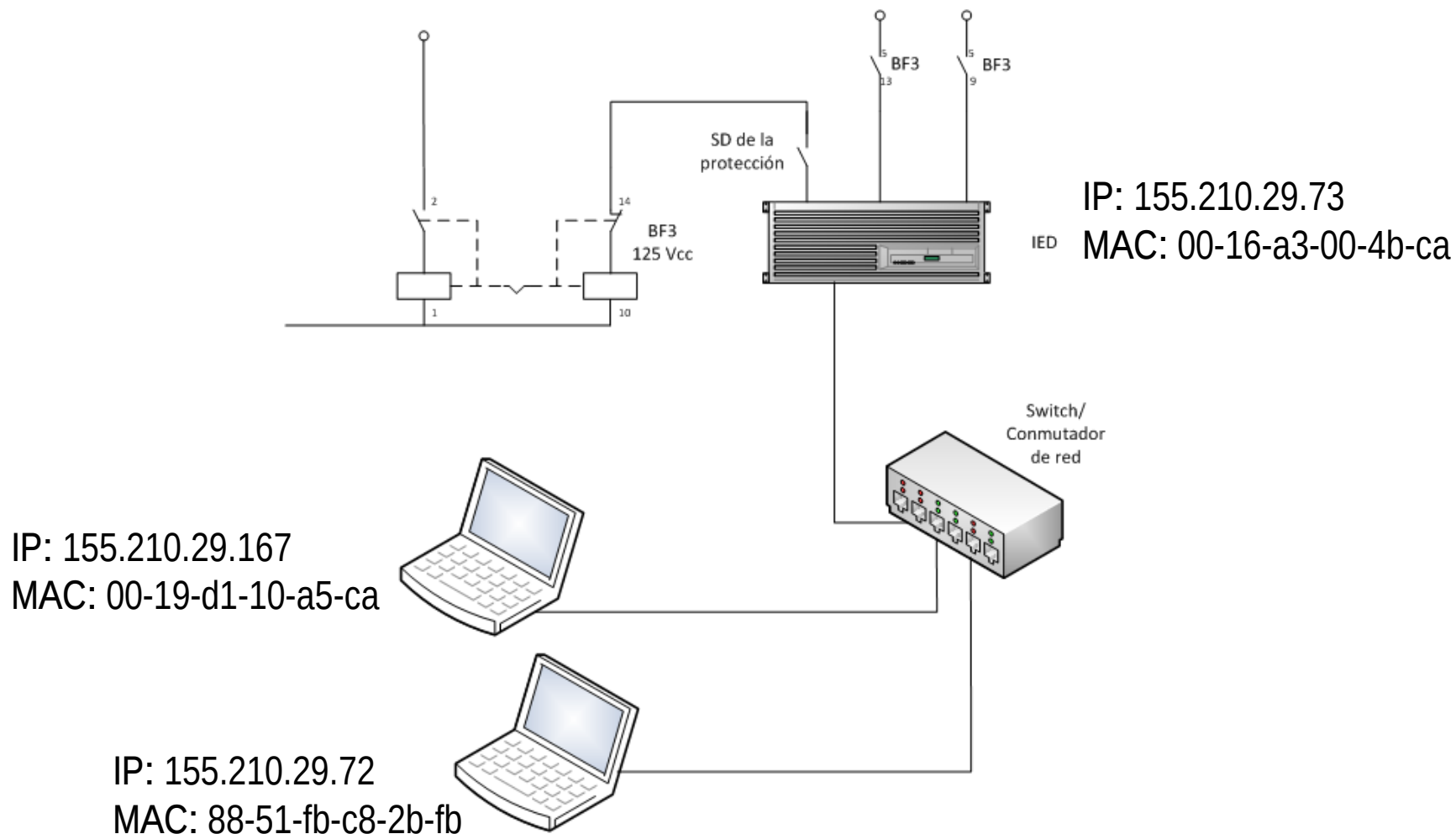
# Estructura de la clase

---

1. Escenario de pruebas
2. Codificación BER y notación ASN.1
3. Análisis de mensajes GOOSE
4. Análisis de mensajes MMS
5. Análisis de mensajes SV

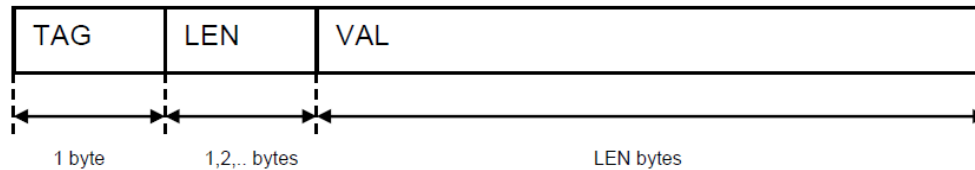


# 1. Escenario de pruebas



## 2. Codificación BER y notación ASN.1

- Codificación BER



- Notación abstracta ASN.1

```
Read-request ::= SEQUENCE
{
    specificationWithResult    [0] IMPLICIT BOOLEAN DEFAULT FALSE
    variableAccessSpecification [1] VariableAccessSpecification
}
```



## 2. Codificación BER y notación ASN.1

---

### CODIFICACIÓN BER

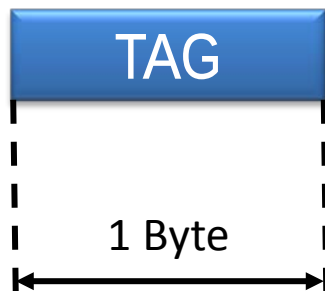
Reglas básicas de codificación de datos de nivel de aplicación.



## 2. Codificación BER y notación ASN.1

---

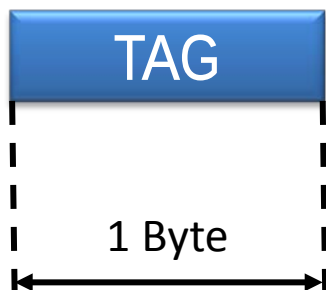
### TAG



- Bits 7 y 6: Tipo de etiqueta
- Bit 5: Primitivo o construido
- Bits 4-0: Valor del TAG

## 2. Codificación BER y notación ASN.1

### TAG

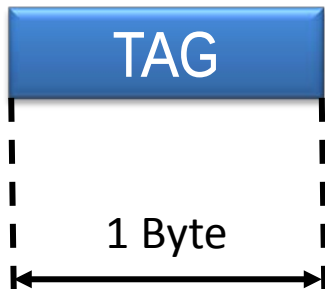


| Bit 7 | Bit 6 | Descripción             |
|-------|-------|-------------------------|
| 0     | 0     | Universal               |
| 0     | 1     | Propio de la aplicación |
| 1     | 0     | Específico del contexto |
| 1     | 1     | Privado                 |

- Bits 7 y 6: Tipo de etiqueta
- Bit 5: Primitivo o construido
- Bits 4-0: Valor del TAG

## 2. Codificación BER y notación ASN.1

### TAG



- Bits 7 y 6: Tipo de etiqueta
- Bit 5: Primitivo o construido
- Bits 4-0: Valor del TAG

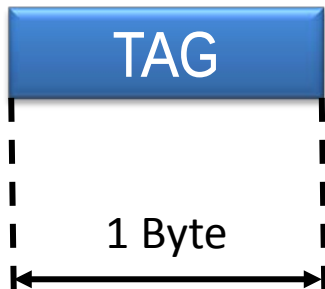
### Valores típicos para MMS

| Bit 7 | Bit 6 | Bit 5 | Descripción                        |
|-------|-------|-------|------------------------------------|
| 0     | 0     | 0     | Universal primitivo                |
| 0     | 0     | 1     | Universal construido               |
| 1     | 0     | 0     | Específico del contexto primitivo  |
| 1     | 0     | 1     | Específico del contexto construido |



## 2. Codificación BER y notación ASN.1

TAG

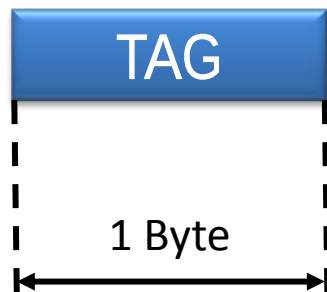


### Primitivas universales

| Tipo              | Etiqueta |
|-------------------|----------|
| BOOLEAN           | 01       |
| INTEGER           | 02       |
| BITSTRING         | 03       |
| OCTETSTRING       | 04       |
| NULL              | 05       |
| OBJECT IDENTIFIER | 06       |
| SEQUENCE          | 10       |
| IA5STRING         | 16       |
| UTCTIME           | 17       |
| GENERALIZETIME    | 18       |
| VISIBLESTRING     | 1A       |

## 2. Codificación BER y notación ASN.1

### TAG



### Construidos universales

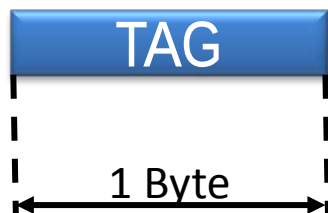
| Tipo        | Etiqueta |
|-------------|----------|
| SEQUENCE    | 16       |
| SEQUENCE OF | 16       |
| SET         | 17       |
| SET OF      | 17       |
| CHOICE      | -        |
| SELECTION   | -        |

| Bit 7 | Bit 6 | Bit 5 | Descripción                        |
|-------|-------|-------|------------------------------------|
| 0     | 0     | 0     | Universal primitivo                |
| 0     | 0     | 1     | Universal construido               |
| 1     | 0     | 0     | Específico del contexto primitivo  |
| 1     | 0     | 1     | Específico del contexto construido |



## 2. Codificación BER y notación ASN.1

### TAG



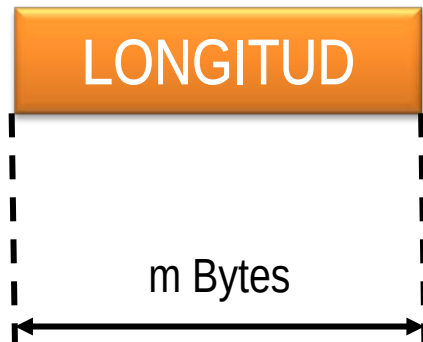
```
Data ::= CHOICE {  
  -- context tag 0 is reserved for AccessResult  
  IF ( str1 )  
    array [1] IMPLICIT SEQUENCE OF Data,  
  ELSE  
    array [1] IMPLICIT NULL,  
  ENDIF  
  IF ( str2 )  
    structure [2] IMPLICIT SEQUENCE OF Data,  
  ELSE  
    structure [2] IMPLICIT NULL,  
  ENDIF  
  boolean [3] IMPLICIT BOOLEAN,  
  bit-string [4] IMPLICIT BIT STRING,  
  integer [5] IMPLICIT INTEGER,  
  unsigned [6] IMPLICIT INTEGER, -- shall not be negative  
  floating-point [7] IMPLICIT FloatingPoint,  
  -- [8] is reserved  
  octet-string [9] IMPLICIT OCTET STRING,  
  visible-string [10] IMPLICIT VisibleString,  
  generalized-time [11] IMPLICIT GeneralizedTime,  
  binary-time [12] IMPLICIT TimeOfDay,  
  bcd [13] IMPLICIT INTEGER, -- shall not be negative  
  booleanArray [14] IMPLICIT BIT STRING,  
  objId [15] IMPLICIT OBJECT IDENTIFIER,  
  ...,  
  mMSString [16] IMPLICIT MMSString  
}
```

### Primitivas específicas del contexto para MMS

| Tipo              | Etiqueta (Hex) |
|-------------------|----------------|
| Array             | 81             |
| Estructura        | 82             |
| Booleano          | 83             |
| Bitstring         | 84             |
| Entero            | 85             |
| Entero sin signo  | 86             |
| Real              | 87             |
| Cadena de octetos | 89             |
| Visiblestring     | 8a             |
| Generalized time  | 8b             |

## 2. Codificación BER y notación ASN.1

### LONGITUD



- Longitud  $< 128 \rightarrow m = 1 \text{ Byte} = \text{longitud}$

02 01 10  $\rightarrow$  INTEGER(0x02) de 1 byte(0x01) y valor 16(0x10)

- Longitud  $\geq 128$ :
  - $8x \rightarrow x = n^{\circ} \text{ bytes que codifican la longitud} \rightarrow m = 1 + x$

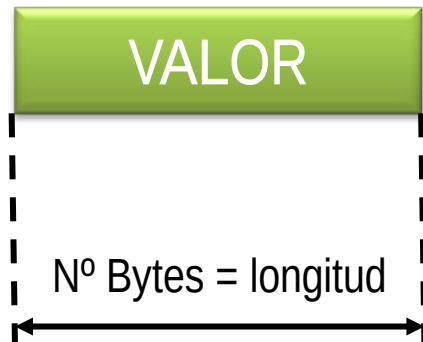
02 82 01 00 xxx...xxx  $\rightarrow$  INTEGER(0X02), cuya longitud está codificada en 2 bytes(82) y es igual a 256 (0X0100). El valor del entero es xxx...xxx (de longitud 256)

## 2. Codificación BER y notación ASN.1

---

### VALOR

Depende de la notación ASN.1 utilizada.



- Sin etiqueta: codificación mediante etiquetas universales.
- IMPLICIT: se sustituye la etiqueta existente por la indicada en “[ ]”.
- EXPLICIT: la etiqueta indicada en “[ ]” se añade a la existente.

## 2. Codificación BER y notación ASN.1

---

EJEMPLOS → Codificación de un entero de valor 3

- INTEGER → Primitiva universal  
02 01 03
- [5] IMPLICIT INTEGER → MMS define su propio conjunto de etiquetas para codificar sus datos.  
85 01 03
- [5] EXPLICIT INTEGER → MMS define su propio conjunto de etiquetas para codificar sus datos.  
a5 03 02 01 03



## 2. Codificación BER y notación ASN.1

---

Caso especial: Bitstring, además de TAG + LONGITUD + VALOR, también codifica los bits que desecha.

- BITSTRING → Primitiva universal

03 03 05 FF 00 → Bitstring(03) de longitud 3 (03), que desecha los últimos 5 bits(05) y valor  
1111 1111 0000-0000

- [4] IMPLICIT BITSTRING → Primitiva específica del contexto

84 03 05 FF 00

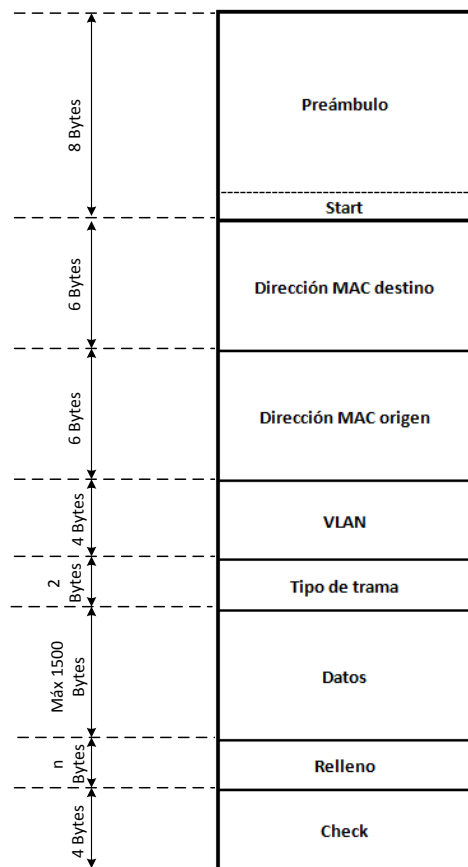


# Análisis de mensajes GOOSE

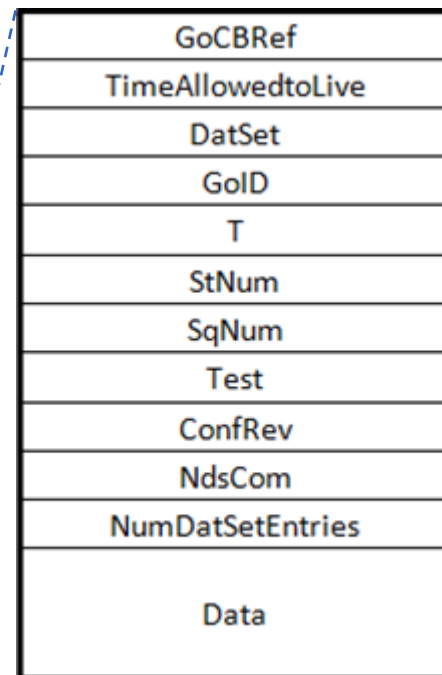


### 3. Análisis GOOSE

#### MENSAJE GOOSE



|             |                                             |
|-------------|---------------------------------------------|
| MAC destino | 01-0C-CD-01-00-00<br>→<br>01-0C-CD-01-01-FF |
| Ethertype   | 0x88B8                                      |
| AppID       | 0x0000 – 0x3FFF                             |



### 3. Análisis GOOSE

## MENSAJE GOOSE

|                   |
|-------------------|
| GoCBRef           |
| TimeAllowedtoLive |
| DatSet            |
| GoID              |
| T                 |
| StNum             |
| SqNum             |
| Test              |
| ConfRev           |
| NdsCom            |
| NumDatSetEntries  |
| Data              |

```
IECGoosePdu ::= SEQUENCE {  
    gocbRef          [0]  IMPLICIT VISIBLE-STRING,  
    timeAllowedtoLive [1]  IMPLICIT INTEGER,  
    datSet           [2]  IMPLICIT VISIBLE-STRING,  
    goID             [3]  IMPLICIT VISIBLE-STRING OPTIONAL,  
    t                [4]  IMPLICIT UtcTime,  
    stNum            [5]  IMPLICIT INTEGER,  
    sqNum            [6]  IMPLICIT INTEGER,  
    test             [7]  IMPLICIT BOOLEAN DEFAULT FALSE,  
    confRev          [8]  IMPLICIT INTEGER,  
    ndsCom           [9]  IMPLICIT BOOLEAN DEFAULT FALSE,  
    numDatSetEntries [10] IMPLICIT INTEGER,  
    allData          [11] IMPLICIT SEQUENCE OF Data,  
    security          [12] ANY OPTIONAL,  
    -- reserved for digital signature  
}
```



# Análisis de mensajes MMS

## 4. Análisis MMS

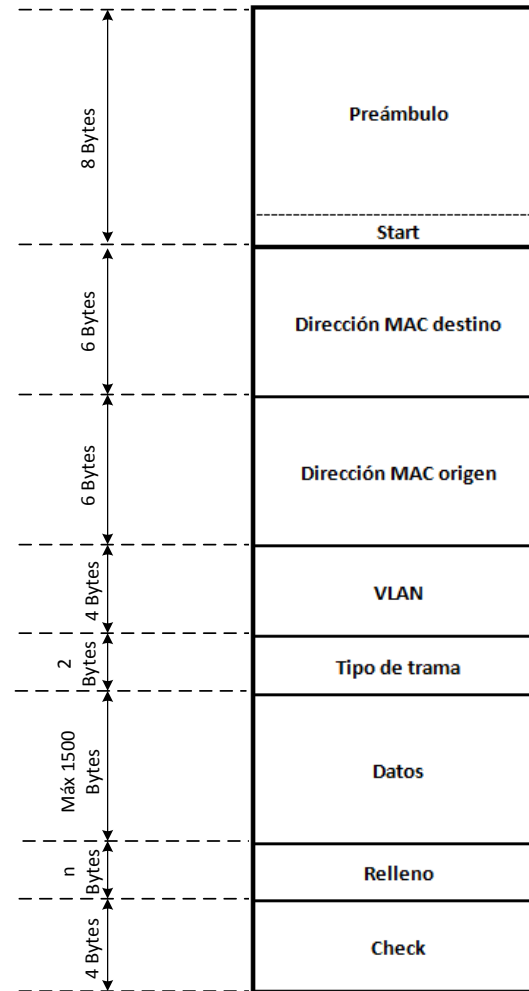
### PILA DE PROTOCOLOS MMS

|                       |                |                                                       |
|-----------------------|----------------|-------------------------------------------------------|
| Nivel de aplicación   |                | MMS (ISO 9506-2)                                      |
|                       |                | ACSE (ISO 8650)                                       |
| Nivel de Presentación | Presentación   | Presentación con conexión (ISO 8823-1)                |
|                       | Especificación | ASN.1 (ISO/IEC 8824-1)                                |
|                       | Codificación   | BER (ISO/IEC 8825-1)                                  |
| Sesión                |                | Sesión con conexión ISO/IEC 8327-1                    |
| Nivel de Transporte   | ISO            | Transporte sin conexión TP0. ISO/IEC 8073             |
|                       | Adaptación     | RFC-1006                                              |
|                       | TCP            | ICMP, TCP (RFC-792, RFC-793)                          |
| Nivel de red          |                | IP (RFC-791)                                          |
| Nivel de Enlace       |                | RFC-894, RFC-826 (ARP)                                |
| Nivel Físico          |                | Ethernet ISO/IEC 8802-3<br>10Base-T/100-BaseT (cobre) |
|                       |                | Ethernet ISO/IEC 8802-3<br>100-BaseFX (fibra)         |



## 4. Análisis de mensajes MMS

### Trama ETHERNET



## 4. Análisis de mensajes MMS

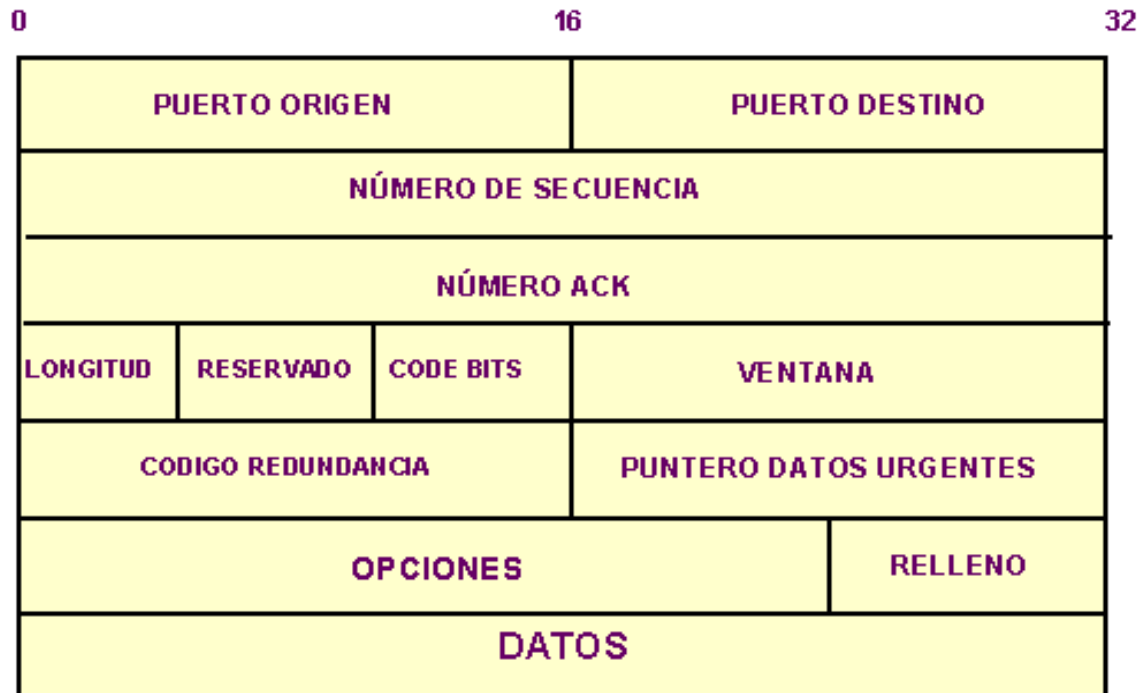
### Datagrama IP



## 4. Análisis de mensajes MMS

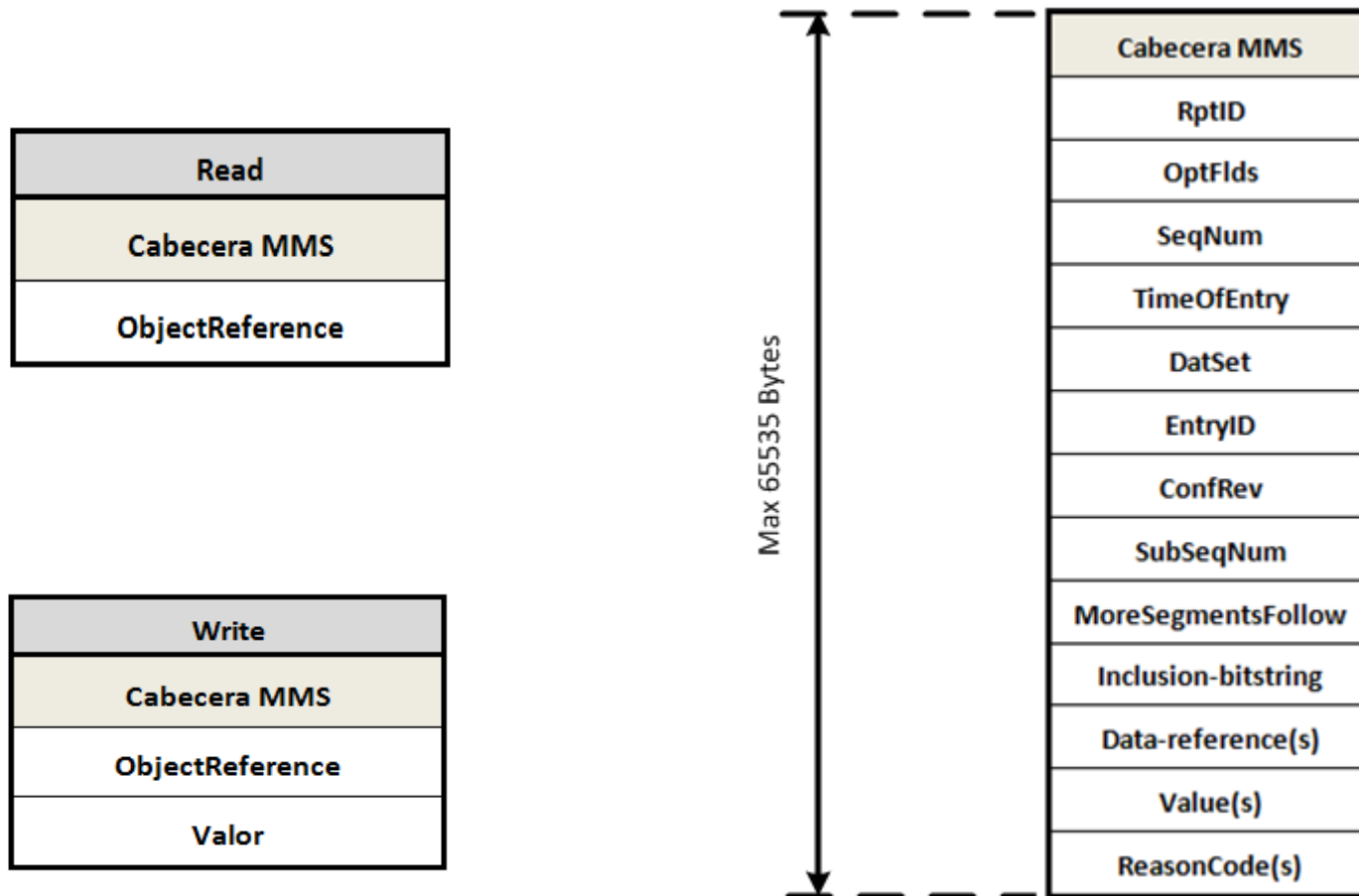
---

### Trama TCP



## 4. Análisis de mensajes MMS

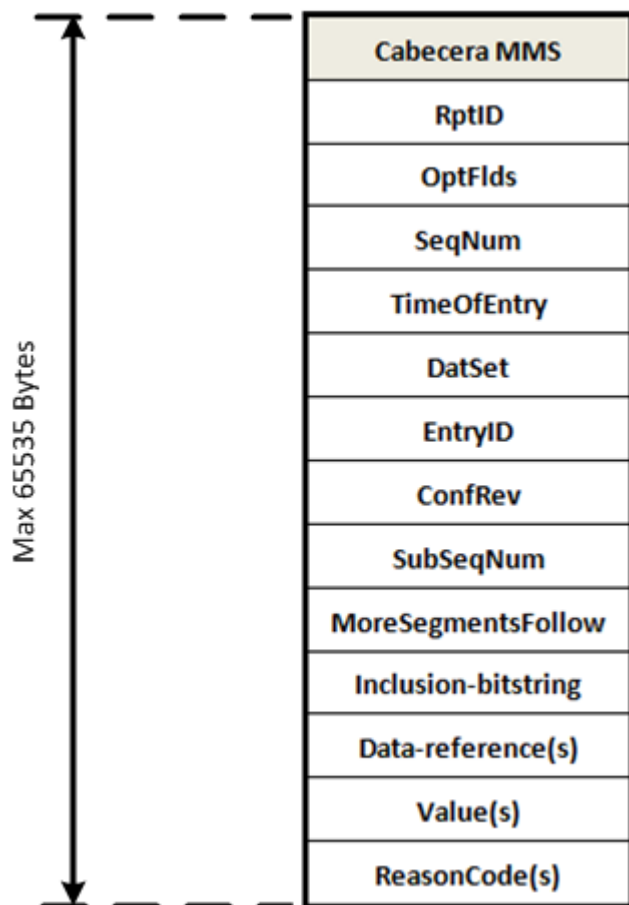
### MENSAJES MMS





## 4. Análisis de mensajes MMS

### OPTION FIELDS



| Option Fields |                      |
|---------------|----------------------|
| Bit           | Significado          |
| 0             | Reservado            |
| 1             | Sequence-number      |
| 2             | Report-timestamp     |
| 3             | Reason-for-inclusion |
| 4             | Data-set-name        |
| 5             | Data-reference       |
| 6             | Buffer-overflow      |
| 7             | entryID              |
| 8             | Conf-revision        |
| 9             | segmentation         |

| Reason for inclusion |                       |
|----------------------|-----------------------|
| Bit                  | Significado           |
| 0                    | Reservado             |
| 1                    | Data-change           |
| 2                    | Quality-change        |
| 3                    | Data-update           |
| 4                    | Integrity             |
| 5                    | General-interrogation |



## 4. Análisis de mensajes MMS

### Tipo de PDU

```

MMSpdu ::= CHOICE {
    confirmed-RequestPDU      [0] IMPLICIT Confirmed-RequestPDU,
    confirmed-ResponsePDU    [1] IMPLICIT Confirmed-ResponsePDU,
    confirmed-ErrorPDU       [2] IMPLICIT Confirmed-ErrorPDU,
    IF ( unsolicitedStatus informationReport eventNotification )
        unconfirmed-PDU      [3] IMPLICIT Unconfirmed-PDU,
    ELSE
        unconfirmed-PDU      [3] IMPLICIT NULL,
    ENDIF
    rejectPDU                [4] IMPLICIT RejectPDU,
    IF (cancel)
        cancel-RequestPDU    [5] IMPLICIT Cancel-RequestPDU,
        cancel-ResponsePDU   [6] IMPLICIT Cancel-ResponsePDU,
        cancel-ErrorPDU      [7] IMPLICIT Cancel-ErrorPDU,
    ELSE
        cancel-RequestPDU    [5] IMPLICIT NULL,
        cancel-ResponsePDU   [6] IMPLICIT NULL,
        cancel-ErrorPDU      [7] IMPLICIT NULL,
    ENDIF
    initiate-RequestPDU      [8] IMPLICIT Initiate-RequestPDU,
    initiate-ResponsePDU     [9] IMPLICIT Initiate-ResponsePDU,
    initiate-ErrorPDU        [10] IMPLICIT Initiate-ErrorPDU,
    conclude-RequestPDU      [11] IMPLICIT Conclude-RequestPDU,
    conclude-ResponsePDU     [12] IMPLICIT Conclude-ResponsePDU,
    conclude-ErrorPDU        [13] IMPLICIT Conclude-ErrorPDU
}

```

| MMS PDU               | TAG |
|-----------------------|-----|
|                       |     |
| confirmed-RequestPDU  | a0  |
| confirmed-ResponsePDU | a1  |
| confirmed-ErrorPDU    | a2  |
| unconfirmed-PDU       | a3  |
| rejectPDU             | a4  |
| cancel-RequestPDU     | a5  |
| cancel-ResponsePDU    | a6  |
| cancel-ErrorPDU       | a7  |
| initiate-RequestPDU   | a8  |
| initiate-ResponsePDU  | a9  |
| initiate-ErrorPDU     | aa  |
| conclude-RequestPDU   | ab  |
| conclude-ResponsePDU  | ac  |
| conclude-ErrorPDU     | ad  |



## 4. Análisis de mensajes MMS

### Initiate

```
Initiate-RequestPDU ::= SEQUENCE {  
    localDetailCalling  
    proposedMaxServOutstandingCalling  
    proposedMaxServOutstandingCalled  
    proposedDataStructureNestingLevel  
    initRequestDetail  
        proposedVersionNumber  
        proposedParameterCBB  
        servicesSupportedCalling  
    ...  
    IF (csr cspi)  
        , additionalSupportedCalling  
    ,  
    IF (cspi)  
        , additionalCbbSupportedCalling  
        privilegeClassIdentityCalling  
    ,  
    ENDIF  
    }  
    }  
  
Initiate-ResponsePDU ::= SEQUENCE {  
    localDetailCalled  
    negotiatedMaxServOutstandingCalling  
    negotiatedMaxServOutstandingCalled  
    negotiatedDataStructureNestingLevel  
    initResponseDetail  
        negotiatedVersionNumber  
        negotiatedParameterCBB  
        servicesSupportedCalled  
    ...  
    IF (csr cspi)  
        , additionalSupportedCalled  
    ,  
    IF (cspi )  
        , additionalCbbSupportedCalled  
        privilegeClassIdentityCalled  
    ,  
    ENDIF  
    }  
    }
```

[0] IMPLICIT Integer32 OPTIONAL,  
[1] IMPLICIT Integer16,  
[2] IMPLICIT Integer16,  
[3] IMPLICIT Integer8 OPTIONAL,  
[4] IMPLICIT SEQUENCE {  
[0] IMPLICIT Integer16,  
[1] IMPLICIT ParameterSupportOptions,  
[2] IMPLICIT ServiceSupportOptions ,  
  
[3] IMPLICIT AdditionalSupportOptions  
  
[4] IMPLICIT AdditionalCBBOptions,  
[5] IMPLICIT VisibleString

[0] IMPLICIT Integer32 OPTIONAL,  
[1] IMPLICIT Integer16,  
[2] IMPLICIT Integer16,  
[3] IMPLICIT Integer8 OPTIONAL,  
[4] IMPLICIT SEQUENCE {  
[0] IMPLICIT Integer16,  
[1] IMPLICIT ParameterSupportOptions,  
[2] IMPLICIT ServiceSupportOptions,  
  
[3] IMPLICIT AdditionalSupportOptions  
  
[4] IMPLICIT AdditionalCBBOptions,  
[5] IMPLICIT VisibleString



## 4. Análisis de mensajes MMS

### PARAMETER y SERVICE SUPPORT

```
ParameterSupportOptions ::= BIT STRING {
    str1          (0),
    str2          (1),
    vnam          (2),
    valt          (3),
    vadr          (4),
    -- bit 5 is reserved for the services defined in Annex E.
    vsca          (5),
    tpy           (6),
    vlis          (7),
    -- bit 8 is reserved
    -- bit 9 is reserved
    cei           (10),
    aco           (11),
    sem           (12),
    csr           (13),
    csnc          (14),
    csplc         (15),
    cspl          (16),
    char          (17) } (SIZE(18))
```

```
ServicesSupportOptions ::= BIT STRING {
    status        (0),
    getNameList   (1),
    identify      (2),
    rename        (3),
    read          (4),
    write         (5),
    getVariableAccessAttributes (6),
    defineNamedVariable (7),
    -- bit 8 is reserved for use of a service defined in annex E
    defineScatteredAccess (8),
    -- bit 9 is reserved for use of a service defined in annex E
    getScatteredAccessAttributes (9),
    deleteVariableAccess (10),
    defineNamedVariableList (11),
    getNamedVariableListAttributes (12),
    deleteNamedVariableList (13),
    defineNamedType (14),
    getNamedTypeAttributes (15),
    deleteNamedType (16),
    input         (17),
    output        (18),
    takeControl   (19),
    relinquishControl (20),
    defineSemaphore (21),
    deleteSemaphore (22),
    reportSemaphoreStatus (23),
    reportPoolSemaphoreStatus (24),
    reportSemaphoreEntryStatus (25),
    initiateDownloadSequence (26),
    downloadSegment (27),
    terminateDownloadSequence (28),
    initiateUploadSequence (29),
    uploadSegment (30),
    terminateUploadSequence (31),
    requestDomainDownload (32),
    requestDomainUpload (33),
    loadDomainContent (34) .
```



## 4. Análisis de mensajes MMS

---

### PDU CONFIRMED REQUEST

```
Confirmed-RequestPDU ::= SEQUENCE {  
    invokeID                Unsigned32,  
    IF (attachToEventCondition attachToSemaphore )  
        listOfModifiers      SEQUENCE OF Modifier OPTIONAL,  
    ENDIF  
    service                  ConfirmedServiceRequest,  
    ...
```



## 4. Análisis de mensajes MMS

### CONFIRMED SERVICE REQUEST

```
ConfirmedServiceRequest ::= CHOICE {  
  IF ( status )  
    status  
    [0] IMPLICIT Status-Request  
  ELSE  
    status  
    [0] IMPLICIT NULL  
  ENDIF  
  IF ( getNameList )  
    , getNameList  
    [1] IMPLICIT GetNameList-Request  
  ELSE  
    , getNameList  
    [1] IMPLICIT NULL  
  ENDIF  
  IF ( identify )  
    , identify  
    [2] IMPLICIT Identify-Request  
  ELSE  
    , identify  
    [2] IMPLICIT NULL  
  ENDIF  
  IF ( rename )  
    , rename  
    [3] IMPLICIT Rename-Request  
  ELSE  
    , rename  
    [3] IMPLICIT NULL  
  ENDIF  
  IF ( read )  
    , read  
    [4] IMPLICIT Read-Request  
  ELSE  
    , read  
    [4] IMPLICIT NULL  
  ENDIF  
  IF ( write )  
    , write  
    [5] IMPLICIT Write-Request  
  ELSE  
    , write  
    [5] IMPLICIT NULL  
  ENDIF  
  IF ( vnam vadr )
```



## 4. Análisis de mensajes MMS

---

### READ

```
Read-Request ::= SEQUENCE {  
    specificationWithResult      [0] IMPLICIT BOOLEAN DEFAULT FALSE,  
    variableAccessSpecification [1] VariableAccessSpecification }  
  
Read-Response ::= SEQUENCE {  
    variableAccessSpecification [0] VariableAccessSpecification OPTIONAL,  
    listOfAccessResult          [1] IMPLICIT SEQUENCE OF AccessResult }
```

### WRITE

```
Write-Request ::= SEQUENCE {  
    variableAccessSpecification VariableAccessSpecification,  
    listOfData                  [0] IMPLICIT SEQUENCE OF Data }  
  
Write-Response ::= SEQUENCE OF CHOICE {  
    failure      [0] IMPLICIT DataAccessError,  
    success      [1] IMPLICIT NULL }
```



## 4. Análisis de mensajes MMS

---

### VARIABLE ACCESS SPECIFICATION

```
VariableAccessSpecification ::= CHOICE {  
    listOfVariable      [0] IMPLICIT SEQUENCE OF SEQUENCE {  
        variableSpecification      VariableSpecification,  
    IF ( valt )  
        alternateAccess      [5] IMPLICIT AlternateAccess OPTIONAL  
    ENDIF  
    }  
    IF ( vlis )  
    , variableListName      [1] ObjectName  
    ENDIF  
    }
```

### VARIABLE SPECIFICATION

```
VariableSpecification ::= CHOICE {  
    IF ( vnam )  
        name      [0] ObjectName,  
    ENDIF  
    IF ( vadr )  
        address      [1] Address,  
        variableDescription [2] IMPLICIT SEQUENCE {  
            address      Address,  
            typeSpecification      TypeSpecification  
        },  
    ENDIF  
    -- the following element is only present to support the services  
    -- defined in annex E  
    IF ( vsca )  
        scatteredAccessDescription      [3] IMPLICIT ScatteredAccessDescription,  
    ELSE  
        scatteredAccessDescription      [3] IMPLICIT NULL,  
    ENDIF  
    invalidated      [4] IMPLICIT NULL  
    }
```





## 4. Análisis de mensajes MMS

---

### OBJECT NAME

```
ObjectName ::= CHOICE {  
    vmd-specific      [0] IMPLICIT Identifier,  
    domain-specific   [1] IMPLICIT SEQUENCE {  
        domainID      Identifier,  
        itemID         Identifier  
    },  
    aa-specific       [2] IMPLICIT Identifier  
}
```



## 4. Análisis de mensajes MMS

---

### UNCONFIRMED PDU

```
Unconfirmed-PDU ::= SEQUENCE {
    service          UnconfirmedService,
    ...
    IF ( cspi )
        , service-ext      [79] Unconfirmed-Detail OPTIONAL
    ENDIF
    -- shall not be transmitted if value is the value
    -- of a tagged type derived from NULL
}
```

### UNCONFIRMED SERVICE

```
UnconfirmedService ::= CHOICE {
    IF (informationReport )
        informationReport
        [0] IMPLICIT InformationReport
    ELSE
        informationReport
        [0] IMPLICIT NULL
    ENDIF
    IF ( unsolicitedStatus )
        , unsolicitedStatus
        [1] IMPLICIT UnsolicitedStatus
    ELSE
        , unsolicitedStatus
        [1] IMPLICIT NULL
    ENDIF
    IF ( eventNotification )
        , eventNotification
    }
```



## 4. Análisis de mensajes MMS

---

### INFORMATION REPORT

```
InformationReport ::= SEQUENCE {  
    variableAccessSpecification    VariableAccessSpecification,  
    listOfAccessResult             [0] IMPLICIT SEQUENCE OF AccessResult }
```

### ACCESS RESULT

```
AccessResult ::= CHOICE {  
    failure    [0] IMPLICIT DataAccessError,  
    success    Data }
```



## 4. Análisis de mensajes MMS

---

### DATA

```
Data ::= CHOICE {  
    -- context tag 0 is reserved for AccessResult  
    IF ( str1 )  
        array [1] IMPLICIT SEQUENCE OF Data,  
    ELSE  
        array [1] IMPLICIT NULL,  
    ENDIF  
    IF ( str2 )  
        structure [2] IMPLICIT SEQUENCE OF Data,  
    ELSE  
        structure [2] IMPLICIT NULL,  
    ENDIF  
    boolean [3] IMPLICIT BOOLEAN,  
    bit-string [4] IMPLICIT BIT STRING,  
    integer [5] IMPLICIT INTEGER,  
    unsigned [6] IMPLICIT INTEGER, -- shall not be negative  
    floating-point [7] IMPLICIT FloatingPoint,  
    -- [8] is reserved  
    octet-string [9] IMPLICIT OCTET STRING,  
    visible-string [10] IMPLICIT VisibleString,  
    generalized-time [11] IMPLICIT GeneralizedTime,  
    binary-time [12] IMPLICIT TimeOfDay,  
    bcd [13] IMPLICIT INTEGER, -- shall not be negative  
    booleanArray [14] IMPLICIT BIT STRING,  
    objId [15] IMPLICIT OBJECT IDENTIFIER,  
    ...,  
    mMSString [16] IMPLICIT MMSString  
}
```



## 4. Análisis de mensajes MMS

---

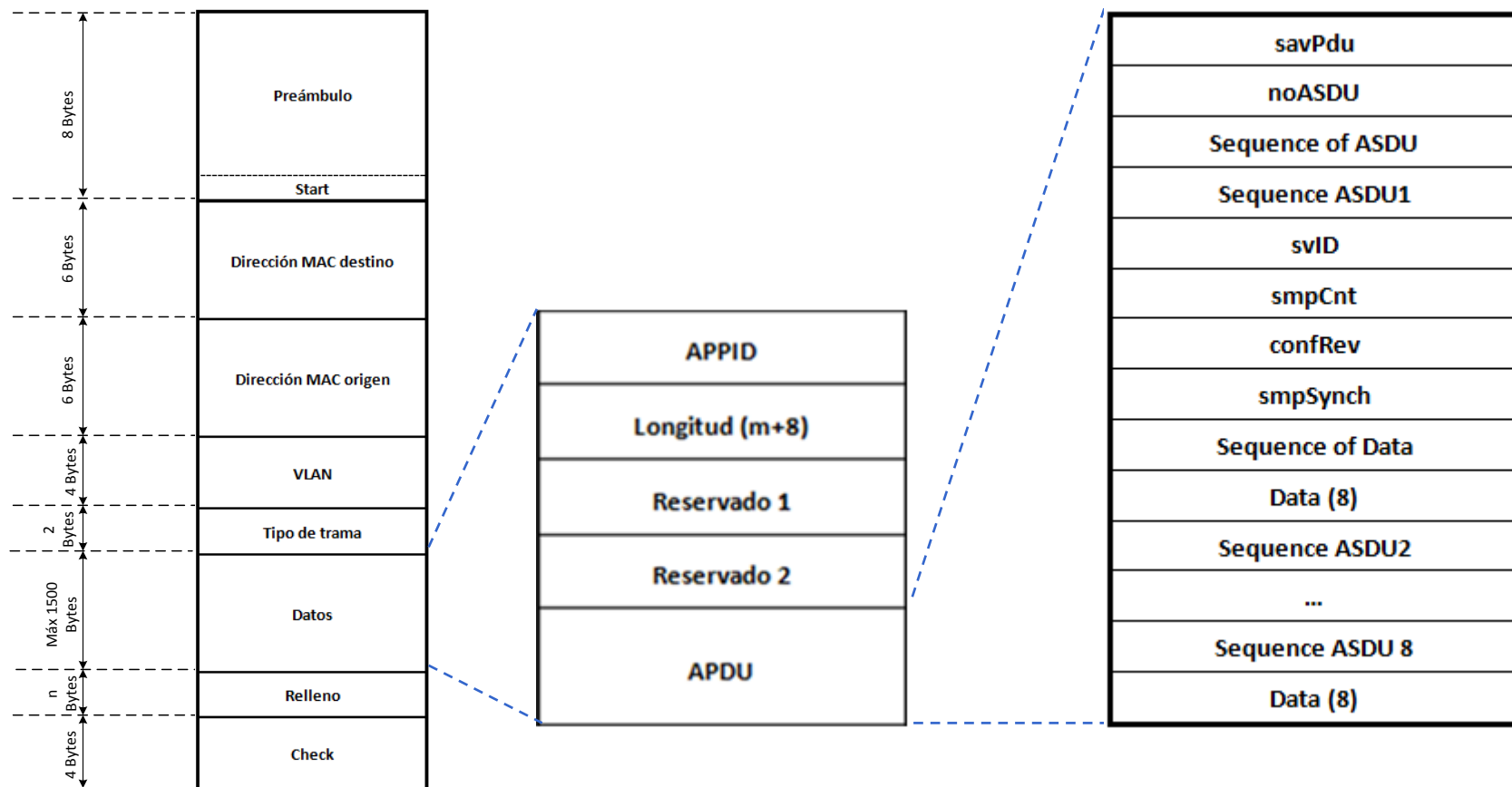
| Oper   |                                                              |
|--------|--------------------------------------------------------------|
| ctlVal | Valor de control cuya modificación causa el cambio de estado |
| Origin | Origen del mando                                             |
| ctlNum | Número de secuencia                                          |
| operTm | Momento de ejecución del mando                               |
| T      | Momento de la emisión del mando                              |
| Test   | Ejecución / Prueba                                           |
| Check  | Verificaciones previas a la realización del mando            |



# Análisis de mensajes Sampled Values

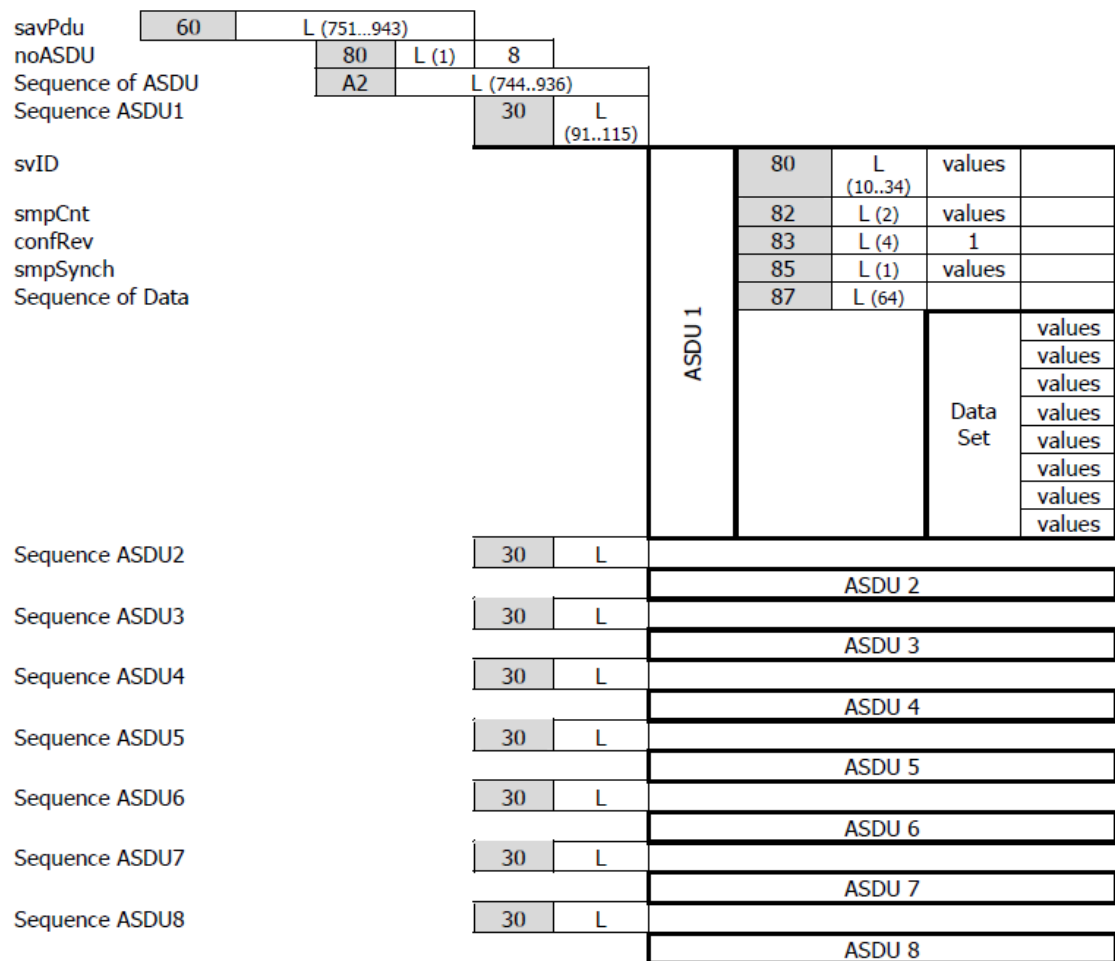
## 5. Análisis Sampled Values

### MENSAJE SV – Lite Edition



## 5. Análisis Sampled Values

### MENSAJE SV – Lite Edition





## 5. Análisis Sampled Values

### El protocolo SV

IEC 61850 9-2 Lite Edition

|                  |
|------------------|
| savPdu           |
| noASDU           |
| Sequence of ASDU |
| Sequence ASDU1   |
| svid             |
| smpCnt           |
| confRev          |
| smpSynch         |
| Sequence of Data |
| Data (8)         |
| Sequence ASDU2   |
| ...              |
| Sequence ASDU 8  |
| Data (8)         |

|                          |   |   |   |   |   |   |   |     |
|--------------------------|---|---|---|---|---|---|---|-----|
| msb                      | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1   |
| InnATCTR1.Amp.InstMag.I  |   |   |   |   |   |   |   |     |
| InnATCTR1.Amp.q          |   |   |   |   |   |   |   | lsb |
|                          |   |   |   |   |   |   |   |     |
|                          |   |   |   |   |   |   |   |     |
| InnBTCTR2.Amp.InstMag.I  |   |   |   |   |   |   |   |     |
| InnBTCTR2.Amp.q          |   |   |   |   |   |   |   |     |
| InnCTCTR3.Amp.InstMag.I  |   |   |   |   |   |   |   |     |
| InnCTCTR3.Amp.q          |   |   |   |   |   |   |   |     |
| InnNmTCTR4.Amp.InstMag.I |   |   |   |   |   |   |   |     |
| InnNmTCTR4.Amp.q         |   |   |   |   |   |   |   |     |
| UnnATVTR1.Vol.InstMag.I  |   |   |   |   |   |   |   |     |
| UnnATVTR1.Vol.q          |   |   |   |   |   |   |   |     |
| UnnBTVTR2.Vol.InstMag.I  |   |   |   |   |   |   |   |     |
| UnnBTVTR2.Vol.q          |   |   |   |   |   |   |   |     |
| UnnCTVTR3.Vol.InstMag.I  |   |   |   |   |   |   |   |     |
| UnnCTVTR3.Vol.q          |   |   |   |   |   |   |   |     |
| UnnNmTVTR4.Vol.InstMag.I |   |   |   |   |   |   |   |     |
| UnnNmTVTR4.Vol.q         |   |   |   |   |   |   |   |     |



Gracias por su atención

-----

Yasmina Galve Pastor – [ygalve@fcirce.es](mailto:ygalve@fcirce.es)