

SALIENT FEATURES

EV1000 Clinical Platform Product Specifications

Power/Electrical	AC Mains: 100-240 VAC, 50/60 Hz 24V power adapter / 1.5 A Max
Color Monitor	800 x 600 / 10.4 in TFT, 230 cd/m brightness 500:1 contrast ratio
Measuring Time	CO updated every 20 seconds or 5 minutes ScvO ₂ , updated every 2 seconds
Trend Range	0.1 – 72 hours
Monitor Dimensions	Height: 226 mm Width: 296 mm Depth: 58.6 mm
Monitor Weight	2.1 kg (4.6 lb)
Monitor Display	Active Area: 257 mm (10.4 in) Resolution: 800 x 600 LCD
Monitor Operating System	Windows XPe
Monitor Speaker Count	2
Databox Dimensions	Height: 145 mm Width: 206 mm Depth: 64 mm
Databox Weight	0.9 kg (2.0 lb)

NOTE: THE EQUIPMENT MUST BE ABLE TO INTERFACE WITH CPRS VIA HL7 PROTOCOL.

IT ALSO MUST BE ABLE TO INTERFACE WITH IFM SERIAL CONNECTION.


[Products](#) [Education](#) [Resources](#) [Careers](#) [Investors](#) [Contact Us](#)

SEARCH

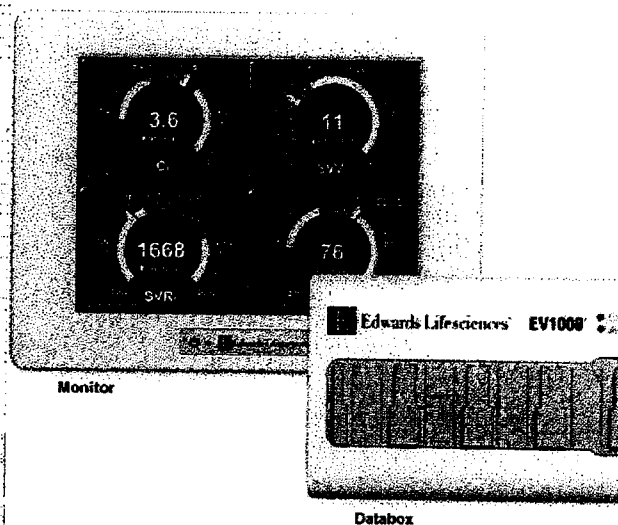

[My Account](#) [My Cart](#)

[ccNextin System](#)
[FloTrac Sensor](#)
[VolumeView Set](#)
[Swan-Ganz Catheters](#)
[PreSep Catheter](#)
[PediaSat Catheter](#)
[EV1000 Clinical Platform](#)
[Vigilao Monitor](#)
[Critical Care Education](#)
[Healthcare Provider Solutions](#)
[Product Catalog](#)
[Request Information](#)
[Reference Links](#)
[CCContinuum Educational Review](#)
[Reimbursement](#)
[You Tube](#) [Edwards Lifesciences](#)
[You Tube](#) [Edwards Critical Care Education](#)

[Products](#) / [Advanced Hemodynamic Monitoring](#) / [EV1000 Clinical Platform](#) / [EV1000 Platform](#)

Advanced Hemodynamic Monitoring

The EV1000 Clinical Platform



Product Details

- Animation
- Brochure
- Connectivity
- Models
- Specifications
- Screens
- **The Platform**
- EV1000 & FloTrac Sensor System Setup
- EV1000 & VolumeView Sensor System Setup

Enhanced Surgical Recovery

- Perioperative Goal-Directed Therapy
- Hemodynamic Optimization Benefits

Educational Materials

- eLearning Module
- Quick Guide 2nd Edition
- Physiologic Implications of Appropriate Resuscitation
- Fluid Responsiveness in the Critically Ill Patient

Related Products

- FloTrac Sensor
- PreSep Catheter
- PediaSat Catheter

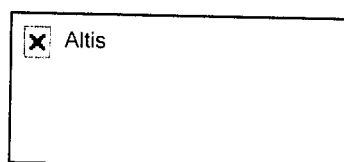


Experience the
Edwards Critical Care System
MICROSITE

CAUTION: Federal (United States) law restricts this device to sale by or on the order of a physician. See instructions for use for full prescribing information, including indications, contraindications, warnings, precautions and adverse events.

FloTrac EV1000 System:

The FloTrac EV1000 system provides hemodynamic monitoring that can improve the care of our critically ill patients. The system is a significant improvement from our current capabilities. Currently, our hemodynamic monitoring is limited to pulmonary artery catheters or echocardiography. This system is significantly less invasive than placement of a pulmonary artery catheter, as it only requires placement of an arterial line or a central venous line which are routine ICU procedures. Echocardiography is non-invasive, but does not provide continuous information as this system does. The FloTrac system will provide us with a minimally invasive way of obtaining continuous monitoring to optimize the resuscitation of patients with shock. This will be widely applicable in our patient population where we frequently encounter septic shock, hemorrhagic shock, and cardiogenic shock.



- [home](#)
- [who we are](#)
- [what we do](#)
- [resources](#)
- [quick consultant](#)
- [contact](#)
- [news](#)

[Building Snowmobiles](#)

 Search

What is HL7?

Overview—Interface Protocols

HL7 is an example of a set of communications rules called a protocol. Interface protocols define the rules by which two applications will exchange data in an orderly way. The following, more detailed definition describes the key attributes of an interface protocol:

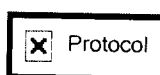


Figure: Interface Protocol Defined

While HL7 has become the dominant interface protocol in healthcare, proprietary protocols still exist. If the particular variant of a proprietary scheme can meet the definition of an interface protocol, integrating that application can be considered possible. Whether or not it would be practical is a matter of further analysis in that case.

HL7 Background

Over the past decade or so, application integration using the Health Level 7 (HL7) protocol has gained wide acceptance and support. This discussion focuses on pre-3.0 versions of HL7. HL7 3.0 represents a significant technical departure and will be treated specifically in a separate article. Before HL7, interfacing two healthcare applications involved a negotiation that had virtually no ground rules. Details as minor as record separators could be a source of disagreement and delay. Vendor preference more than customer requirements tended to influence the outcome. HL7 emerged as a response, providing a common vocabulary and architecture to tie the analysis/negotiation process much more closely to customer requirements.

For the customer, prescribing HL7 as a healthcare messaging standard establishes the baseline architectural rules for integration and the vocabulary for accomplishing it. For the software developer, HL7 provides a necessary framework, but the HL7 specification itself does not define canonical rules that allow a "standard" implementation. While HL7 implies a data model and business rules, it is not based on an explicit reference architecture, so compliance testing is not practical. (This has been a conscious trade-off within the HL7 community.) Some natural requirements and questions flow from decisions to use HL7 to support an integration approach:

Customer:

- What integration capabilities must be expected of a supplier?
- What is the most effective way to establish and articulate our requirements in an HL7 context?

Supplier:

- How do we establish and manage the boundary between application functions and message functions?
- How do we support HL7 messaging as a general approach without having to program support for all possible combinations of transactions (including ones that are almost never used)?

Rather than present exclusive views, these questions interlock to form a requirements-based conversation between customers and suppliers. The most common manifestation of this conversation is the interface negotiation that establishes how a given interface will actually be implemented.

HL7 neither intends nor necessarily recommends that a complete implementation of an interface must include support for the entire hierarchy of protocol features. The answer to the question "Which transactions must I support?" is normally "Support the ones that you need." The HL7 specification allows tremendous flexibility for a developer working on a specific implementation by supporting many common requirements. For example, an ADT vendor may support less than half the ADT message definitions the standard describes yet be functionally and operationally compliant. This is even more the case with Order Entry where vendor capabilities can vary considerably.

At one point, a "plug and play" proposal circulated in the HL7 community, but prescriptive approaches to interfacing have not been treated as feasible. Version 3.0 takes a significant step toward a more plug-and-play model. Since HL7 does not establish (or even endorse) canonical rules for compliance testing, an interactive interface development process is a natural consequence. None of the parties to an interface can work in isolation from the others.

HL7 is a broad protocol framework, so no one set of prescribed functions really exists. Application business rules determine which areas of the protocol make the most sense to support. Attempting to simply prescribe or anticipate a transaction set may run aground on the various functional concerns of interface requirements. From the point of view of a software integrator, HL7 messaging requires setting boundaries between application functions and messaging infrastructure. The most common technical approach is to use an interface engine as a messaging hub within which to address issues with vendor support for HL7 as well as to provide managed guaranteed delivery and queue management functions.

Protocol Description

HL7 is a structured, message-oriented protocol framework for computer communication between healthcare application systems. Rather than start from scratch, the original designers of HL7 turned to what is now ASC X.12 (EDI) for an example. (It does not take much examination to see some of the similarities.) The similarity is accidental for practical purposes since HL7 has evolved down its own path. Over the past several years, Health Level 7 (HL7) has grown in popularity from an initial vendor proposal for an OSI layer seven protocol offered in the public domain to an ANSI-recognized standard.

The protocol architecture is hierarchical, moving from high-level groupings and structures to a set of several hundred data fields. Each level of the hierarchy serves a different organizing purpose.

Functional Group

Areas of the protocol are grouped according to common application function; for example, ADT, Order Entry, Finance, Control, and Ancillary Reporting all represent groups described in the standard. Different functional groups are typically given individual chapters in the HL7 specification document.

Message Type

Within a functional group are defined one or more message types that can be implemented in various combinations to support high-level business rules for the applications involved. For example, ADT only specifies one message type while Order Entry describes more than a dozen.

Message Definition

Within each message type, one or more message definitions describe the specific set or combination of segments that make up a properly formed message. For example, ADT distinguishes among more than thirty separate message definitions based on "trigger events" or more detailed business rules. Each message definition includes one or more segments.

Segment Definition

Segments provide a logical grouping for data elements. For example, the Patient Identification segment (PID) includes fields for such identifying information as patient name, Social Security number, medical record number, account number, and miscellaneous demographic details. How fields are grouped in segments forms part of the HL7 implied data model. Segments can be required or optional, can be nested, and can repeat. A parsed message, then, can take on a relatively arbitrary yet unambiguous form. This is an important characteristic in the context of decoding and encoding messages.

Field

The standard identifies several hundred data elements for communicating patient demographic, clinical, and financial information. HL7 uses more than a dozen abstract data types to define the nature of the fields. (A consequence is that some fields may hold more than one data element.) For example, a field that holds a time stamp (TS) follows a prescribed format. In addition many fields are (or can be) coded, and the standard includes a variety of code tables to define acceptable contents. While each field is defined with a maximum length, the standard really doesn't intend to prescribe format to that level of detail. It merely includes lengths "because it helps readers understand the purpose of the field and it may have pragmatic importance in specific implementations." The "Logical Observation Identifiers Names and Codes" (LOINC) database offers a standardized way of naming clinical details that get exchanged.

On a functional level, HL7 can be thought of as an architecture with a set of constituent protocols. The division of message definitions into functional groups (and the division of HL7 itself into functional groups) has resulted in key conceptual differences between groups. The idea of the "trigger event" governs ADT messages. Each ADT message definition includes an event (EVN) segment that describes the necessary action. Order messages on the other hand follow a similar, but not identical scheme. Rather than an event segment, order messages use a common order segment (ORC) with an order control code (SEQ 1) that determines the function. The control code is analogous to a trigger event, but is further broken down into classes—request, acknowledgement, and notification.

In addition, rather than define ancillary-specific segments that fall under the general order message (ORM) definition, some ancillaries have distinct message definitions with the order entry group. Fortunately, these message definitions follow the same general encoding rules. Specific ancillary support for these fairly complex rules, however, is practically always a matter of negotiation. Not every vendor supports HL7 messaging in exactly the same way. No vendor supports all forms of all messages.

HL7 Implied Data Model

HL7 is not based on an explicit data model. As a consequence, compliance testing against a reference architecture has not been part of HL7 culture. The HL7 community itself tends to accept this as a reasonable trade-off for the sake of flexibility. Vendors and their systems are diverse and evolve slowly, so the benefits of greater purity of method would be at the cost of actual utility.

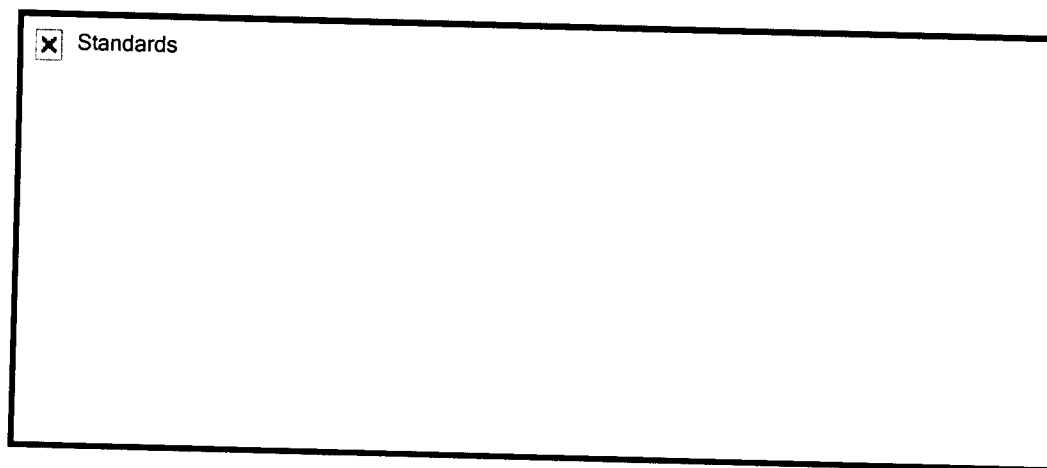
The HL7 protocol architecture, does imply a data model that is reflected in the message structure. HL7 has made a reasonable attempt to insure that data elements do not overlap in segment definitions, so the organization of data elements into message segments is analogous to a set of tables in a relational database. (Analogous because some abstract data types could violate the basic normalizing rule of one column/one element.) Message definitions would be analogous to database views that involve relating multiple tables to satisfy implied business rules.

Neither the implied data model nor the implied business rules should be considered a generic reflection of the shared attributes of all healthcare applications, even if they do represent much that is commonly accepted. As a consequence of this rather loose, architectural approach to integration, creating an HL7 interface between systems is a collaborative effort involving the application experts

(users) of the systems as well as the technical experts responsible for actually creating, testing, and supporting the interfaces.

Further Information

The HL7 community is large and well organized, and many resources are available. The natural starting place for additional authoritative information is HL7 itself. The progress of relevant healthcare application integration standards over the last several years has made for much more efficient integration of complex systems. The list below includes some links for browsing some of the technology background. Some efforts have been overtaken by events, but the principles are relevant and useful.



[< Back](#) *Altis—Process excellence through evidence-based management*

[Next >](#)

Copyright © 2010-2012 by Altis®, Inc. All Rights Reserved
Use of this web site signifies your agreement to the [Terms of Use](#).

20101222-975