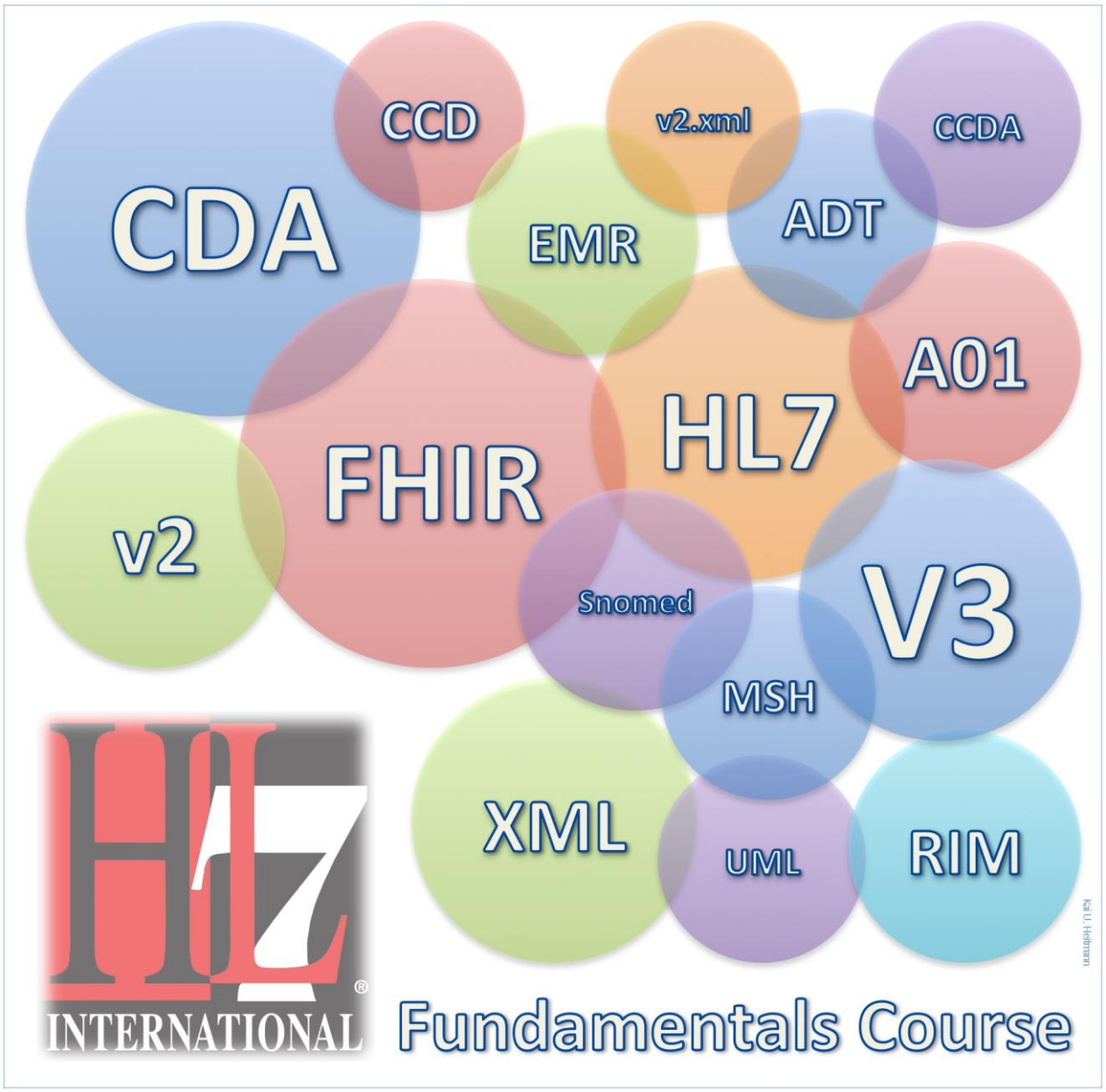

V2.x Module, Unit 1:

Introduction to HL7 V2.x

Reading Material

©2014 HL7 Argentina & HL7 International

Table of Contents

Table of Contents 2
Unit Content and Learning Objectives 3
1. What is Health Level 7 4
HL7 Mission 4
2. What is HL7 V2.x? 7
Versions 7
3. What is the Structure of the HL7 V2.x Standard? 9
4. How to Use the HL7 V2.x Standard 11
Communication Environment 12
5. What is a V2.x Message? 13
Segments 13
Fields 17
Components 17
Subcomponents 17
Message Delimiters 18
Summary 18
6. The V2.x Data Types 19
Data Types Overview 19
Data Types for Strings and Text 19
Data Types for Numbers and Quantities 20
Data Types for Times and Dates 21
Data Types for Identifiers 22
Data Types for Coded Values 24
Data Types for Persons, Organizations, Addresses and Phone Numbers 24
Data Types for Multimedia Objects 31
7. HL7 V2.x Message Processing Rules 33
HL7 V2.x Outbound Message Creation Rules 33
HL7 V2.x Inbound Message Creation Rules 33
Acknowledgement of Message Reception (ACK) 34
Original Acknowledgement Mode 34
Enhanced Acknowledgement Mode 35
Unit Summary and Conclusion 37
Additional Reference Material 38

Unit Content and Learning Objectives

The goals of this Unit are that you will learn how to:

- Understand the mechanics of HL7 Version 2 message exchange
- Write a simple V2.x message
- Read and "understand" an HL7 V2.x message

Note: In general aspects of HL7 Version 2, this Unit refers to V2.x; however, where specific characteristics of V2.x are described such as Messages/Trigger Events, Data Types, etc., the Unit refers to V2.8 of the Version 2 standard.

1. What is Health Level 7

The Health Level 7 organization was founded in 1987 by a group of hospital information technology users in the USA. Since then HL7 has become the globally authoritative standards development organization (SDO) healthcare systems interoperability.

The HL7 healthcare information exchange standards are at the application level, e.g. layer seven of the OSI (Open Systems Interconnection) network model, hence its name: Health Level Seven.

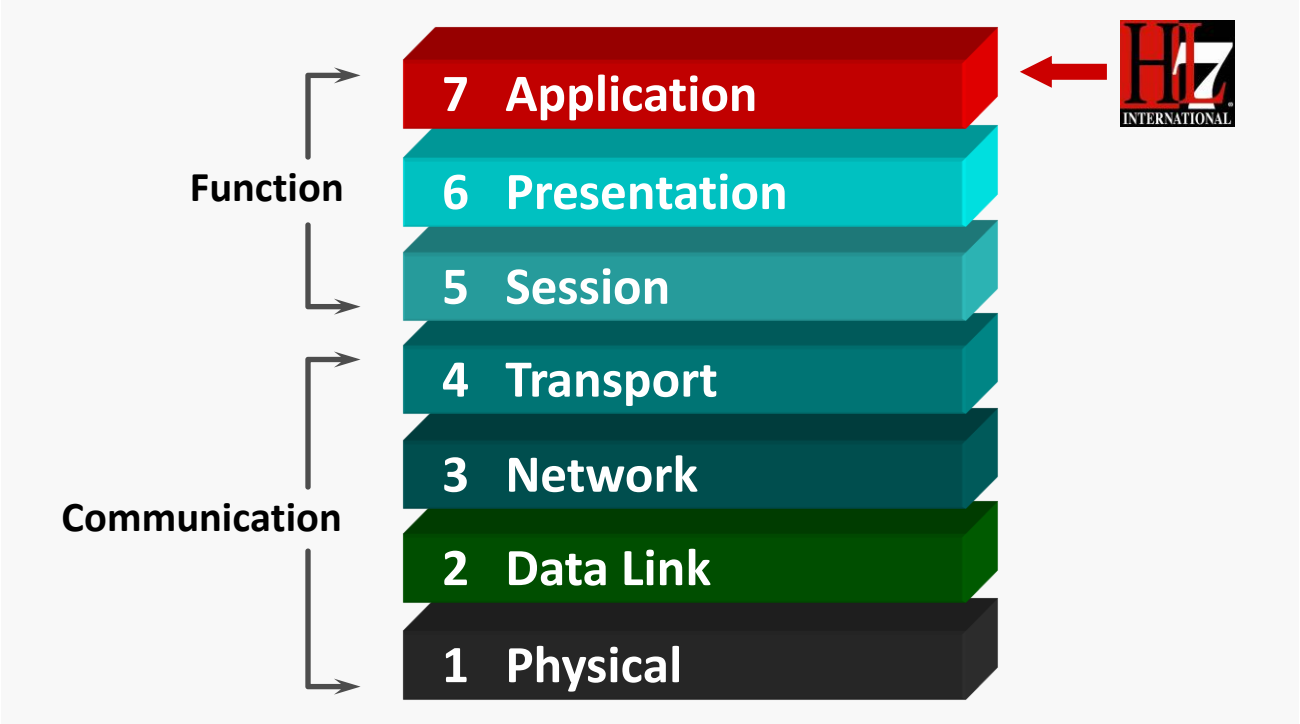

Figure 1: The 7-Layer ISO Communications Model

For more information on the ISO OSI network model, the following reading is recommended: http://en.wikipedia.org/wiki/OSI_model

HL7 Mission

The HL7 International organization defines its mission as providing standards for interoperability that improve care delivery, optimize workflow, reduce ambiguity and enhance knowledge transfer among all of stakeholders, including healthcare providers, government agencies, the vendor community, other SDOs and patients. See www.HL7.org/about

The HL7 standards define messages and message exchange protocols that support clinical practice and management. In Unit 1 it was stated that *clinical management* is a broad concept that seeks to analyze the "cost and effectiveness of medical acts" in order to define healthcare policies that will achieve and maintain high quality outcomes. Data exchange between health care applications is essential to achieving that goal.

In its Vision Statement, HL7 International states that it strives to create flexible, cost-effective standards, guidelines and methodologies that provide interoperability between healthcare systems and enables exchange of meaningful electronic health information.

Since 1994, the HL7 International organization has been accredited by the American National Standards Institute (ANSI) as a Standards Developing Organization (SDO) for the USA. As an SDO, HL7 International has adopted a structure and formal procedures to achieve consensus and a balance of interest among its various stakeholders: systems and application vendors, health insurance companies, governmental jurisdictions, universities, hospitals, health care providers, consultants, etc. It does this in a volunteer-based, not-for-profit and democratic environment.

The use of HL7 standards worldwide has been made possible in large part by the more than 500 companies holding organizational membership, representing more than 2300 members in 55 countries, and ~35 International Affiliate organizations. About a quarter of the worldwide membership meets periodically in Working Group Meetings, which are held all over the globe.

Specific HL7 International Work Groups concentrate on various areas or "domains" of healthcare information exchange. The output of the Work Groups must get the consensus and approval of a representation of the HL7 International membership before a standard may be published.

Membership in an HL7 International Work Group is open to anyone interested in a specific area.

Figure 2 lists some of the countries in which HL7 International Affiliate organizations operate:

Argentina	Australia	Austria
Bosnia & Herzegovina	Brazil	Canada
China	Colombia	Croatia
Czech Republic	Finland	France
Germany	Greece	Hong Kong
India	Italy	Japan
Korea	Netherlands	New Zealand
Norway	Pakistan	Philippines
Puerto Rico	Romania	Russia
Singapore	Spain	Sweden
Switzerland	Taiwan	Turkey
UK	Uruguay	

Figure 2: The HL7 International Affiliates

All of these countries have their own independent Affiliate organizations and are represented on the HL7 International Affiliate Council for international harmonization and adapting standards in different parts of the world.[1]

[1] For the current list of HL7 International Affiliates, see www.hl7.org/Special/committees/international/leadership.cfm

This Course module focuses on HL7 V2.x, the first messaging standard for the exchange of clinical, administrative and financial data. HL7 International has also developed other standards including the following:

- **HL7 Clinical Document Architecture (CDA)** - a very important initiative within HL7 International, defining a standard for the exchange of clinical documents, such as discharge or consultation notes and summaries, imaging and laboratory reports, etc. CDA natively uses XML as the communications format (e.g. "wire format" or "encoding"). CDA Release 2.0 ("CDA R2") is closely aligned with the Version 3 Reference Information Model and uses V3 data types. CDA Release 3 is currently being worked on. The CDA specifications are flexible and allow for the use of templates according to different scenarios and settings. A later Module in this Course will cover CDA in detail and examine several examples of CDA implementations.
- The latest HL7 Standard, **Fast Health Interoperability Resources (FHIR)** has quickly become popular after being published in February 2014.
- For **Electronic Decision Support**, HL7 International has developed four standards: Arden Syntax, Guideline Interchange Format (GLIF), "GELLO" and InfoButton. The Arden Syntax is a grammar for expressing and sharing rules of clinical knowledge, which generally are well developed in clinical practice guidelines. For example, a rule can be formulated in Arden Syntax such that "*a patient is to be administered digoxin but only if his potassium is less than 3 mEq/L*" so an alarm may be generated to suspend the medication. GLIF is a format that allows these rules to be easily exchanged. GELLO is a standard query and expression language that provides a suitable framework for manipulation of clinical data for decision support in healthcare and the "InfoButton" standard enables context-aware knowledge retrieval into clinical information systems (CIS), such as electronic health record (EHR) and personal health record (PHR) systems. This is an increasingly promising approach for delivering relevant clinical knowledge to the point of care as well as patient-tailored educational material to support patient-centered care.
- The **HL7 Version 3** standard has not attracted many implementations as yet, but its Reference Information Model (RIM) is an excellent "map" of the healthcare business domains and a useful research and teaching tool. It also provides background to the FHIR and CDA standards.
- **CCOW**, a standard developed by the HL7 International Clinical Context Object Workgroup, is a framework for sharing context (user, patient, study, etc.) between applications and allowing single sign-on and desktop sharing among applications.
- The HL7 International **EHR Functional Model** is a standard that specifies the functionality desired by users of electronic health records ("EHRs")

For the complete list of HL7 Standards, see www.HL7.org/implement/standards/index.cfm

2. What is HL7 V2.x?

HL7 V2.x **is a simple protocol** for the **exchange of clinical data** through ASCII-text-based, character-delimited, positional and variable-length messages. Typically, HL7 V2.x messages are encoded as ASCII text strings with delimiters using pure ASCII text in a flat file format.

What HL7 V2.x *is not* includes the following:

- It is not an application
- It is not a data structure or a database specification.
- It is not an architecture to design applications for hospitals.
- It is not a specification for a message router
- It does not specify operating systems, databases or development environments.

The standard does not define how the messages are sent (TCP/IP, HTTP, file transfers, via a serial port, through a Web service, etc.) These 'low level' transport protocols ("LLP") are left to the implementer. Although HL7 V2.x offers some guidance in this regard, none of these lower-level protocols is mandatory or normative in the HL7 standard.

Initially, HL7 V2.x had few inherent semantic restrictions; the vocabularies to be used in the codified elements of the messages were usually subject to negotiation between the parties. However, the more recent versions of HL7 V2.x have become increasingly strict in specifying code sets for each data item. In V2.6, support for mandatory closed codes sets was introduced with the CNE data type.

HL7 V2.x has few semantic restrictions - it does not have an underlying model, as does HL7 Version 3.

Versions

The first HL7 messaging standard, Version 2.1, was published in June 1990. Since then the following releases of V2 have been published:

V2.1 (1990): First implemented version - very basic, but still used

V2.2 (1994): Basic enhancements

V2.3 (1997): Scheduling and Finance messages added

V2.3.1 (1999): Pathology, Allergies, Referral as Scheduling

V2.4 (2000): Clinical Focus: Referrals and Discharge Summaries

V2.5 (2003): Data field lengths made implementable and standardised

V2.5.1 (2007): Four data items added due to US regulatory requirements

V2.6 (2007): Data type changes: CE → CNE/CWE, TS → DTM
New chapters: e-Claiming & Materials Management
Mood Codes (temporal & action based content)

V2.7 (2011): Collaborative Care Message

V2.7.1 (2012): Lab Orders/Results features added for US "Meaningful Use"

V2.8 (2014): Authorization and Ordering info added; code tables moved to Ch. 2C

Work on V2.9 is well under way and V2.10 is being planned!

HL7 International has also developed specifications for encoding V2.x messages in XML (v2.xml).

This Module is based on V2.8, which was published in February 2014 and includes the content of earlier versions that are widely supported by existing applications.

3. What is the Structure of the HL7 V2.x Standard?

HL7 V2.x is organized into chapters, each of which has a specific purpose or domain:

Chapters 2 and 2A define how to encode and decode messages, Chapter 2B outlines the V2.x approach to Conformance and Ch 2C lists all the code tables. The rest of the chapters concentrate on various areas or "domains" of healthcare.

Chapter 3 contains the messages and message components for the administration of patient visits in healthcare organizations. This domain is also known as Admission, Discharge and Transfer (ADT) or Admission, Transfer and Discharge (ATD). The content of Ch. 3 is used by the other domain chapters in HL7 V2.x:

For example, Chapter 7, "Results" refers to the PID segment, which conveys information about the patient. However, the PID segment is not defined in Chapter 7; instead, the implementer is referred to Chapter 3, which defines the PID segment.

Here is a list of the chapters of the HL7 V2.8 Standard:

Chapter	Description
1	Introduction
2	Control (Message Structure)
2A	Control (Data Types)
2B	Control (Conformance)
2C	Control (Code Tables)
3	Patient Administration (Admission, Discharge & Transfer)
4	Order Entry - part 1 (Laboratory, Dietary, Supply, Blood Transfusion and general use)
4A	Order Entry - part 2 (Pharmacy, Treatment, Vaccination, etc.)
5	Query
6	Financial Management (Billing / Patient Accounts)
7	Observation Reporting (typically from laboratories, imaging, etc.)
8	Master Files
9	Medical Records/Information Management (Document Management)
10	Scheduling
11	Patient Referral
12	Patient Care
13	Clinical Laboratory Automation
14	Application Management
15	Personnel Management

16	Claims & Reimbursement
17	Materials Management

Figure 3: Chapters of Version 2.8

4. How to Use the HL7 V2.x Standard

In order to be able to use the standard, it is essential to know the four "Control" Chapters 2, 2A, 2B and 2C. Thorough knowledge of these chapters is important to enable the application of the rest of the standard. Knowledge of these four chapters is also vital to pass the Control Specialist Certification exam for HL7 V2.8.

The chapters 2, 2A and 2C define the elements (message construction, message types, trigger events, segments, fields and data types) required to build messages for the specific domains of healthcare information systems. Chapter 2B covers the HL7 approach to conformance.

Each of these domain chapters typically includes the following parts:

- Introduction/scope
- trigger events
- message definitions
- examples
- outstanding issues

Figure 4 shows a basic model of an HL7 V2.x transaction:

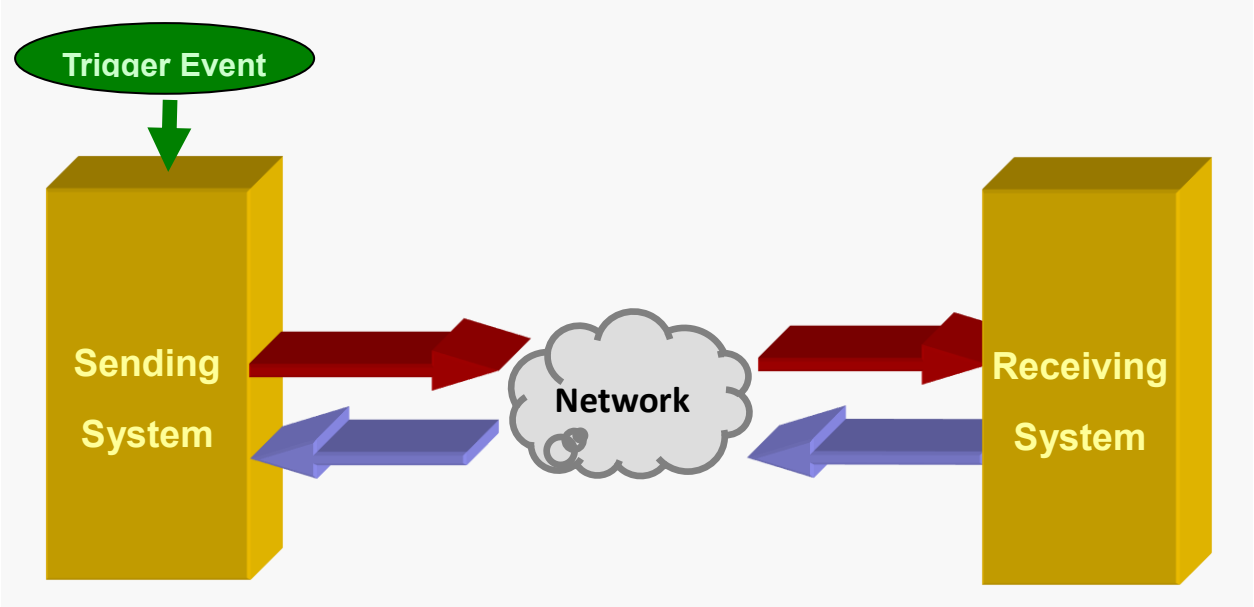

Figure 4: Basic Model of an HL7 V2.x Transaction

The model is quite basic: a trigger event (something that occurred in the real world) causes the Sending System to send a message. This message is received by the Receiving System, which sends a response back to the Sending System.

As a real-life example, when a patient is admitted into the hospital, the patient registration system sends a message to the billing system to inform it of the new patient and request, then an account be opened for this episode. This message is received by the billing system, which sends a response message to the patient registration system confirming that the original message has been processed.

Communication Environment

The HL7 V2.x Standards assume that the communication environment will provide the following capabilities:

- **Error-free transmission** - Applications can assume that they will correctly receive all of the transmitted bytes in the order in which they were sent. This implies that error checking is done at a lower level.
- **Character conversion** - In a case where different machines use different character systems representations (*e.g.*, ASCII, EBCDIC, etc.) the communication environment will be expected to convert the data from one representation to the other.
- **Message Length** - HL7 sets no limits on the length of a message. The standard assumes that the communication environment can transport messages of any length necessary.
- **Security** - HL7 does not specify security and encryption requirements, but assumes that the communications occur through secure channels that maintain the required privacy. Note that HL7 does allow the communication of special security and confidentiality requirements of a patient, such as VIP, special provisions, etc.

5. What is a V2.x Message?

A message is the atomic unit of data transferred between systems. Figure 5 below shows a typical HL7 ADT message:

```
MSH|^~\&|NSI||LAB||2014030627120759||ADT^A01^ADT_A01|NSI1|P|2.8<cr>
EVN|A01|20140327120759<cr>
PID|1|||444-22-2222^^^HC^MR|Everywoman^Eve^E||19780113000000|F|||220 Home Street^^ANN
  ARBOR^^99999<cr>
NK1|1|Everyman^Adam|SPO|220 Home Street|555-555-2004<cr>
PV1|1|I|301|R|||1436^Primary^Patricia^P|1026^Admit^Alan|998^Attend^Aaron|MED|||A|4|A0|N|1026^Se
  nder^Sam|OB|H0100240<cr>
```

Figure 5: An HL7 Sample Message ("Admit a Patient" A01)

Each message is created from the following:

1. segments
2. fields
3. components
4. sub-components
5. repeats
6. delimiter characters

The following sections provide more detail on these elements of the V2.x messages.

Segments

An HL7 segment is a logical grouping of data fields. Segments of a message may be **required** or **optional**. They may occur **only once** or they may be **allowed to repeat**. Each segment is identified by a three-character code, known as the Segment ID, and a name. For example, the ADT example message above contains the following segments: Message Header (MSH), Event Type (EVN), Patient ID (PID), Next of Kin (NK1), Patient Visit (PV1) and Insurance (IN1).

Segment ID codes beginning with the letter Z are reserved for locally defined segments. Such constructs are useful for the exchange of information not defined by the standard. However, caution must be exercised when using Z segments and other locally defined structures.

The following table shows the segments that might be transmitted in a typical patient registration message (message/event/trigger A01). This is commonly referred to as the abstract message syntax:

Segments	Description	Chapter
MSH	Message Header	2
[{ SFT }]	Software Segment	2
[UAC]	User Authentication Credential	2
EVN	Event Type	3
PID	Patient Identification	3
[PD1]	Additional Demographics	3
[{ ARV }]	Access Restrictions	3
[{ ROL }]	Role	15
[{ NK1 }]	Next of Kin / Associated Parties	3
PV1	Patient Visit	3
[PV2]	Patient Visit - Additional Info.	3
[{ ARV }]	Access Restrictions	3
[{ ROL }]	Role	15
[{ DB1 }]	Disability Information	3
[{ OBX }]	Observation/Result	7
[{ AL1 }]	Allergy Information	3
[{ DG1 }]	Diagnosis Information	6
[DRG]	Diagnosis Related Group	6
[{	--- PROCEDURE begin	
PR1	Procedures	6
[{ ROL }]	Role	15
}]	--- PROCEDURE end	
[{ GT1 }]	Guarantor	6
[{	--- INSURANCE begin	
IN1	Insurance	6
[IN2]	Insurance Additional Info.	6
[{ IN3 }]	Insurance Additional Info - Cert.	6
[{ ROL }]	Role	15
[{ AUT }]	Authorization Record	11
[{ RF1 }]	Referral Information	11
}]	--- INSURANCE end	
[ACC]	Accident Information	6
[UB1]	Universal Bill Information	6
[UB2]	Universal Bill 92 Information	6
[PDA]	Patient Death and Autopsy	3

Figure 6: HL7 V2.x "Admit a Patient" (A01) Message Structure

Please remember these rules:
If the segment ID appears between square brackets [...] then the segment is optional.
If the segment ID appears between curly braces { … } then the segment may repeat.
If the segment ID appears between both curly braces and square brackets [{ … }] then it is both optional and repeating.

Each segment is composed of a number of fields, which will be discussed later in this Unit.

The first segment (MSH) identifies the type of message and the trigger event that caused the message to be sent. A real-world **event** that causes **a message exchange** is called a "**trigger event**".

In the V2.8 ADT section in Ch. 3, there are 63 different trigger events, such as the admission of the patient, a change in location of the patient or discharge of a patient. For example, a message for trigger event A01 indicates that a new patient has been admitted. The A02 trigger event indicates that the patient has changed location, while trigger event A03 indicates that the patient was discharged, etc.

Figure 7 below shows an example of transactions between two systems using V2.x Chapter 3 messages and trigger events:

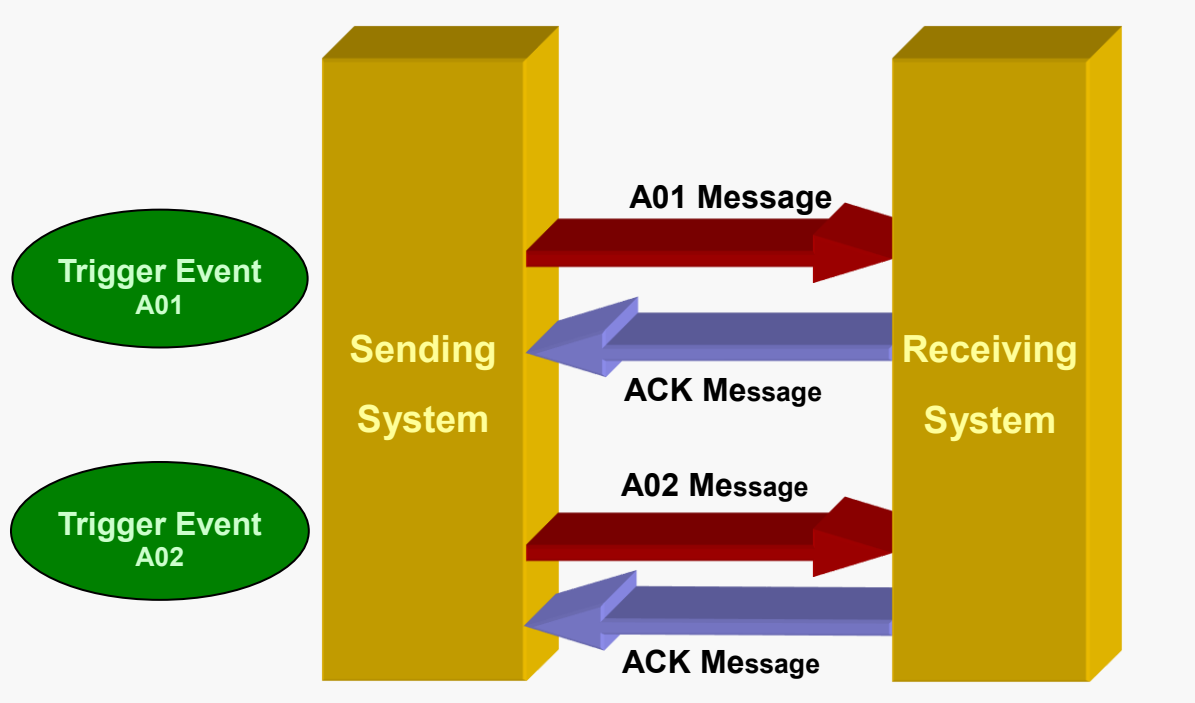

Figure 7: Two typical V2.x Chapter 3 ADT Transactions

When a patient admission is recorded, a message of type ADT event A01 is sent. Likewise, to report a change of patient location, a message of type ADT event A02 is sent.

In both cases, the receiving system returns a general acknowledgment message (ACK).

Usually the information in an ADT message type has been entered into the patient administration system and is then transmitted to other systems throughout the organization.

For example, an A01 event can be used to notify the laboratory system that a patient has been admitted and that studies can be requested for the patient. It also contains the patient's location, sex and age.

If the patient is subsequently transferred to a new permanent location, an A02 trigger event is used to notify the laboratory of the new location, so the lab staff can locate the patient when needing to take a blood sample.

An **abstract message syntax,** such as that shown for the ADT message above, defines which segments make up the message, as well as their order, grouping, and optionality. However, the abstract message syntax does not describe the segment content.

Two or more segments may be organized as a logical unit called a **segment group**. As with individual segments, a segment group may be required or optional. A segment group is assigned a name that represents a permanent identifier that may not be changed; unlike an individual segment ID, the name of a segment group generally does not appear in the HL7 message.

The figure below shows the attributes of the fields in the PID segment, as defined in V2.8:

SEQ	LEN	C.LEN	DT	OPT	RP/#	TBL#	ITEM#	ELEMENT NAME
1	1..4		SI	O			00104	Set ID - PID
2				W			00105	Patient ID
3			CX	R	Y		00106	Patient Identifier List
4				W			00107	Alternate Patient ID - PID
5			XPN	R	Y	0200	00108	Patient Name
6			XPN	O	Y		00109	Mother's Maiden Name
7			DTM	O			00110	Date/Time of Birth
8			CWE	O		0001	00111	Administrative Sex
9				W			00112	Patient Alias
10			CWE	O	Y	0005	00113	Race
11			XAD	O	Y		00114	Patient Address
12				W			00115	County Code
13			XTN	B	Y		00116	Phone Number - Home
14			XTN	B	Y		00117	Phone Number - Business
15			CWE	O		0296	00118	Primary Language
16			CWE	O		0002	00119	Marital Status
17			CWE	O		0006	00120	Religion
18			CX	O		0061	00121	Patient Account Number
19				W			00122	SSN Number - Patient
20				W			00123	Driver's License Number - Patient
21			CX	O	Y	0061	00124	Mother's Identifier
22			CWE	O	Y	0189	00125	Ethnic Group
23		250#	ST	O			00126	Birth Place
24	1..1		ID	O		0136	00127	Multiple Birth Indicator
25		2=	NM	O			00128	Birth Order
26			CWE	O	Y	0171	00129	Citizenship
27			CWE	O		0172	00130	Veterans Military Status
28				W			00739	Nationality
29			DTM	O			00740	Patient Death Date and Time
30	1..1		ID	O		0136	00741	Patient Death Indicator
31	1..1		ID	O		0136	01535	Identity Unknown Indicator
32			CWE	O	Y	0445	01536	Identity Reliability Code
33			DTM	O			01537	Last Update Date/Time
34			HD	O			01538	Last Update Facility
35			CWE	O			01539	Taxonomic Classification Code
36			CWE	B		0447	01540	Breed Code
37		80=	ST	O			01541	Strain
38			CWE	O	2	0429	01542	Production Class Code
39			CWE	O	Y	0171	01840	Tribal Citizenship

SEQ	LEN	C.LEN	DT	OPT	RP/#	TBL#	ITEM#	ELEMENT NAME
40			XTN	O	Y		02289	Patient Telecommunication Information

Figure 8: The "Patient Information" PID segment Definition

Each column in the table above defines a specific attribute of the segment:

SEQ - Ordinal position of the field within the segment
LEN - Maximum number of characters that the field may contain
C.LEN - Conformance length
DT - Data type defined by HL7
OPT - Optionality (R-required, O-optional, C-conditional, X-discontinued, B- maintained for backward compatibility with previous versions)
RP/# - Repetition and maximum number of occurrences allowed
TBL# - Reference to a table of possible values, usually in Ch. 2C
ITEM - an integer number that uniquely identifies the field / data item
ELEMENT NAME - the names of the field / data item

Fields

A field is a string **of characters defined by an HL7 data type**. Appendix A of the HL7 V2.x standards provides an alphabetical list of all fields and the data types were listed in Ch. 2 up to V2.6 and in Ch. 2A from V2.7 onwards.

Components

Depending on its data type, a field can have one or more components. A model of the components of each field in a segment is contained in the chapter of the HL7 standard in which the segment is defined. Figure 9 below shows the 22 components of the element "Citizenship" (PID-26, data type CWE) as defined in HL7 V2.8:

3.3.2.26 PID-26 Citizenship (CWE) 00129¶

Components: <Identifier (ST)> ^ <Text (ST)> ^ <Name of Coding System (ID)> ^ <Alternate Identifier (ST)> ^ <Alternate Text (ST)> ^ <Name of Alternate Coding System (ID)> ^ <Coding System Version ID (ST)> ^ <Alternate Coding System Version ID (ST)> ^ <Original Text (ST)> ^ <Second Alternate Identifier (ST)> ^ <Second Alternate Text (ST)> ^ <Name of Second Alternate Coding System (ID)> ^ <Second Alternate Coding System Version ID (ST)> ^ <Coding System OID (ST)> ^ <Value Set OID (ST)> ^ <Value Set Version ID (DTM)> ^ <Alternate Coding System OID (ST)> ^ <Alternate Value Set OID (ST)> ^ <Alternate Value Set Version ID (DTM)> ^ <Second Alternate Coding System OID (ST)> ^ <Second Alternate Value Set OID (ST)> ^ <Second Alternate Value Set Version ID (DTM)>¶

This field contains the information related to a person's country citizenship. For country citizenship HL7 recommends using ISO table 3166. For a local definition, *User-defined Table 0171 - Citizenship* in Chapter 2C, Code Tables, should be used.¶

Figure 9: Definition of the Components of the "Patient Name" element

Subcomponents

When the component contains multiple parts, each of its parts is called a "subcomponent".

Message Delimiters

In constructing a message, certain special characters are used. The recommended delimiters are:

Segment Terminator	<CR>	(ASCII 13)
Field separator	|	(ASCII 124)
Component Separator	^	(ASCII 94)
Subcomponent Separator	&	(ASCII 38)
Repetition Separator	~	(ASCII 126)
Escape Character	\	(ASCII 92)
Truncation character	#	(ASCII 35)

Note: With the exception of the Segment Terminator <CR>, these delimiters are defined by the first two fields of the MSH segment.
This means that all message content delimiters can be re-defined.
Careful, not all HL7 implementations support this feature!

Summary

A message consists of segments separated from each other by the segment terminator (always <CR>). Each **segment** consists of fields separated by the field separator (usually the pipe or vertical bar character "|"). Each **field** is composed of one or more components separated by the component separator (usually the caret character "^") and the raw content of each **component** is defined by a specific data type. Depending on its data type, a component can contain one or more **subcomponents** separated by the subcomponent separator (usually the ampersand character "&").

The next section will address the data types that define components and subcomponents within fields.

6. The V2.x Data Types

Each HL7 element is defined as having a data type; the HL7 V2.8 standard defines over 90 different data types.

A data type may have a single component or may be defined as a set of components. The purpose of a data type is to constrain the contents of a field.

In the segment attribute table the data type defining the structure of each field is indicated in the DT column.

Data Types Overview

This is a list of the HL7 V2.8 Data types grouped into categories:

- String (ST TX FT)
- Numerical (CQ MO NM SI SN)
- Identifiers (ID IS HD EI RP PL PT VID)
- Date / Time (DTM DT TM)
- Coded values (CF CK CX XCN CNE CWE)
- Generic (CM)
- Demographic (XAD XPN XON XTN SAD FN)
- Waveforms (CD MA NA ED)
- Price (CP)
- Finance (FC)
- Master File tables (DLN JCC VH)
- Medical records (PPN)
- Temporary series (DR RI SCV TQ)

Data Types for Strings and Text

These data types are for communicate textual information using any alphanumeric, ASCII printable characters:

Data Type	Description
ST - String	Left justified, trailing spaces optional. Usually used for short strings: Longer strings or formatted for user display will usually be of data type TX or FT. Example: …|This is a short text|…

Data Type	Description
TX - Text	Intended to be shown to the user through a terminal or a printer. This data type allows leading spaces to improve presentation. It can be long (up to 64 K). Example: …| This is a short indented text|…
FT - Formatted Text	FT allows the communication of formatted text, using some of the following codes: **.sp #** Ends the present line and skips the number # of lines **.br** Begin new line **.fi** Begin word wrap **.nf** Ends word wrap **.in #** Indent # of spaces **.ti #** Temporary indent # of spaces **.ce** End the present line and center the following line Example: the FT text Observation:\.sp\\.in+4\\.ti-4\1. The cardio mediastinal silhouette is now within normal limits.\.sp\\.ti-4\2. Lung fields show minimal ground glass appearance. will be displayed as: Observation: 1. The cardio mediastinal silhouette is now within normal limits. 2. Lung fields show minimal ground glass appearance.

Data Types for Numbers and Quantities

The data types in the following table are used to express numbers and quantities:

Data Type	Description
SI - Sequence Identifier	This data type is a positive whole number. It is used to identify the ordinal position of a repeating segment within a message. Example: |OBX|4|…
NM - Numeric	This data type allows zero or more ASCII numeric characters as well as an optional + or - sign and an optional decimal point Examples: …|1234|… …|+1234|… …|-123.4567|…

Data Type	Description
SN - Structured Numeric	This data type is often used to communicate clinical results with some type of qualifier. The structure is as follows: ***<Comparator (ST)> ^ <Number 1 (NM)> ^ <Separator/ Suffix (ST)> ^ <Number 2 (NM)>*** Comparators can be "=", ">", "<", ">=", "<=" or "<>" Separators can be "- ", "+", "/", ". ", or ": " Examples: Greater than 100: …|>^100|… Between 50 and 120: …|^50^-^120|… Serology titration of 1:256: …|^1^:^256|… Occult blood positivity: …|^3^+|…
CQ - Composite Quantity with units	This data type has two components, the first for the amount and the second for the units of measure: ***<Quantity (NM)> ^ <Units (CE)>*** Example: a patient weighs 123.7 Kilograms: …|123.7^kg|…
MO - Money	This data type is specifically for communicating currency amounts and has two components. The first one is the amount and the second is the denomination of currency: ***<Quantity (NM)> ^ <Denomination (ID)>*** Example: a medicine costs €79.50 euros[2]: …|79.50^EUR|…

Data Types for Times and Dates

These data types communicate dates (date of birth, date of death, etc.), times (appointments, specimen retrieval, test reports, etc.) and time stamps (message creation, transactions, etc.):

Data Type	Description
DT - Date	This data type is specifically for communicating dates; the structure is: ***YYYY[MM[DD]]*** The portions in square brackets (i.e. month-day and the day) are optional. Examples: October 15, 2006: …|20061015|… October 2006: …|200610|… Year 2006: …|2006|…

[2] See list of currency codes: http://en.wikipedia.org/wiki/ISO_4217

Data Type	Description
TM - Time	This data type is used to communicate time, without the date. The structure is: ***HH[MM[SS[.S[S[S[S]]]]]][+/-ZZZZ]*** The digits symbolized by HH indicate hours, MM indicates minutes and SS.SSSS indicate seconds and fractions of seconds. +ZZZZ or -ZZZZ designates the time zone offset from UTC (formerly Greenwich Mean Time). Examples: 10.45am in the GMT+11 time zone: …|104500+1100|… 20.15 in the evening, local time: …|2015|… 7:35am and 44.231 seconds, local time: …|073544.231|… Noon, local time: …|1200|…
DTM - Date/Time	This data type combines date and time. Its structure is: ***YYYY[MM[DD[HH[MM[SS[.S[S[S[S]]]]]]]]][+/-ZZZZ]*** Example: 1:15am on July 4, 1976, local time: …|197607040115|… ***Note: the "TS" data type was made obsolete in V2.6!***
DR - Date/Time Range	This data type is used to communicate a duration of time; it is a combination of two DTM data types. ***<Start Time (DTM)> ^ <End Time (T=DTM)>*** Example: 9am to 5pm on 21 October 2015: …|201510210900^201510211700|…

Data Types for Identifiers

These data types communicate alphanumeric strings that identify entities and objects:

Data Type	Description
ID - Coded Value (from HL7-defined Table)	This data type communicates a value from an HL7-defined table listed in a segment definition's TBL# column. Example: HL7 Table 0027 "Priority" **S** Stat (do immediately) **A** As soon as possible (a priority lower than stat) **R** Routine **P** Preoperative (to be done prior to surgery) **T** Timing critical (do as near as possible to requested time) Do test as soon as possible: …|A|…

Data Type	Description
IS - Coded Value (from user-defined Table)	This data type communicates a value from a user-defined table listed in a segment definition's TBL# column. Example: User Table 0004 "Patient Class" **E** Emergency **I** Inpatient **O** Outpatient **P** Preadmit **R** Recurring patient **B** Obstetrics … Patient is an Outpatient: …|O|…
HD - Hierarchic Designator	This type of identifier is usually for applications and assigning authorities. Its structure is: ***<Namespace ID (IS)> ^ <Universal ID (ST)> ^ <Universal ID Type (ID)>*** The first part is an entry from a user-defined table and the second is a string formatted according to the rules of the ID type indicated in the third component. If the identifier is for a locally defined entity, only the Namespace ID is required. Example: …|GHH_Clinic|… If the identifier is for a global entity, a Universal ID unique within the domain of the Universal ID Type (e.g. DNS, GUID, etc.) is communicated. Example: …|^HL7.org^DNS|…
EI - Entity Identifier	This data type expresses identifiers assigned by an application. Its structure is: ***<Entity Identifier (ST)> ^ <Namespace ID (IS)> ^ <Universal ID (ST)> ^ <Universal ID Type (ID)>*** The first part is the identifier (lab accession number, Order Number, Inpatient Encounter Number, Authorization Number, Facility Identifier, etc.) and the remaining parts identify the authority assigning the identifier. Example ID 125658 assigned by the hospital information system of the Italian Hospital of Buenos Aires: …|125658^HIS_HIBA|…

Data Types for Coded Values

These data types communicate strings that identify entities and objects from code tables:

Data Type	Description
CNE - Coded Data (no exceptions)	This data type communicates values that must only be selected from the coding system designated in the element definition without exception. The designated coding system may not be locally extended and its codes may not be redefined. CNE is similar to the CE data type that was obsoleted in V2.6 but includes versioning and the original text entered into the system. Its structure is: ***<Identifier (ST)> ^ <Text (ST)> ^ <Name of Coding System (ID)> ^ <Alternate Identifier (ST)> ^ <Alternate Text (ST)> ^ <Name of Alternate Coding System (ID)> ^ <Version ID (ST)> ^ <Alternate Coding System Version ID (ST)> ^ <Original Text (ST)>*** Example coding Hemoglobin in LOINC V1.0: …|718-7^Hemoglobin^LN^^^^1.0|…
CWE - Coded Data (with exceptions)	This data type communicates values that may be selected from the coding system designated in the HL7 element definition. Locally defined coded may be sent, but existing codes may not be redefined. CWE is also similar to the CE data type that was obsoleted in V2.6 but includes versioning and the original text entered into the system. Its structure is the same as the CNE.

Data Types for Persons, Organizations, Addresses and Phone Numbers

These data types communicate the details of persons, patients and organizations as well as their contact details (physical addresses, e-mail addresses, phone number(s), etc.):

Data Type	Description
XPN - Person Name	This data type communicates patients and persons. Its structure is: ***<Family Name (FN)> ^ <Given Name (ST)> ^ <Second and Further Given Names or Initials Thereof (ST)> ^ <Suffix (e.g., JR or III) (ST)> ^ <Prefix (e.g., DR) (ST)> ^ <WITHDRAWN Constituent> ^ <Name Type Code (ID)> ^ <Name Representation Code (ID)> ^ <Name Context (CWE)> ^ <WITHDRAWN Constituent> ^ <Name Assembly Order (ID)> ^ <Effective Date (DTM)> ^ <Expiration Date (DTM)> ^ <Professional Suffix (ST)> ^ <Called By (ST)>***

Data Type	Description
	The parts of the Family Name (FN) are: ***<Surname (ST)> & <Own Surname Prefix (ST)> & <Own Surname (ST)> & <Surname Prefix from Partner/Spouse (ST)> & <Surname from Partner/Spouse (ST)>***
	The parts of the Name Context (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	Example Charlie C. Chopper, MD: …|Chopper^Charlie^C^^MD|…
XCN - Person Name and ID	This data type communicates the identity of patients and persons. Its structure is: ***<Person Identifier (ST)> ^ <Family Name (FN)> ^ <Given Name (ST)> ^ <Second and Further Given Names or Initials Thereof (ST)> ^ <Suffix (e.g., JR or III) (ST)> ^ <Prefix (e.g., DR) (ST)> ^ ^ [3]^ <Assigning Authority (HD)> ^ <Name Type Code (ID)> ^ <Identifier Check Digit (ST)> ^ <Check Digit Scheme (ID)> ^ <Identifier Type Code (ID)> ^ <Assigning Facility (HD)> ^ <Name Representation Code (ID)> ^ <Name Context (CWE)> ^ ^ <Name Assembly Order (ID)> ^ <Effective Date (DTM)> ^ <Expiration Date (DTM)> ^ <Professional Suffix (ST)> ^ <Assigning Jurisdiction (CWE)> ^ <Assigning Agency or Department (CWE)> ^ <Security Check (ST)> ^ <Security Check Scheme (ID)>***
	The parts of the Family Name (FN) are: ***<Surname (ST)> & <Own Surname Prefix (ST)> & <Own Surname (ST)> & <Surname Prefix from Partner/Spouse (ST)> & <Surname from Partner/Spouse (ST)>***

[3] Note that these two parts of the data type are obsolete

Data Type	Description
	The parts of the Assigning Authority (HD) are: ***<Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>***
	The parts of the Assigning Facility (HD) are: ***<Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>***
	The parts of the Name Context (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Assigning Jurisdiction (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***

Data Type	Description
	The parts of the Assigning Agency or Department (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	Example Charlie C. Chopper, MD, ID# 765432: …|765432^Chopper^Charlie^C^^MD|…
XON - Organization Name and ID	This data type is used to identify an organization and its associated ID. Its structure is: ***<Organization Name (ST)> ^ <Organization Name Type Code (CWE)> ^ ^ ^ ^ <Assigning Authority (HD)> ^ <Identifier Type Code (ID)> ^ <Assigning Facility (HD)> ^ <Name Representation Code (ID)> ^ <Organization Identifier (ST)>***
	The parts of the Organization Name Type Code (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Assigning Authority (HD) are: ***<Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>>***

Data Type	Description
	The parts of the Assigning Facility (HD) are: ***<Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>***
	Example Goodhealth Hospital , assigned by CMS: …|Good Health Hospital^L^^^^CMS^^^A|…
XAD - Address	This data type is used to communicate the physical address of a person, patient or organization. Its structure is: ***<Street Address (SAD)> ^ <Other Designation (ST)> ^ <City (ST)> ^ <State or Province (ST)> ^ <Zip or Postal Code (ST)> ^ <Country (ID)> ^ <Address Type (ID)> ^ <Other Geographic Designation (ST)> ^ <County/Parish Code (CWE)> ^ <Census Tract (CWE)> ^ <Address Representation Code (ID)> ^ <WITHDRAWN Constituent> ^ <Effective Date (DTM)> ^ <Expiration Date (DTM)> ^ <Expiration Reason (CWE)> ^ <Temporary Indicator (ID)> ^ <Bad Address Indicator (ID)> ^ <Address Usage (ID)> ^ <Addressee (ST)> ^ <Comment (ST)> ^ <Preference Order (NM)> ^ <Protection Code (CWE)> ^ <Address Identifier (EI)>***
	The parts of the Street Address (SAD) are: ***<Street or Mailing Address (ST)> & <Street Name (ST)> & <Dwelling Number (ST)>***
	The parts of the County/Parish Code (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***

Data Type	Description
	The parts of the Census Tract (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Expiration Reason (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Address Identifier (EI) are: ***<Entity Identifier (ST)> & <Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>***
	Example 14th Floor, 150 Hospital Road, Sydney, NSW, 2000: …|14th Floor^150 Hospital Road^Sydney^NSW^2000|…

Data Type	Description
XTN - Telephone Number	This data type is used to communicate the contact details (phone number(s), e-mail address, pager, etc.) of a person, patient or organization. Its structure is: ***^ <Telecommunication Use Code (ID)> ^ <Telecommunication Equipment Type (ID)> ^ <Communication Address (ST)> ^ <Country Code (SNM)> ^ <Area/City Code (SNM)> ^ <Local Number (SNM)> ^ <Extension (SNM)> ^ <Any Text (ST)> ^ <Extension Prefix (ST)> ^ <Speed Dial Code (ST)> ^ <Unformatted Telephone number (ST)> ^ <Effective Start Date (DTM)> ^ <Expiration Date (DTM)> ^ <Expiration Reason (CWE)> ^ <Protection Code (CWE)> ^ <Shared Telecommunication Identifier (EI)> ^ <Preference Order (NM)>***
	The parts of the Expiration Reason (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Protection Code (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***

Data Type	Description
	The parts of the Protection Code (CWE) are: ***<Identifier (ST)> & <Text (ST)> & <Name of Coding System (ID)> & <Alternate Identifier (ST)> & <Alternate Text (ST)> & <Name of Alternate Coding System (ID)> & <Coding System Version ID (ST)> & <Alternate Coding System Version ID (ST)> & <Original Text (ST)> & <Second Alternate Identifier (ST)> & <Second Alternate Text (ST)> & <Name of Second Alternate Coding System (ID)> & <Second Alternate Coding System Version ID (ST)> & <Coding System OID (ST)> & <Value Set OID (ST)> & <Value Set Version ID (DTM)> & <Alternate Coding System OID (ST)> & <Alternate Value Set OID (ST)> & <Alternate Value Set Version ID (DTM)> & <Second Alternate Coding System OID (ST)> & <Second Alternate Value Set OID (ST)> & <Second Alternate Value Set Version ID (DTM)>***
	The parts of the Shared Telecommunication Identifier (EI) are: ***<Entity Identifier (ST)> & <Namespace ID (IS)> & <Universal ID (ST)> & <Universal ID Type (ID)>***
	Examples: Work Phone: …|^WPN^PH^^61^403^116367… Work Fax: …|^WPN^FX^^1^734^6777777|… e-Mail: …|^WPN^Internet^someone@somecorp.com|…

Data Types for Multimedia Objects

These data types communicate binary objects (e.g. photos, images, sound files, etc.) either within the messages or referenced via a pointer, Uniform Resource Locator (URL) or Uniform Resource Identifier (URI):

Data Type	Description
ED - Encapsulated Data	This data type communicates base64-encoded[4] binary data in the message. Its structure is: ***<Source Application (HD)> ^ <Type of Data (ID)> ^ <Data Subtype (ID)> ^ <Encoding (ID)> ^ <Data (TX***

[4] base64 allows binary data to be communicated in an ASCII string format by translating it into printable characters. For more information see http://en.wikipedia.org/wiki/Base64

Data Type	Description
RP - Reference Pointer	This data type communicates information on "multimedia objects" stored in another system's repository. Its structure is: ***<Pointer (ST)> ^ <Application ID (HD)> ^ <Type of Data (ID)> ^ <Subtype (ID)>*** The first component indicates the resource location and the second identifies the application responsible for the information. The third and fourth components define the type and subtype of the information. Example: GIF Image …|http://www.hl7.org/images/1232.gif^HL7^image^GIF|…

Note that the data types listed here are those most commonly used in HL7 V2.x messages; for the full list see HL7 V2.8 Chapter 2A.

7. HL7 V2.x Message Processing Rules

The creation and processing of HL7 V2.x messages follows a strict set of rules that are described in Chapter 2 of the V2.x standards.

HL7 V2.x Outbound Message Creation Rules

The Ten Outbound HL7 Message Creation Commandments:

1. *Assemble the message in the order specified in the abstract message format*
2. *The 3-character segment ID must be at the beginning of EACH segment*
3. *Precede each element / data field with the data field delimiter, usually "|"*
4. *Add the data fields in the order specified in the segment definition table*
5. *Data fields 'not present' require no characters.*
6. *Data fields 'present but null' are encoded with "" (double quotes)*
7. *If components, subcomponents or repetitions at the end of a data field are 'not present' their delimiters may be omitted*
8. *If no more fields are present at the end of a segment, the data field delimiters may be omitted*
9. *'Padding' does not violate these rules, but it is not good practice and leads to un-necessary processing and message length.*
10. *End each segment with the segment terminator, always "<cr>"*

HL7 V2.x Inbound Message Creation Rules

The Three Inbound HL7 Message Parsing Commandments:

1. *Ignore the Unexpected! Segments, fields, components, subcomponents and repetitions that are present but not expected - they are not your Problem!!!*
2. *If it's not there, assume it's 'not present'. Treat segments, fields, components, subcomponents and repetitions that are expected but not received as not present - take no action.*
3. *Required fields that are missing are serious errors and you must reject the message and retuned info on the error in the ERR segment.*

Acknowledgement of Message Reception (ACK)

In a HL7 V2.x transaction, the Sending System sends a message, which is received by the Receiving System. The Receiving System then processes the message and sends a response back:

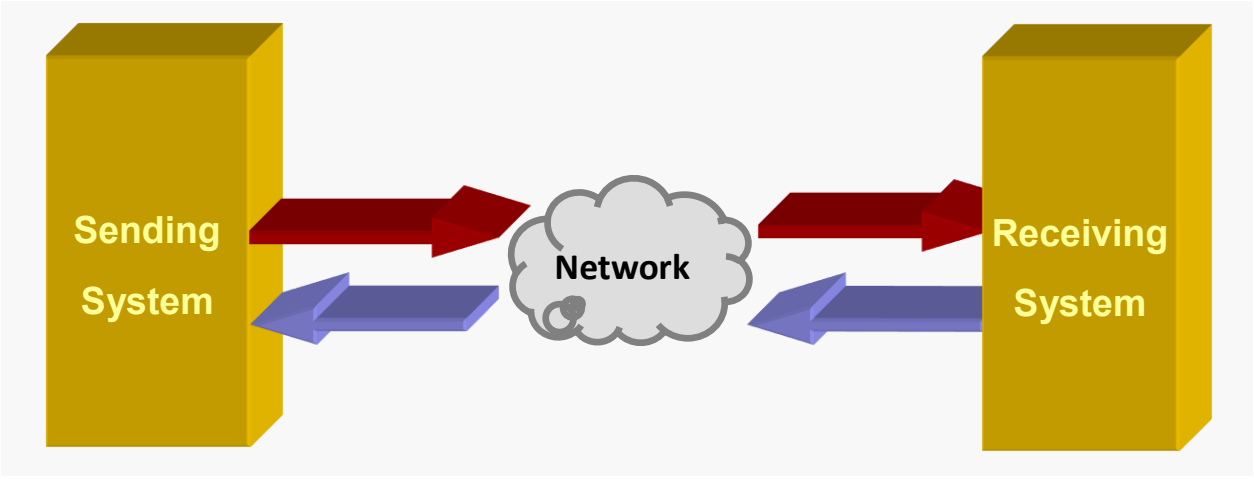

Figure 10: Basic Model of an HL7 V2.x Transaction

The response from the Receiving System is called an **acknowledgment message**. HL7 V2.x supports two modes of acknowledgment that may be used:

- Original Acknowledgement mode
- Enhanced Acknowledgement mode

Original Acknowledgement Mode

In the Original Acknowledgement mode, the successful reception, storage and processing of messages are together acknowledged by a single acknowledgment message from the receiving application. The rationale is that for reliable operation it is necessary to know that the underlying communications system has delivered the message to the receiver and the receiving application has processed the data successfully.

Because the receiving application acknowledges a completed transaction, the Original Acknowledgement allows reliable message exchange with a single message pair:

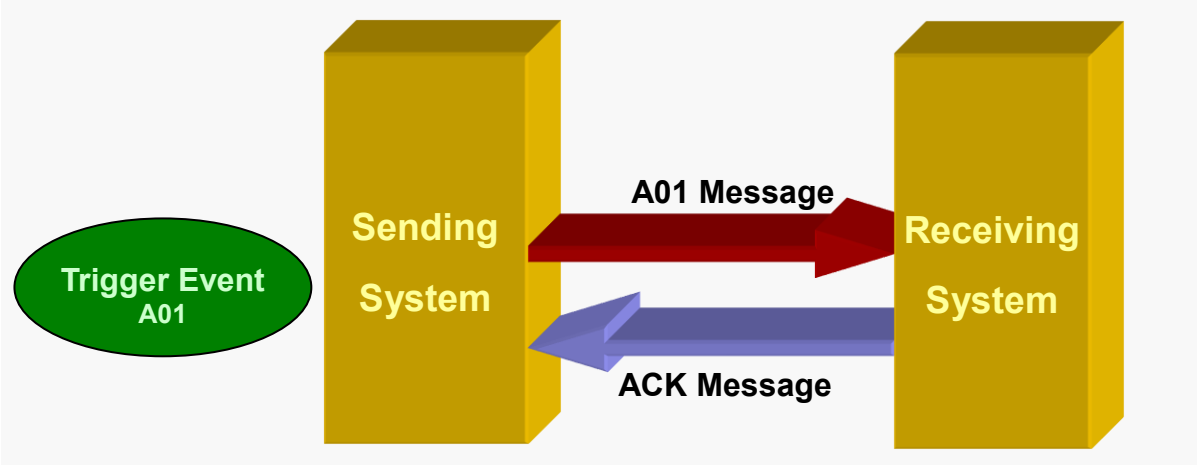

Figure 11: Transaction Flow - Original Acknowledgement Mode

In some cases (ORR, RF1, etc.), the acknowledgment message may also contain data (lab number, referral accept #, etc.) that are important to the system that initiated the exchange.

For example, a patient care provider wants to order a lab test and sends an ORM message to a lab application identifying the patient, the test ordered and other information about the order.
The lab application will acknowledge the order after it has been successfully processed the request and created a lab order number ("lab accession number"). It will do so by means of an ORR application acknowledgment message that not only acknowledges receipt of the order message but also includes an order status and lab order number in the ORC segment. (The receiving system could also send back the ORR application acknowledgment message immediately after putting the lab order in a queue and guaranteeing that the order will be processed within a reasonable time.).

A successful processing of an incoming message is acknowledged by the receiver with an application acknowledgement ("AA"). If there were errors in the message content (segments out of sequence, mandatory/required element missing, wrong data type, coded value not in table, etc.) then an application error ("AE") is returned to the sender. If there were problems unrelated to the message content (can't process MSH-9 message type, wrong HL7 version, system down, internal systems error, etc.) an application reject ("AR") error message is returned to the sender.

However, the disadvantage of the Original Acknowledgement mode is that if the receiving application cannot respond, the sending application assumes that the receiving application has not received the message and will repeatedly re-send the message.
This can result in communication blocks and/or huge message queues.

The Enhanced Acknowledgment Mode solves the problem of unresponsive applications as it splits the communications and processing parts of a transaction.

Enhanced Acknowledgement Mode

The Enhanced Acknowledgment Mode, added in V2.2, separates acknowledgements into two phases:

1. The receiving system acknowledges that the message has been received and it is acceptable for processing
2. The receiving application acknowledges that the received message has been processed and acted on.

As a result, one transaction (e.g. Admit a Patient, Order Laboratory Tests, etc.) consists of two phases and two message pairs:

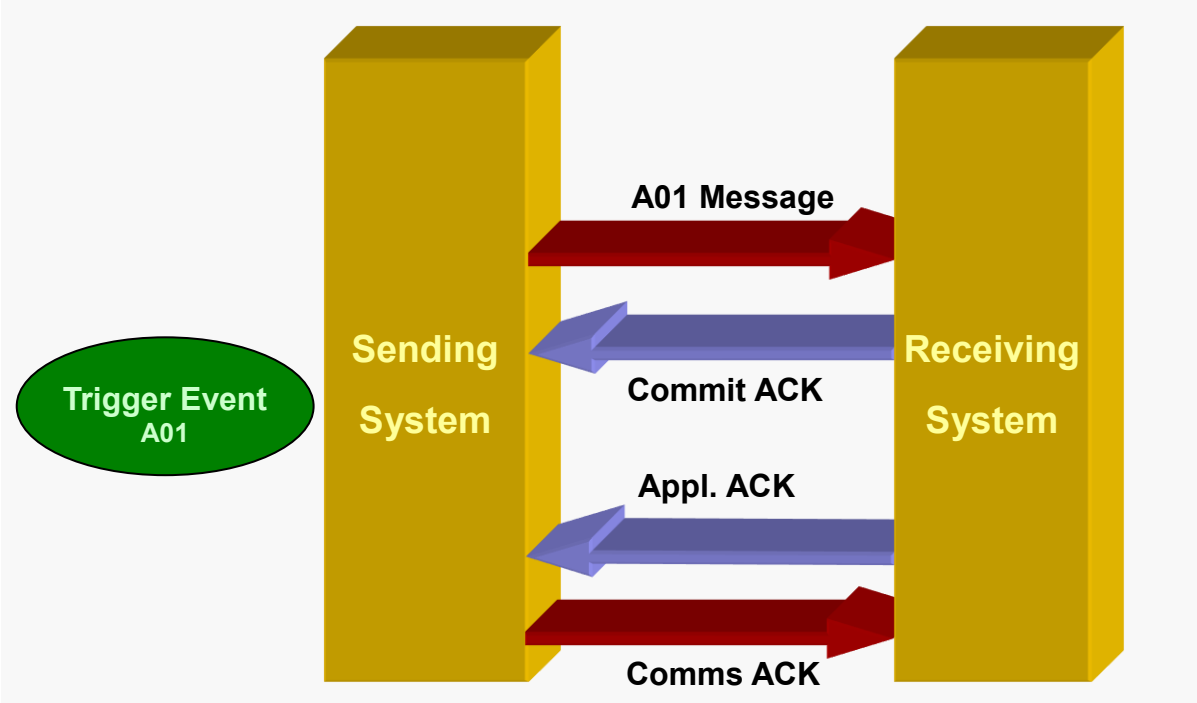

Figure 12: Transaction Flow - Enhanced Acknowledgement Mode

Phase 1: After receiving the Admit a Patient (A01) message from sending system, the receiving system commits the message to safe storage in a manner that releases the sending system from the need to resend the message and returns a positive accept acknowledgment.

A successful acceptance of an incoming message is acknowledged by the receiver with a commit acknowledgement ("CA"). If there were problems unrelated to the message content (can't process MSH-9 message type, wrong HL7 version, system down, internal systems error, etc.) a commit reject ("CR") error message is returned to the sender. If there were other problems (sequence number error, etc.) a commit error ("CE") message is returned to the sender.

Later the receiving system then proceeds to process the acknowledged message.

Phase 2: After the message has been processed by the receiving system at the application level, an application acknowledgment is sent to the sending system.

If there were errors in the message content (segments out of sequence, mandatory/required element missing, wrong data type, coded value not in table, etc.) then an application error ("AE") is returned to the sender. If there were problems unrelated to the message content an application reject ("AR") error message is returned to the sender. Otherwise, the successful processing of the message is acknowledged by the receiver with an application acknowledgement ("AA").

Implementing the Enhanced Acknowledgment Mode in V2.x messaging ensures that the transactions have been communicated while avoiding any issues with communication blocks and/or huge message queues.

Unit Summary and Conclusion

In this Unit, you have learned the basics of HL7's first standard, Version 2. You now know the history of the Health Level 7 (HL7) standards development organization (SDO) as well as the other standards that HL7 has published.

You also now know the main features of the various versions of the V2.x standards, an overview of the ADT and Orders/Results messages as well as the main data types. The two message acknowledgement methods have also been outlined.

In next Units of this Module, you will study ADT and Orders/Results messages in details, learn the principles and pitfalls of HL7 implementation project management and know how to encode V2.x messages in XML rather than "pipe & hat" or "stick & caret" formats.

Additional Reference Material

HL7 Version 2.2 (ANSI/HL7 V2.2-1996)
HL7 Version 2.3 (ANSI/HL7 V2.34-1997)
HL7 Version 2.3.1 (ANSI/HL7 V2.3.1-1999)
HL7 Version 2.4 (ANSI/HL7 V2.4-2000)
HL7 Version 2.5 (ISO/HL7 27931, ANSI/HL7 V2.5-2003)
HL7 Version 2.5.1 (ANSI/HL7 V2.5.1-2007)
HL7 Version 2.6 (ANSI/HL7 V2.6-2007)
HL7 Version 2.7 (ANSI/HL7 V2.7-2011)
HL7 Version 2.7.1 (ANSI/HL7 V2.7.1-2012)
HL7 Version 2.8 (ANSI/HL7 V2.8-2014)

HL7 Version 2.x XML Encoding Rules, Release 1 (ANSI/HL7 V2 XML-2003)
HL7 Version 2.x XML Encoding Rules, Release 2 (ANSI/HL7 V2 XML, R2-2012)

Note: You can download the HL7 V2.x standards free from the Health Level 7 International web site (www.HL7.org) after you create a free log-in account.