AMHS Workshop 2

2023

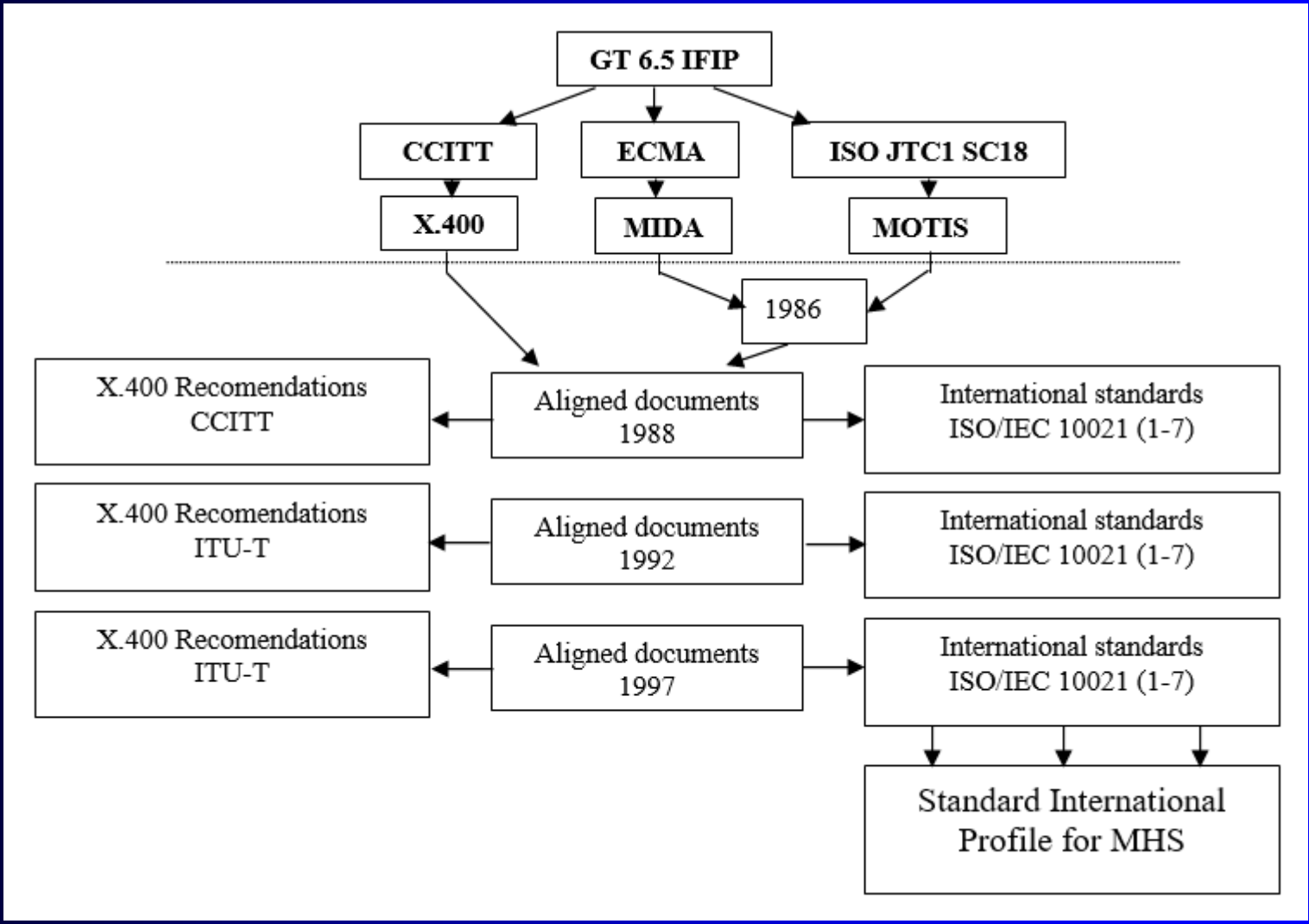
Francisco Almeida – SAM CNS Officer

X.400 x SMTP

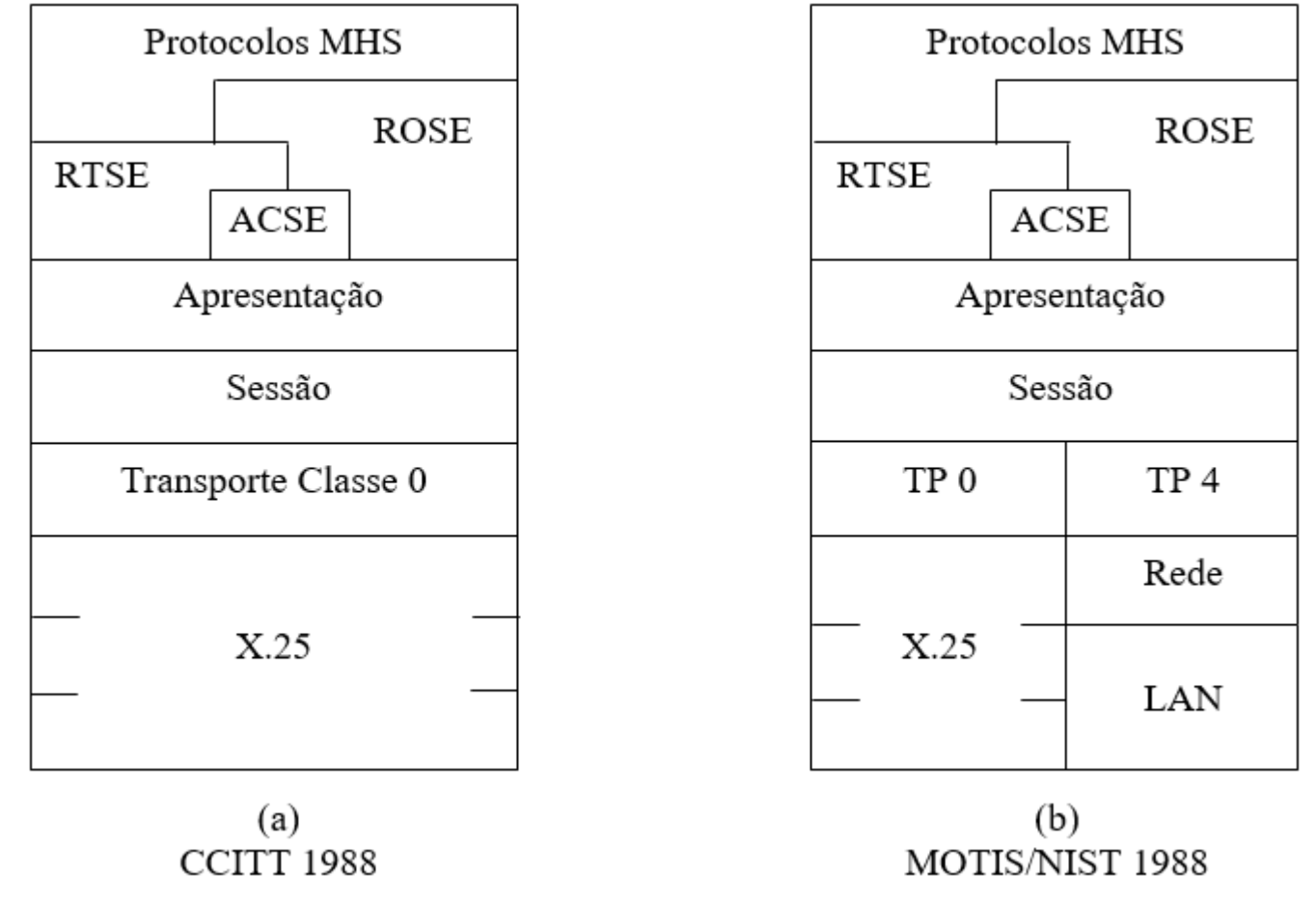
- X.400:
- Most succinct message format
- can restrict the topology of the backbone
- better notifications
- closely related to X.435, the current EDI standard
- Closer to X.500, which offers a more efficient directory service than LDAP

- SMTP:
- Simpler to implement and maintain
- It has the "impulse" of Internet
- It is more flexible and adaptable

X.400/MOTIS

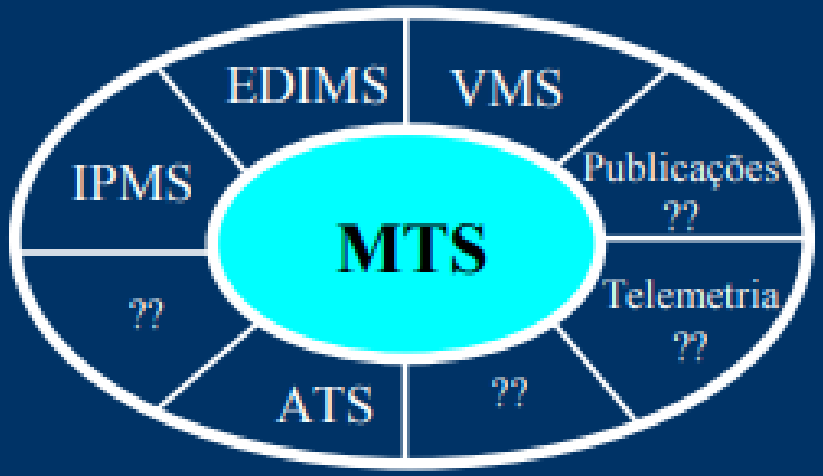

GT 6.5 IFIP
CCITT
ECMA
ISO JTC1 SC18
X.400
MIDA
MOTIS
1986
X.400 Recomendations CCITT
Aligned documents 1988
International standards ISO/IEC 10021 (1-7)
X.400 Recomendations ITU-T
Aligned documents 1992
International standards ISO/IEC 10021 (1-7)
X.400 Recomendations ITU-T
Aligned documents 1997
International standards ISO/IEC 10021 (1-7)
Standard International Profile for MHS

MHS – Layered architecture

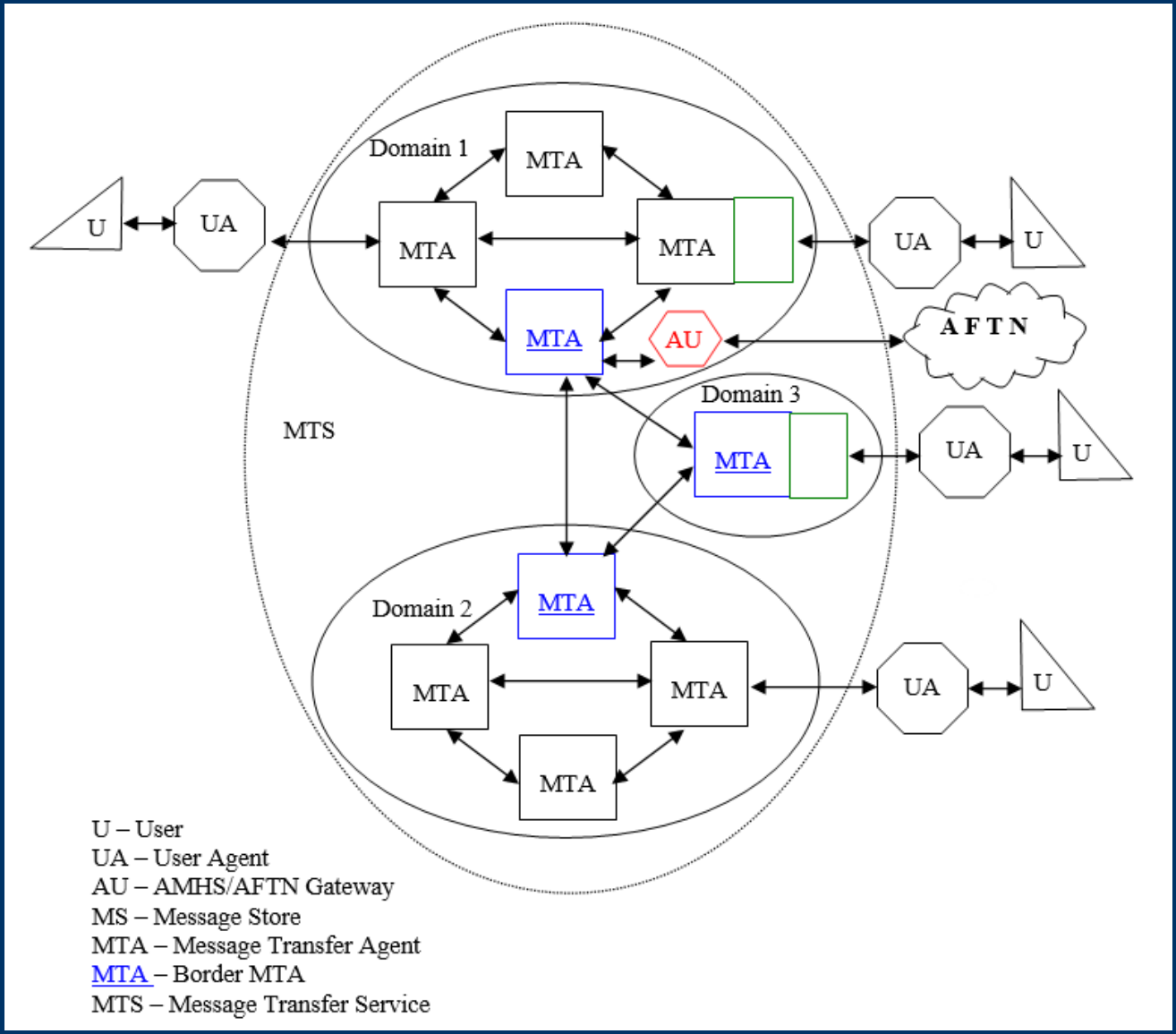


MHS Services

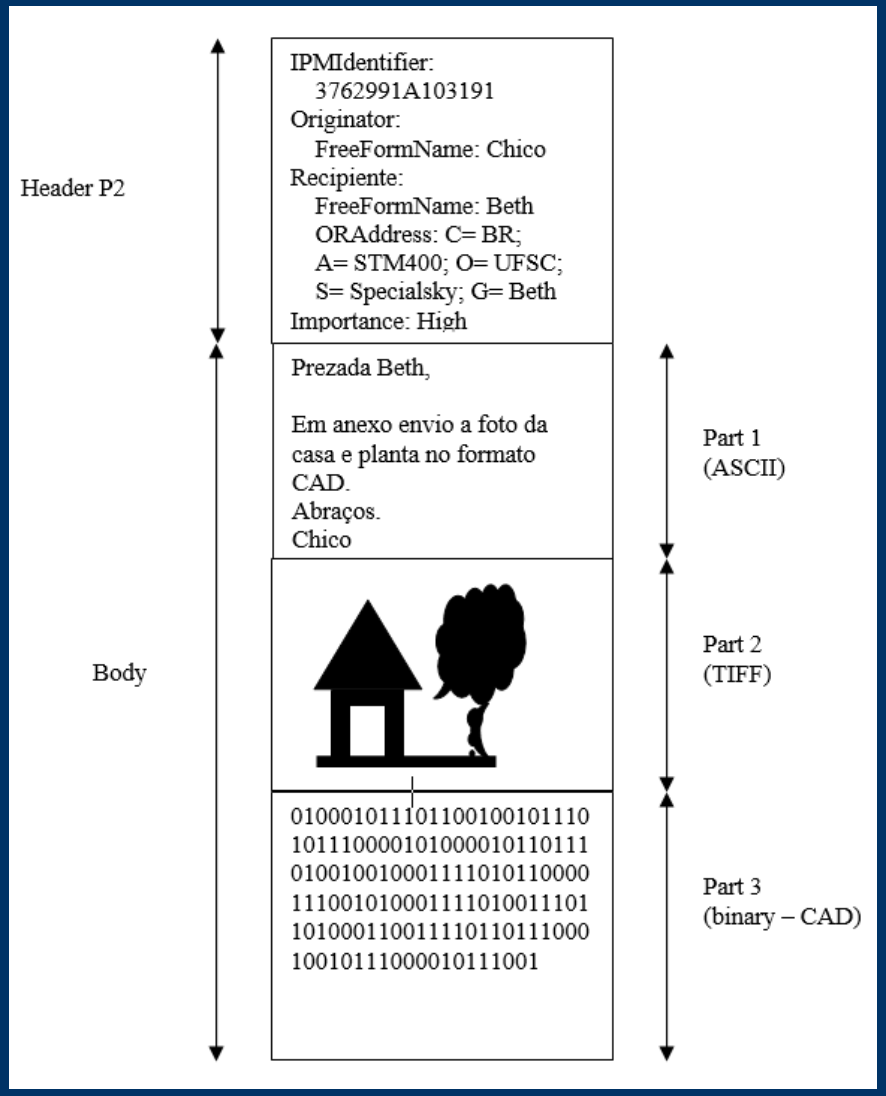
- MHS standards include specifications for a number of services for users:
  - Message Transfer Service (MTS)
  - Interpersonal Message Service (IPM)
  - Electronic Data Interchange Messaging Service (EDI)
  - Other services



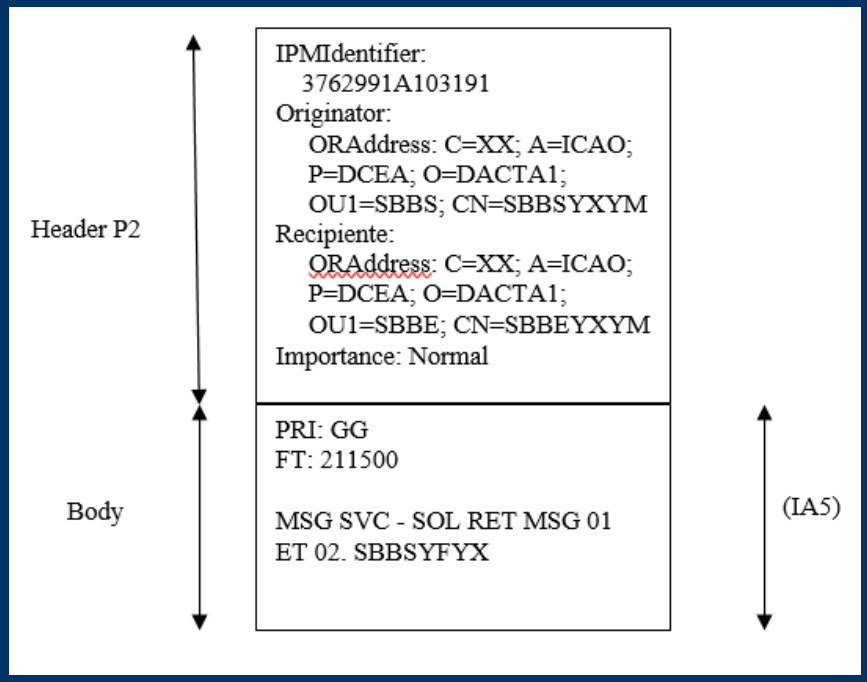
AMHS Functional Model



Information Model

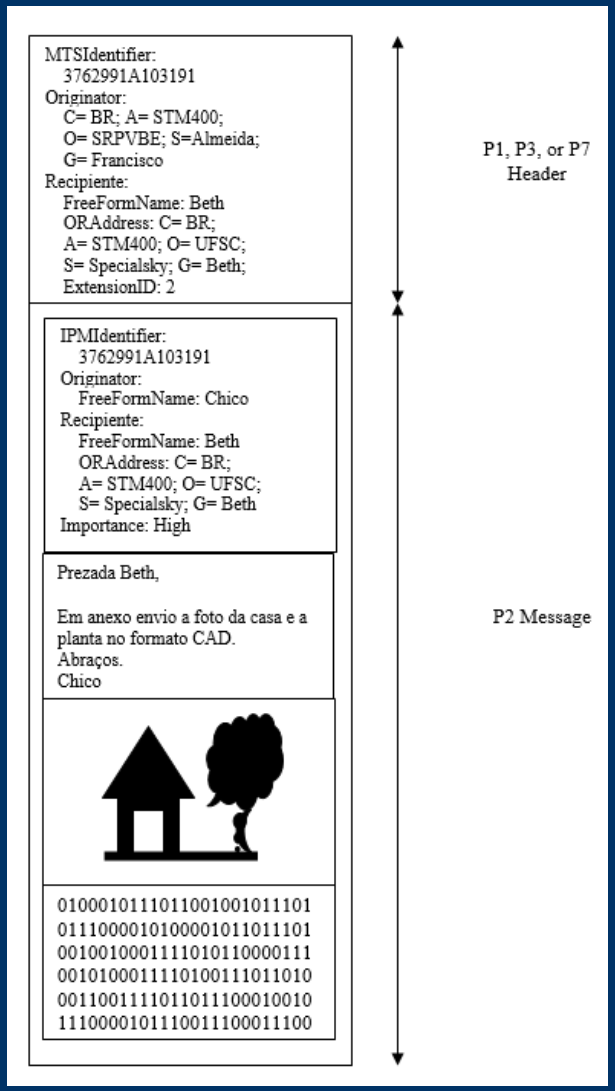


MHS IPM (P2)

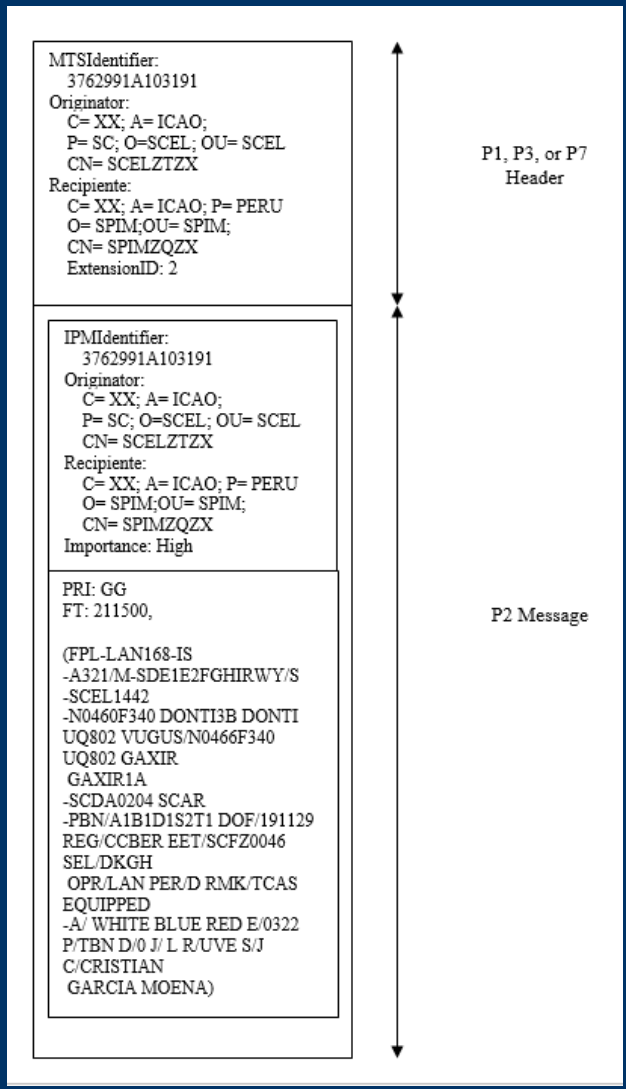


AMHS IPM (P2)

Information Model

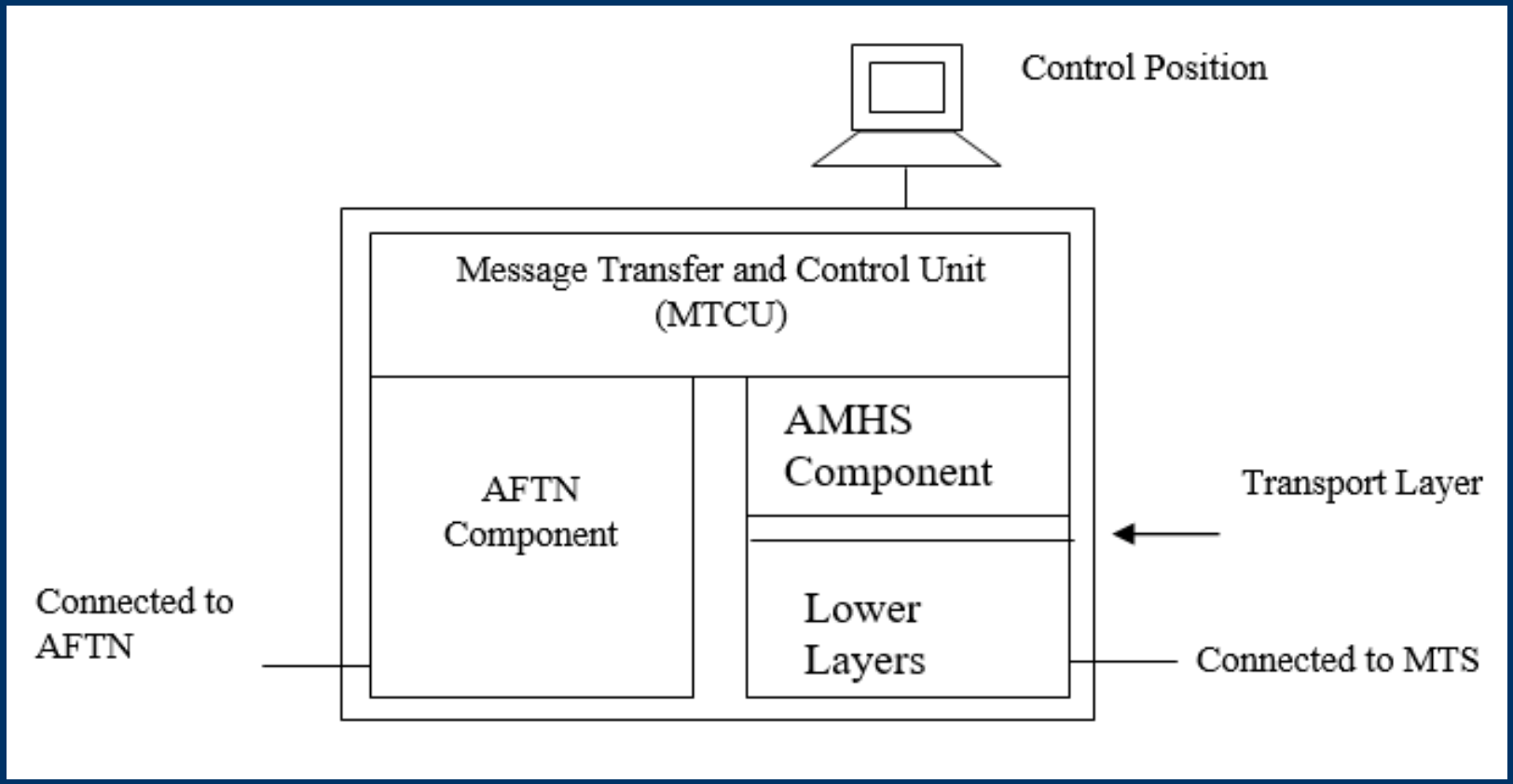


MHS



AMHS

Gateway AFTN/AMHS



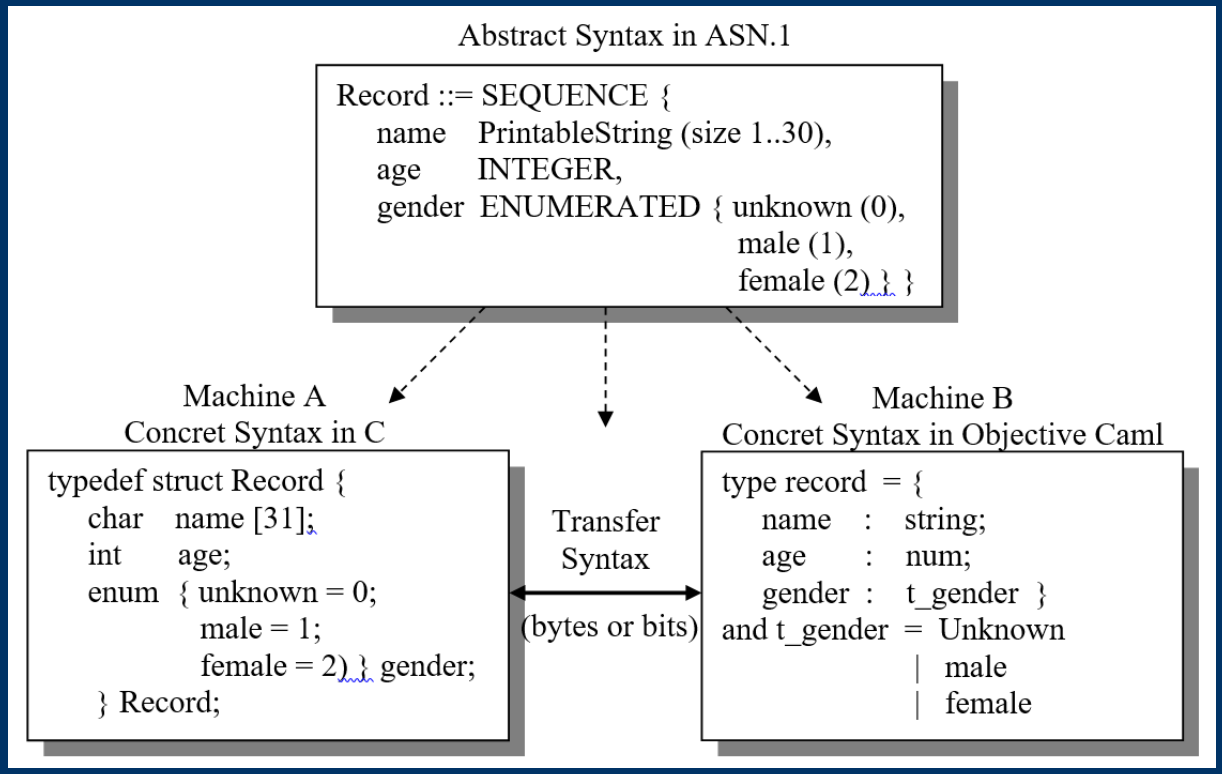
Protocols

- "A well-defined set of messages (bit pattern), each with a definite meaning (semantics), along with rules governing how and when a transmission should be executed."

- Two widely used possibilities:
  - **Character-based specification:** The protocol is defined as a series of lines of ASCII-encoded text; and
  - **Bit-based specification (binary):** The protocol is defined as a string of bit octets.

Syntaxes

- Concrete Syntax
- Abstract Syntax
- Transfer Syntax



Abstract Syntax Notation 1 (ASN.1)

- Standard Notation
- Define complex data structures
- Developed for the X.400 Protocol
- Used to specify other protocols
  - X.500
  - ISDN
  - SNMP
- Used for the specification of the ATN (Aeronautical Telecommunication Network) protocols

ASN.1 Type Classes

- Universal
- Wide application
- Context-specific
- Private

Each type is distinguished by a label, which specifies the class of the type and identifies the type particularly. For example, UNIVERSAL 4 refers to OctetString, which is of the UNIVERSAL class and has ID 4 within that class.

Personnel Record

- Name: John P. Smith
- Title: Director
- Employee Number: 51
- Date of Hire: 17 September 1971
- Name of Spouse: Mary T. Smith
- Number of Child: 2
- Child Information
- Name: Ralph T. Smith
- Date of Birth: 11 November 1957
- Child Information
- Name: Susan B. Jones
- Date of Birth: 17 July 1959

Formal Description of the Personnel Record

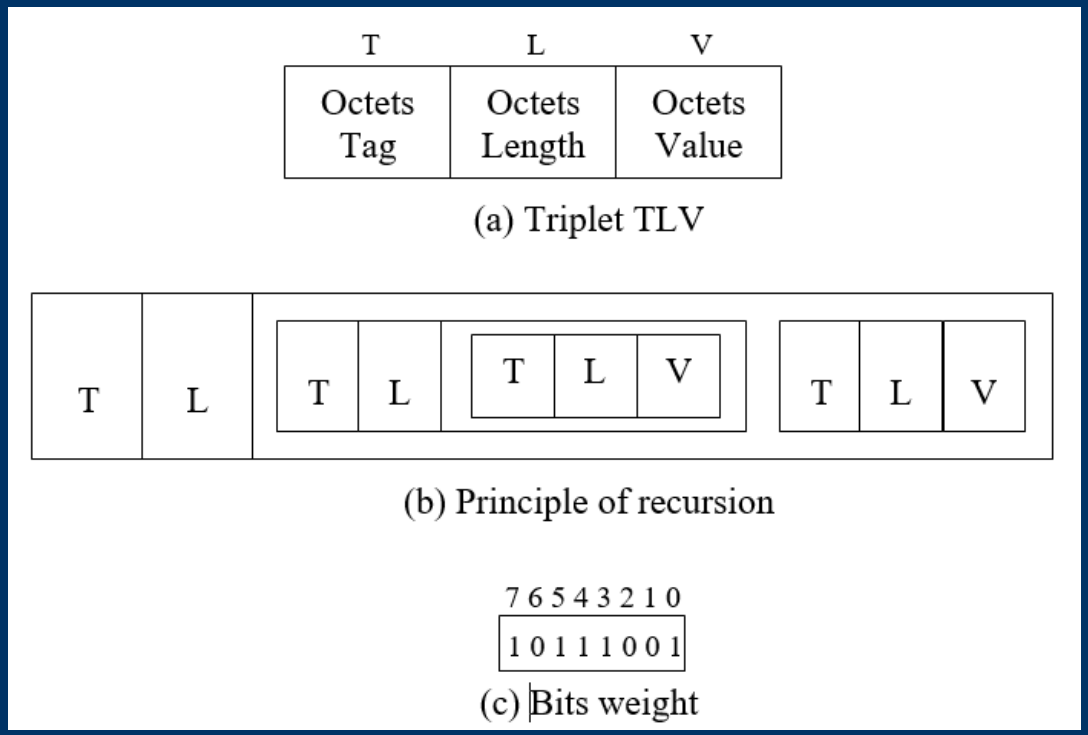
```
PersonnelRecord ::= [APPLICATION 0] IMPLICIT SET {
        Name,
        title [0] IA5String,
        EmployeeNumber,
        dateOfHire [1] Date,
        nameOfSpouse [2] Name,
        [3] IMPLICIT SEQUENCE OF ChildInformation DEFAULT {}}

ChildInformation ::= SET {
        Name,
        dateOfBirth [0] Date}

Name ::= [APPLICATION 1] IMPLICIT SEQUENCE {
        givenName IA5String,
        initial IA5String,
        FamilyName IA5String}

EmployeeNumber ::= [APPLICATION 2] IMPLICIT INTEGER

Date ::= [APPLICATION 3] IMPLICIT IA5String -- YYYYMMDD
```

Formal Description of the Personnel Record

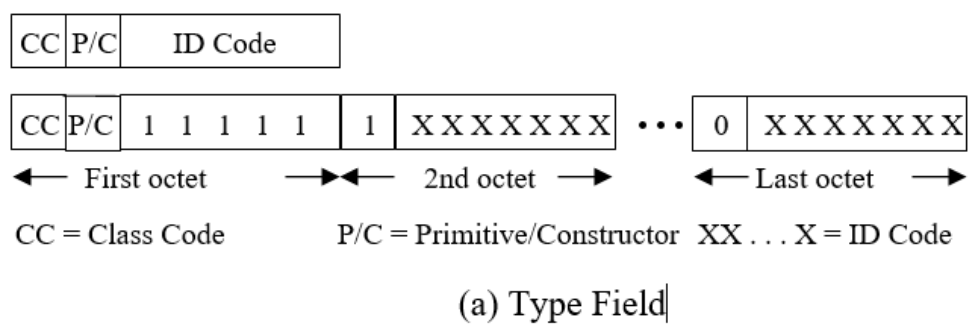
```
PersonnelRecord ::= [APPLICATION 0] IMPLICIT SET {
        Name,
        title [0] IA5String,
        EmployeeNumber,
        dateOfHire [1] Date,
        nameOfSpouse [2] Name,
        [3] IMPLICIT SEQUENCE OF ChildInformation DEFAULT {}}

ChildInformation ::= SET {
        Name,
        dateOfBirth [0] Date}

Name ::= [APPLICATION 1] IMPLICIT SEQUENCE {
        givenName IA5String,
        initial IA5String,
        FamilyName IA5String}

EmployeeNumber ::= [APPLICATION 2] IMPLICIT INTEGER

Date ::= [APPLICATION 3] IMPLICIT IA5String -- YYYYMMDD
```

Basic Encoding Rules

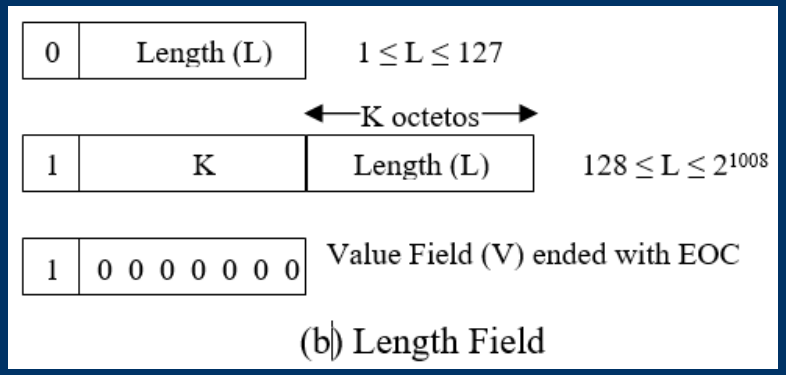
The Basic Encoding Rules (BER) transfer syntax is always in the format of a TLV triplet (Type, Lenght and Value) also expressed as <Tag, Length,Value>. All fields T, L, and V are series of octets. The field V can have in itself another triplet T, L and V, if it is a constructed type, see figure (b).



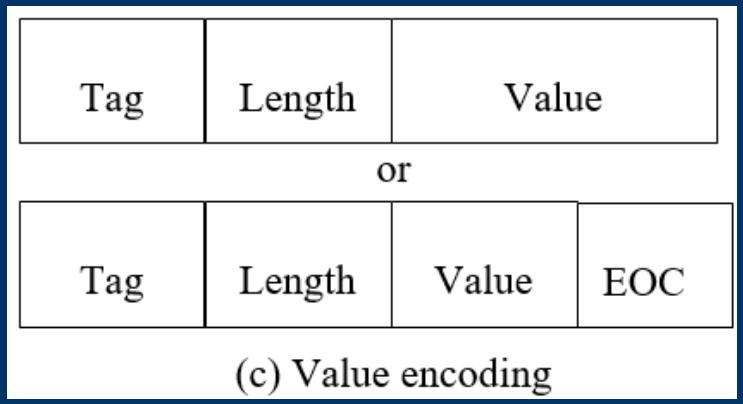
Basic Encoding Rules



(a) Type Field



(b) Length Field



(c) Value encoding