

1484.1™

IEEE Standard for Learning Technology—Learning Technology Systems Architecture (LTSA)

IEEE Computer Society

Sponsored by the
Learning Technology Standards Committee



Published by
The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

12 December 2003

Print: SH95141
PDF: SS95141

IEEE Standard for Learning Technology—Learning Technology Systems Architecture (LTSA)

Sponsor

Learning Technology Standards Committee
of the
IEEE Computer Society

Approved 12 June 2003

IEEE-SA Standards Board

Abstract: A high-level architecture for information technology-supported learning, education, and training systems that describes the high-level system design and the components of these systems is specified in this standard.

Keywords: encoding, high-level architecture, implementation, interaction, learner, Learning Systems Technology Architecture (LTSA), system component, system notation

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2003 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 12 December 2003. Printed in the United States of America.

IEEE is a registered trademarks in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Incorporated.

Print: ISBN 0-7381-3714-6 SH95141
PDF: ISBN 0-7381-3715-4 SS95141

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied “**AS IS.**”

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

[This introduction is not part of IEEE Std 1484.1-2003, IEEE Standard for Learning Technology—Learning Technology Systems Architecture (LTSA).]

The Learning Systems Technology Architecture (LTSA) is an architecture based on abstract components. Higher-level abstractions can be “implemented” in lower levels: either as lower-level abstractions, or as actual implementations.

Learning technology systems (“implementations”) can be mapped to/from LTSA. The boundaries, functions, and decomposition of actual or other abstract learning technology systems might not have the same structure as LTSA, i.e., the mapping to LTSA might not be “one-to-one” (in the mathematical sense). Not all learning technology systems will have all components of LTSA, i.e., the mapping to LTSA might not be “onto” (in the mathematical sense).

The LTSA is intended to have wide applicability to learning technology systems. No particular generation technology is implied with LTSA, so LTSA might apply to the past (e.g., historic systems of 10, 30, and 100 years ago), the present (e.g., existing systems), and future (e.g., systems of 10 years from now).

The LTSA was developed (1996–2002) in the IEEE 1484.1 Architecture and Reference Model Working Group of the IEEE Learning Technology Standards Committee (LTSC). The web page for the 1484.1 Working Group and related resources is: “<http://ltsc.ieee.org/wg1>.”

Participants

At the time this standard was completed, the working group had the following membership:

John Tyler, *Chair*

Brant Cheikes, *Chair (1996–1999)*

Frank Farance, *Technical Editor*

Joshua Tonkel, *Technical Editor*

Carlos C. Amaro
Mitch Bonnett
Debbie Brown
Mike Collett
Erik Duval
Mike Fore
Eddy Forte
Jack Hyde

Igor Aizenberg
Anastasios Koutoumanos
Rolf Lindner
Bill McDonald
William Melton
Bill Olivier
Claude Ostyn
Bruce Peoples
Mike Petit

Dan Rehak
Tyde Richards
Kevin Riley
Jim Schoening
Kathy Sinita
Dan Suthers
Tom Wason
Eamonn Webster

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Carlos C. Amaro
Morris Balamut
Mitch Bonnett
Paul Boyd
Guru Dutt Dhingra
Frank Farance
Mike Fore
Eddy Forte
Wayne Hodgins
Robert Kelsey

Jim Logan
Gregory Luri
Jon Mason
William Melton
Raye Newmen
Bruce Peoples
Dan Rehak
Robby Robson
David Rockwell

Bruno Samuel
Randy Saunders
Jim Schoening
Yuriy Sheynin
Kathy Sinita
Thomas Starai
Joshua Tonkel
John Tyler
Eamonn Webster
Paul Work

When the IEEE-SA Standards Board approved this standard on 12 June 2003, it had the following membership:

Don Wright, *Chair*
Howard M. Frazier, *Vice Chair*
Judith Gorman, *Secretary*

H. Stephen Berger
Joe Bruder
Bob Davis
Richard DeBlasio
Julian Forster*
Toshio Fukuda
Arnold M. Greenspan
Raymond Hapeman

Donald M. Heirman
Laura Hitchcock
Richard H. Hulett
Anant Jain
Lowell G. Johnson
Joseph L. Koepfinger*
Tom McGean
Steve Mills

Daleep C. Mohla
William J. Moylan
Paul Nikolic
Gary Robinson
Malcolm V. Thaden
Geoffrey O. Thompson
Doug Topping
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Alan Cookson, *NIST Representative*
Satish K. Aggarwal, *NRC Representative*

Andrew Ickowicz
IEEE Standards Project Editor

Contents

1. Overview	1
1.1 Scope	1
1.2 Purpose	1
1.3 Normative wording vs. informative wording	2
1.4 Document organization (road map)	2
2. References	2
3. Definitions, abbreviations, and acronyms	4
3.1 Normative definitions	4
3.2 Informative definitions	6
3.3 Abbreviations and acronyms	7
4. Conformance	7
4.1 Identifying conforming LTSA system components	7
4.2 Implementation conformance statement	7
5. System components	8
5.1 Component organization	9
5.2 Learner entity	9
5.3 Learning parameters	10
5.4 Behavior	11
5.5 Evaluation	11
5.6 Learner information stored/retrieved by evaluation	12
5.7 Learner records	13
5.8 Learner information received by system coach	14
5.9 Learner information stored by system coach	14
5.10 Assessment information	15
5.11 Coach	16
5.12 Query	18
5.13 Learning resources	19
5.14 Catalog information	20
5.15 Locator sent by coach	20
5.16 Locator sent by delivery	21
5.17 Learning content	22
5.18 Delivery	22
5.19 Interaction context	23
5.20 Multimedia	24
5.21 Multiple roles for humans (informative)	24
5.22 Conceptual vs. actual implementations (informative)	25
Annex A (informative) Stakeholder perspectives and priorities	26
Annex B (informative) Operational components and interoperability	29
Annex C (informative) Architecture overview and general concepts	31

Annex D (informative) Illustrations of stakeholder mappings	36
Annex E (normative) Pro Forma Implementation Conformance Statement	85
Annex F (informative) Illustrations of conforming implementations	87
Annex G (informative) Conformance to IEEE Std 1471-2000	92
Annex H (informative) A description of system notation	93
Annex I (informative) Bibliography	97

IEEE Standard for Learning Technology—Learning Technology Systems Architecture (LTSA)

1. Overview

1.1 Scope

This standard specifies a high-level architecture for information technology-supported learning, education, and training systems that describes the high-level system design and the components of these systems. This standard covers a wide range of systems, commonly known as learning technology, education and training technology, computer-based training, computer assisted instruction, intelligent tutoring, etc. This standard is pedagogically neutral, content-neutral, culturally neutral, implementation-neutral, and platform-neutral. This standard provides a framework for understanding existing and future systems, promotes interoperability and portability by identifying abstract, high-level system interfaces, and incorporates a technical horizon (applicability) of at least 5–10 years while remaining adaptable to new technologies and learning technology systems. This standard will facilitate the development of configuration guidelines (e.g., profiles) for general learning technology systems. This standard is neither prescriptive nor exclusive.

1.2 Purpose

In general, the purpose of developing system architectures is to create high-level frameworks for understanding certain kinds of systems, their subsystems, and their interactions with related systems, i.e., more than one architecture is possible.

An architecture is not a blueprint for designing a single system, but a framework for designing a range of systems over time, and for the analysis and comparison of these systems, i.e., an architecture is used for analysis and communication.

By revealing the shared components of different systems at the right level of generality, an architecture promotes the design and implementation of components and subsystems that are reusable, cost-effective and adaptable, i.e., abstract, high-level interoperability interfaces and services are identified.

The architectural framework developed in this standard does not address the specific implementation details necessary to create learning technology system components.

1.3 Normative wording vs. informative wording

This document contains two types of technical description:

- *Normative wording*: This wording places technical requirements on conforming implementations—normative wording is the essence of this standard. Conformity assessment (e.g., conformance testing) is based solely on normative wording. Normative wording excludes introductory material, overview, rationale, footnotes, examples, bibliography, and informative clauses, subclauses, and annexes.
- *Informative wording*: This wording is helpful, but not required, for understanding this document. The notes given provide clarification of the text, examples, and guidance—they do not contain technical requirements and do not form an integral part of this standard.

1.4 Document organization (road map)

The following is an overview of each clause and annex appearing in this standard.

- Clause 1 (Introduction): background information and a high-level summary of the features of this standard.
- Clause 2 (Normative references): normative wording that is incorporated by reference to other standards and specifications.
- Clause 3 (Definitions): a list of terms and their definitions, and a list of abbreviations.
- Clause 4 (Conformance): the technical requirements for claiming conformance to this standard.
- Clause 5 (System components): the third refinement layer, the system components identified in learner-related design features, i.e., addressing the strengths and weaknesses of interacting with humans in learning technology systems.
- Annex A (Implementation perspectives and priorities): the next refinement layer, learning systems from a variety of perspectives.
- Annex B (Operational components and interoperability): the bottom refinement layer, generic components and building blocks of information technology systems as applied to learning technology.
- Annex C (Architecture overview and general concepts, informative): an overview of the architecture described in terms of refinement layers. Each layer is summarized.
- Annex D (Illustrations of stakeholder mappings, informative): examples of the fourth refinement layer, learning systems from a variety of perspectives.
- Annex E (Pro forma implementation conformance statement, normative): the form that must be completed so that implementations can claim conformance to this standard.
- Annex F (Illustrations of conforming implementations, informative): a small collection of implementations that illustrate how Clause 4 is applied.
- Annex G (Conformance to IEEE Std 1471-2000, informative): how LTSA conforms to IEEE Std 1471-2000 [B9] for architectural description.
- Annex H (A description of system notation, informative): a description of the notation used in this standard for system components.
- Annex I (Bibliography, informative): informative references to related documentation.

2. References

The following referenced documents are indispensable for the application of this standards. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 2382-1, Information technology—Vocabulary—Part 1: Fundamental Terms.¹

¹ISO/IEC publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse (<http://www.iso.ch/>). ISO/IEC publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA ([http://www.ansi.org/](http://www ansi.org/)).

- ISO/IEC 2382-2, Data Processing—Vocabulary—Section 2: Logic Operation.
- ISO/IEC 2382-3, Information Processing Systems—Vocabulary—Part 3.
- ISO/IEC 2382-4, Information technology—Vocabulary—Part 4: Organization of Data.
- ISO/IEC 2382-5, Information Processing Systems—Vocabulary—Part 5: Representation of Data.
- ISO/IEC 2382-6, Information Processing Systems—Vocabulary—Part 6.
- ISO/IEC 2382-7, Information technology—Vocabulary—Part 7: Computer Programming.
- ISO/IEC 2382-8, Information technology—Vocabulary—Part 8: Security.
- ISO/IEC 2382-9, Information technology—Vocabulary—Part 9: Data Communication.
- ISO/IEC 2382-10, Data Processing—Vocabulary—Section 10: Operating Technology.
- ISO/IEC 2382-11, Information Processing Systems—Vocabulary—Part 11.
- ISO/IEC 2382-12, Information Processing Systems—Vocabulary—Part 12: Peripheral Equipment.
- ISO/IEC 2382-13, Information technology—Vocabulary—Part 13: Computer Graphics.
- ISO/IEC 2382-14, Information technology—Vocabulary—Part 14: Reliability, Maintainability, and Availability.
- ISO/IEC 2382-15, Data Processing—Vocabulary—Part 15: Programming Languages.
- ISO/IEC 2382-16, Information technology—Vocabulary—Part 16: Information Theory.
- ISO/IEC 2382-17, Information technology—Vocabulary—Part 17: Databases.
- ISO/IEC 2382-18, Information Processing Systems—Vocabulary—Part 18: Distributed Data Processing.
- ISO/IEC 2382-19, Information Processing Systems—Vocabulary—Part 19: Analog Computing.
- ISO/IEC 2382-20, Information technology—Vocabulary—Part 20: System Development.
- ISO/IEC 2382-21, Data Processing—Vocabulary—Part 21: Interfaces.
- ISO/IEC 2382-22, Information Processing Systems—Vocabulary—Part 22.
- ISO/IEC 2382-23, Information technology—Vocabulary—Part 23: Text Processing.
- ISO/IEC 2382-24, Information technology—Vocabulary—Part 24: Computer-Integrated Manufacturing.
- ISO/IEC 2382-25, Information technology—Vocabulary—Part 25: Local Area Networks.
- ISO/IEC 2382-26, Information technology—Vocabulary—Part 26: Open Systems Interconnection.
- ISO/IEC 2382-27, Information technology—Vocabulary—Part 27: Office Automation.

ISO/IEC 2382-28, Information technology—Vocabulary—Part 28: Artificial Intelligence—Basic Concepts and Expert Systems.

ISO/IEC 2382-29, Information technology—Vocabulary—Part 29: Artificial Intelligence—Speech Recognition and Synthesis.

ISO/IEC 2382-31, Information technology—Vocabulary—Part 31: Artificial Intelligence—Machine Learning.

ISO/IEC 2382-32, Information technology—Vocabulary—Part 32: Electronic Mail.

ISO/IEC 2382-34, Information technology—Vocabulary—Part 34: Artificial Intelligence—Neural Networks.

3. Definitions, abbreviations, and acronyms

For the purposes of this standard, the following terms and definitions apply. IEEE 100, *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition [B8], should be referenced for terms not defined in this clause.

3.1 Normative definitions

3.1.1 abstraction: (A) A concept or generalization that specifies only those features of its instances that are relevant to a particular purpose. (B) The process of creating an abstraction in sense (A). *See also:* **abstraction-implementation boundary; implementation.**

3.1.2 abstraction-implementation boundary: (A) The mapping of an abstraction to an implementation and vice versa. (B) The boundary between different levels of granularity. *See also:* **abstraction; implementation; refinement layer.**

3.1.3 abstraction-implementation layer: *See:* **refinement layer.**

3.1.4 actual implementation: *See:* **implementation (A).**

3.1.5 binding: An application or mapping from one framework or specification to another.

3.1.6 coding: (A) In information interchange, a formalized or structured representation of information. *See also:* **encoding.** (B) A process of representing information in some structure.

3.1.7 collaboration: In the context of LTSA, communication among the individual learners of the learner entity.

3.1.8 conceptual implementation: *See:* **implementation (B).**

3.1.9 control flow: An information flow that starts, stops, or changes processing. *See also:* **data flow.**

3.1.10 control protocol: The actions, responses, information, and processing states for starting and stopping the flow of information. *See also:* **data protocol.**

3.1.11 control store: A repository for control information. *Example:* an event log is a control store. *See also:* **data store; store.**

3.1.12 data flow: An information flow that represents the main inputs and/or outputs of a system or subsystem. *See also:* **control flow; flow.**

3.1.13 data process: The transformation of data inputs into data outputs, or the transformation of a mix of data and control inputs into data and control outputs.

3.1.14 data protocol: The actions, responses, information, and processing states for the flow of information among subsystems. *See also:* **control protocol.**

3.1.15 data transformation process: *See:* **data process.**

3.1.16 data store: An information repository for data information. *Example:* a database is a data store.

3.1.17 design priority: The rank of technical design issues from largest effect to smallest effect. *See also:* **primary design issue, secondary design issue.**

3.1.18 encoding: The bit and byte format and representation of information. *See also:* **coding.**

3.1.19 flow: The transfer of information from one system or subsystem to another. *See also:* **control flow, data flow.**

3.1.20 implementation: (A) A working instance of an abstraction. *Syn:* **actual implementation.** (B) A low-level abstraction. *Syn:* **conceptual implementation.** (C) The process of creating an implementation in sense (A) or (B). *See also:* **abstraction; refinement layer.**

3.1.21 implementation under test: In conformance testing, the system that is being tested.

3.1.22 interaction (learning technology): An information exchange between a learner entity and a system.

3.1.23 learner: An individual engaged in acquiring knowledge or skills with a learning technology system.

3.1.24 learner entity: An individual learner, or a group of learners considered as a single entity that interacts with a learning technology system. *See also:* **learner.**

3.1.25 one-way flow: The transfer of information in a single direction from one system or subsystem to another. In the case of connecting multiple systems or subsystems, a one-way flow has a single source (origin), or a single sink (destination), or both. *See also:* **flow; two-way flow.**

3.1.26 primary design issue: The main technical focus of implementation (system) development, which has the largest effect on the nature of the implementation (system). There may be more than one primary design issue. *See also:* **design priority; secondary design issue.**

NOTE—The judgement of “largest” and “effect” are determined by the user of the standard.

3.1.27 process: An active system component that transforms its inputs into outputs. *See also:* **data process.**

3.1.28 secondary design issue: After the primary design issue(s), the next largest effect of technical development. *See also:* **design priority; primary design issue.**

NOTE—The judgement of “largest” and “effect” are determined by the user of the standard.

3.1.29 store: An inactive system component used as an information repository. *See also:* **control store, data store.**

3.1.30 system component: In system notation, a process, a store, or a flow.

3.1.31 system notation: A descriptive technique that decomposes systems into subsystems of processes and stores connected by flows. *See also:* **text description**.

NOTE—System notation is useful when there are a relatively small number of subsystems, the functions or roles of each subsystem are well-defined, and the connections between subsystems are established and unchanging.

3.1.32 text description: A method of identifying system and subsystem functionality that uses words instead of graphics. *See also:* **system notation**.

NOTE—Text descriptions are useful if the boundaries of a subsystem are not well defined, or bus notation and system notation are inadequate.

3.1.33 two-way flow: Transfer of information in both directions between systems or subsystems. *See also:* **information flow; one-way flow**.

3.2 Informative definitions

This subclause is informative.

3.2.1 administrator: A person responsible for purchasing systems, managing systems, or managing institutions.

3.2.2 developer: A creator of learning content and/or software.

3.2.3 distance learning system: A learning technology system that (1) has at least one LTSA system component flow (learning parameters, behavior, learner information, assessment information, query, catalog information, locator, learning content, multimedia, interaction context) as a primary design issue, and (2) the primary design issues for said flows include at least one of the following issues: net-work delays, reliability, bandwidth, responsiveness, availability.

3.2.4 distributed system: The dispersion of processing and operations among components.

3.2.5 learning experience: The events surrounding the learner while he/she in learning. Because the nature of learning is complex, it may be difficult or impossible to identify all the events that comprise a learning experience.

3.2.6 nomadic learning: A form of learning in which a learner has continuity of service across different sessions, and possibly, different locations. *Example:* a learner may have a different teacher every year of school; a learner may change institutions from time to time.

3.2.7 nomadicity: (A) Continuity of service across different sessions and, possibly, different locations. (B) A property of network communications that are sometimes connected.

3.2.8 stakeholder: (A) A materially interested person, organization, or entity. (B) A group of persons, organizations, or entities that have a common interest.

Example 1: Each LTSA stakeholder may identify its stakeholder perspective by using an existing LTSA stakeholder diagram or creating a new diagram.

Example 2: A “content developer” stakeholder represents all those who have material interest in content development.

3.3 Abbreviations and acronyms

API	Application Programming Interface
ICS	implementation conformance statement
LTSC	Learning Technology Standards Committee
QoS	quality of service

4. Conformance

A conforming implementation shall complete the pro forma implementation conformance statement (ICS). A conforming implementation shall contain at least one LTSA system component (layer 3) other than the learner entity. For each process or store to which an implementation claims conformance, the implementation shall also conform to the requirements of the input and output flows of that process or store. Annex E contains a printed copy of the ICS form. Annex F contains illustrations of systems that conform to LTSA and includes sample ICS forms.

4.1 Identifying conforming LTSA system components

The ICS shall identify which LTSA system component (layer 3) features conform in the implementation (letter codes in parentheses are abbreviations for LTSA system components):

- LTSA learner entity (LENT)
- LTSA behavior data flow from learner entity to assessment (BEHV)
- LTSA evaluation process (EVAL)
- LTSA assessment data flow from evaluation to coach (ASMN)
- LTSA learner information data flow between evaluation and learner records (LIEV)
- LTSA learner records data store (LREC)
- LTSA learner information data flow from learner records to coach (LIRC)
- LTSA learner information data flow from coach to learner records (LICR)
- LTSA learning parameters between learner entity and coach (LPAR)
- LTSA coach process (COCH)
- LTSA query control flow from coach to learning resources (QUCR)
- LTSA learning resources data store (RESO)
- LTSA catalog information data flow from learning resources to coach (QURC)
- LTSA locator data flow from coach to delivery (LOCD)
- LTSA delivery process (DELV)
- LTSA locator control flow from delivery to learning resources (LODR)
- LTSA learning content data flow from learning resources to delivery (LCNT)
- LTSA interaction context data flow from delivery to evaluation (CTXT)
- LTSA multimedia data flow from delivery to learner entity (MULT)

The ICS shall reference its systems and subsystems, and shall describe the mapping of these systems and subsystems to/from the available LTSA features.

NOTE—Implementations are encouraged to use the LTSA layer 4 notation as described in Annex A.

4.2 Implementation conformance statement

An implementation shall be used in one the following formats:

- “[implementation conformance statement] all components”

- “[implementation conformance statement] components X1-X2-X3-...”
- “[implementation conformance statement] components except X1-X2-X3-...”

where

- The first format indicates all LTSA system components in an implementation conformance statement.
- The second format indicates only certain LTSA system components in an implementation conformance statement.
- The third format indicates all components except certain LTSA system components in an implementation conformance statement.
- The list X1-X2-X3-... is a list of hyphen-separated LTSA system component abbreviations, as described above.

Example 1: The conformance label “[implementation conformance statement] all components” means that the implementation claims to conform to all LTSA system components.

Example 2: The conformance label for a web browser used in a learning environment might be “[implementation conformance statement] components EVAL-ASMN-COCH-LOCD-DELV-CTXT,” see also Annex F.

Example 3: The conformance label for a learning media “player,” e.g., just insert a CD-ROM, might be “[implementation conformance statement] except components RESO”—the learning resources are the only component missing.

5. System components

This clause describes the processes, stores, and flows of the Learning Technology Systems Architecture. Processes are described in terms of boundaries, inputs, process (functionality), and outputs. Stores are described by the type of information stored, and by search, retrieval, and updating methods. Flows are described in terms of connectivity (one-way, two-way, static connections, dynamic connections, etc.) and the type of information across the flow. The LTSA system components are described in Yourdon notation—Annex G.

The LTSA system components identify the abstract, high-level interoperability interfaces for learning technology systems. The LTSA does not identify all interoperability interfaces for particular learning technology systems (e.g., the interoperability interfaces for a particular application or operating platform). The LTSA does not identify interoperability interfaces for related systems, such as content development or administrative systems. Subclause 5.22 and Annex F illustrate how the LTSA may be used to identify critical interoperability interfaces within actual implementations.

As explained elsewhere in this standard, the descriptions are to be understood as specifying general components, and the purpose of the notation is to identify generic features. Actual implementations of learning technology systems may not fit these component boundaries exactly, but might represent implementation variations. For example, many commercial learning management systems combine portions of the evaluation, delivery, and coach processes into a single session presentation tool. This combination might be motivated by implementation and commercial efficiency, but conceptually, the components are separate, and some implementations keep the components separate. In this respect these combination systems resemble automotive vehicles, in which the steering and power management are located together in front of the driver, but are conceptually separate; and some implementations separate the components (e.g., long fire brigade trucks separate steering and power management).

NOTE—This clause is mostly definitional in nature. Thus, this clause contains few assertions, e.g., sentences containing the verbs “shall,” “should,” or “may.”

5.1 Component organization

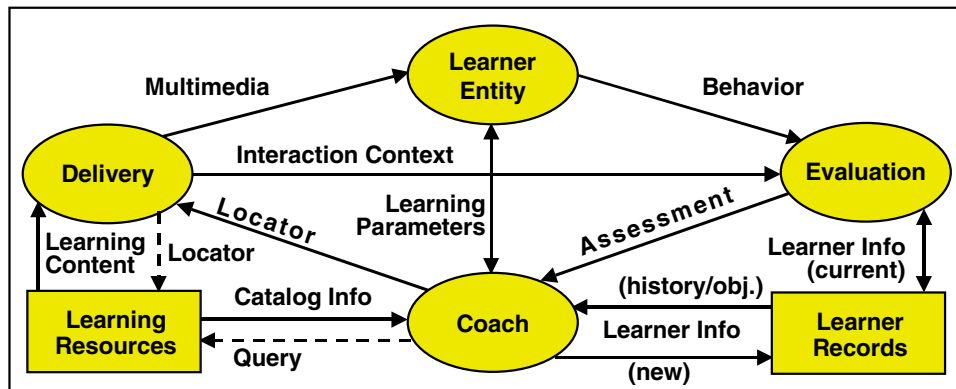


Figure 1 – The LTSA system components

The LTSA system components are:

- *Processes*: learner entity, evaluation, coach, delivery.
- *Stores*: learner records, learning resources.
- *Flows*: learning parameters, behavior, assessment information, learner information (three times), query, catalog info, locator (twice), learning content, multimedia, interaction context.

Throughout this standard the names of the LTSA components may be prefixed with “LTSA” in order to distinguish them from the corresponding generic nouns when the context is not obvious, e.g., “LTSA multimedia” vs. “multimedia.” In some cases, the word “process,” “store,” or “flow” will be used to clarify the usage of the term (e.g., “evaluation process” vs. “evaluation”).

NOTE—Throughout this clause, the LTSA “primary” notation (e.g., bold or red) is used to highlight the system component being defined; the LTSA “secondary” notation (e.g., double line or blue) is used to highlight components that are connected to the component being defined.

5.2 Learner entity

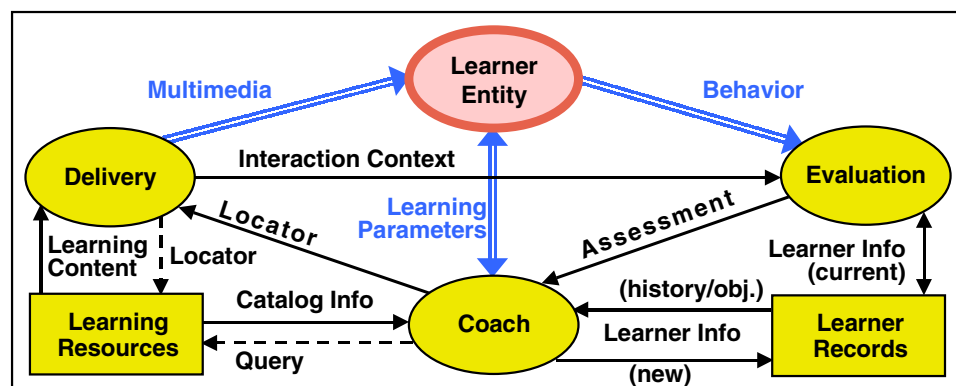


Figure 2—Learner entity (a process): the abstraction of the human learner

5.2.1 Definition

An abstracted process that represents an abstraction of a human learner. The learner entity may represent a single learner, a group of learners learning individually, a group of learners learning collaboratively, a group of learners learning in different roles, and so on.

5.2.2 Inputs/outputs

- *Input*: The learner entity process may receive a multimedia presentation via the multimedia data flow.
- *Output*: The learner's behavior may be observed and emitted via the behavior data flow.
- *2-Way*: The learning parameters are negotiated with the coach process via the learning parameters data flow.

NOTES:

1—At this level of abstraction, the multimedia presentation and observable behavior are diagrammed separately.

2—Actual implementations may combine these features into one or more human interface modules (e.g., windowing systems), session presentation modules (e.g., web browsers), tutoring tools (e.g., specialized applications), experimentation and discovery laboratories, and so forth.

5.3 Learning parameters

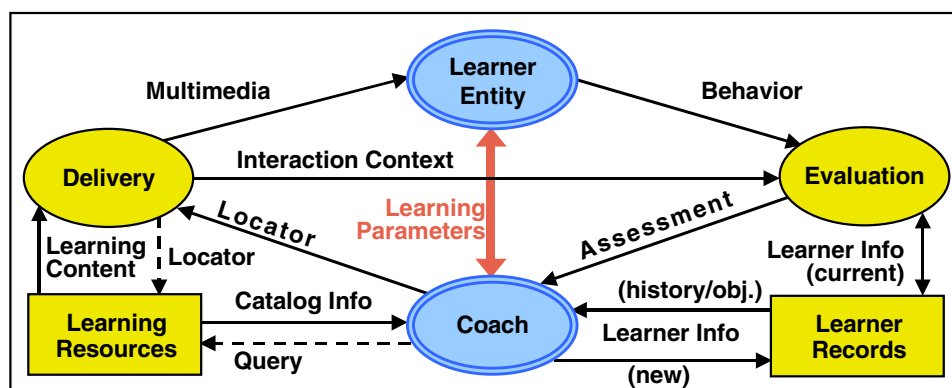


**Figure 3—Learning parameters (a data flow):
all interested parties contribute to learning parameters**

5.3.1 Definition

A 2-way data flow representing exchange (e.g., negotiation) between the learner entity process and the coach process.

NOTE—In addition to the human learner(s), the parent, teacher, mentor, employer, and/or institution may participate in the negotiation of learning parameters.

5.3.2 Information type

Learning parameters may include information types, such as cultural adaptation parameters, accessibility requirements and parameters for people with physical limitations (e.g., blindness, deafness) and cognitive limitations.

5.4 Behavior

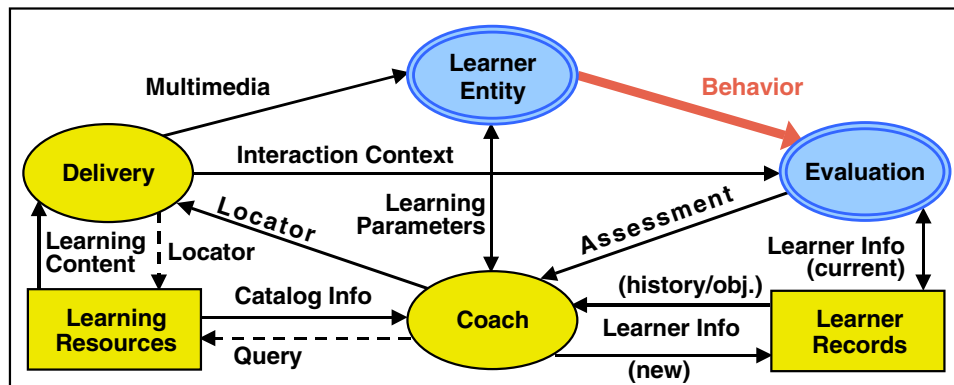


Figure 4—Behavior (a data flow): the coding and encoding of learner entity's actions

5.4.1 Definition

A data flow from the learner entity process to the evaluation process that represents information about the learner's activities and actions, which may be used by evaluation process.

5.4.2 Information type

Behavior information, via the behavior data flow, may include keyboard clicks, mouse clicks, voice response, choices, written responses, etc.

Example: A “control wheel” (a rounded wheel) might be used for airplane flight simulation and automobile driving simulation. This behavior information might be represented by the number of degrees that the wheel moved. Note that the same behavior information, a wheel moving X degrees clockwise, may have substantially different meaning in different contexts, e.g., flight simulation vs. driving simulation.

5.5 Evaluation

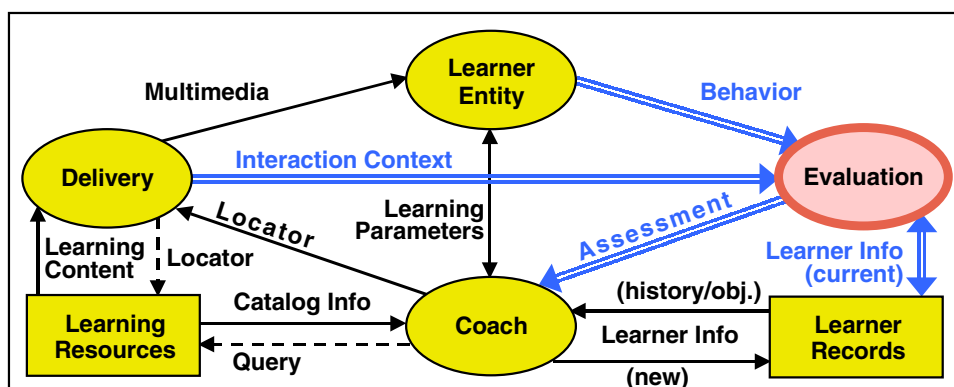


Figure 5—Evaluation (a process): the processing of behavior information to produce assessment and learner information

5.5.1 Definition

An abstracted process that may produce measurement(s) of the learner entity.

5.5.2 Inputs/outputs

- *Input*: The learner entity's observable behavior via the behavior data flow.
- *Input*: The interaction context data flow may provide context to the learner entity's behavior to determine the appropriate evaluation.
- *Output*: Assessment information may be sent to the coach via the assessment data flow.
- *Input/output*: Learner information may be retrieved and stored (via the learner information data flow) during evaluation processing in the learner records.

Example: A learner entity is expected to select from a multiple choice question and the correct answer is “#2.” The evaluation process needs to know the context of the learning interaction to determine which key-strokes “2,” “#2,” and/or “two” are the correct answer. The interaction context may be used to correlate the appropriate behavior(s) with the learning interaction(s).

5.6 Learner information stored/retrieved by evaluation

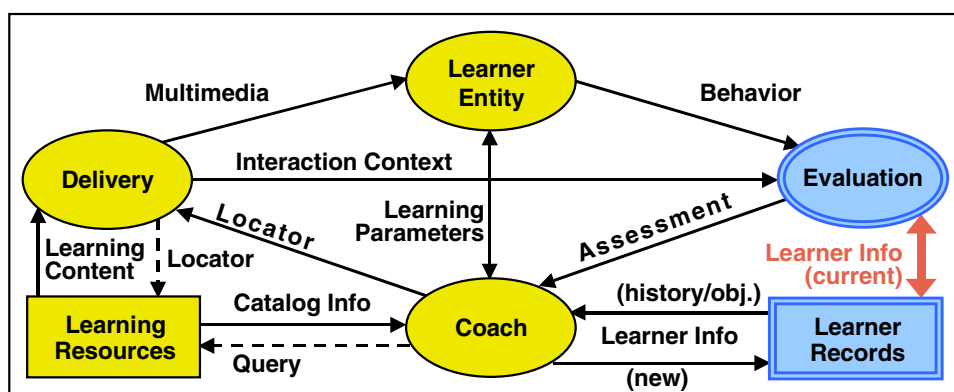


Figure 6—Learner information (a data flow): output from evaluation, to be stored as learner entity history information in learner records

5.6.1 Definition

A 2-way data flow between the evaluation process and the learner records data store that represents the storage and retrieval of learner information.

5.6.2 Information type

Data that represents past, present, or future learner information, such as activities, grades, logs, objectives, etc.

NOTE—The granularity is unspecified for learner information, e.g., the evaluation process may store/retrieve learner information as much as every mouse click or as little as every semester.

Example: “question 17, answered correctly, 85 seconds elapsed.”

5.7 Learner records

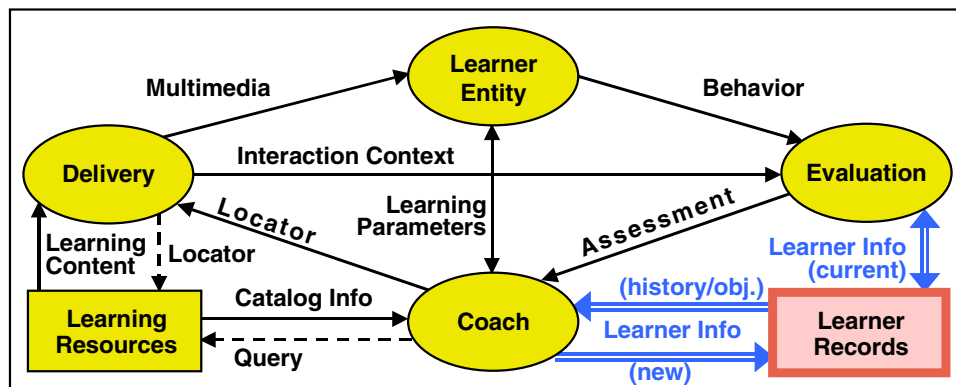


Figure 7—Learner records (a data store): storage and retrieval of learner information of the past (e.g., history), present (e.g., “suspends,” current assessment), and future (e.g., objectives)

5.7.1 Definition

A data store of learner information, such as performance, preference, and other types of information. The learner records may store information (which may be later retrieved) about the past (e.g., historical learner records), but may also hold information about the present (e.g., current assessments for suspending and resuming sessions) and the future (e.g., pedagogy, learner, or employer objectives).

5.7.2 Store/retrieve

- *Store*: The evaluation process may store learner information (e.g., grades on lessons) via the learner information data flow.
- *Retrieve*: The evaluation process may retrieve learner information (e.g., grades on lessons) via the learner information data flow.
- *Store*: The coach process may store learner information (e.g., performance, preference, and other types of learner information) via the learner information data flow.
- *Retrieve*: The coach process may retrieve learner information (e.g., performance, preference, and other types of learner information) via the learner information data flow.

Example 1: The coach may store performance information, such as assessment information and certifications, in the learner records.

Example 2: The coach may store “suspends” (bookmarks) as performance information for saving the learner entity’s session and resumption at some future time.

5.8 Learner information received by system coach

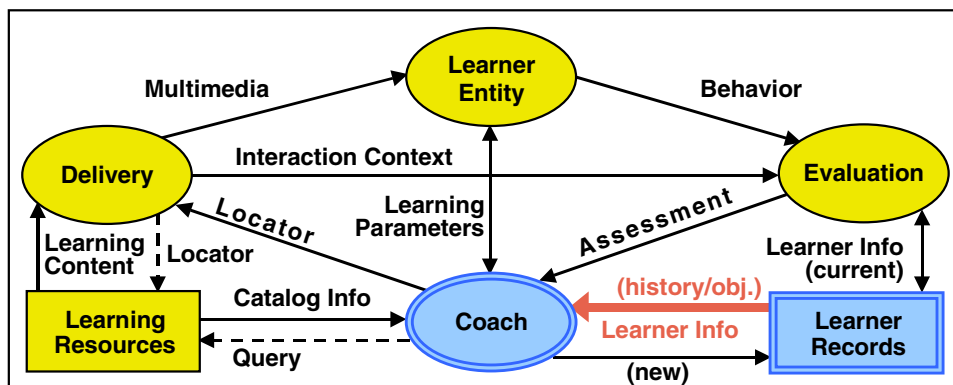


Figure 8—Learner information (a data flow): coach retrieves learner records, returned as learner information

5.8.1 Definition

A 1-way data flow from the learner records data store to the coach process that represents the coach process' requests for information from the learner records.

5.8.2 Information type

Performance, preference, and other learner information.

NOTE—Typically, historical information and preference information is retrieved, but current information (e.g., “suspends” for resuming sessions) and future information (e.g., template of future academic objectives) may be retrieved, too.

5.9 Learner information stored by system coach

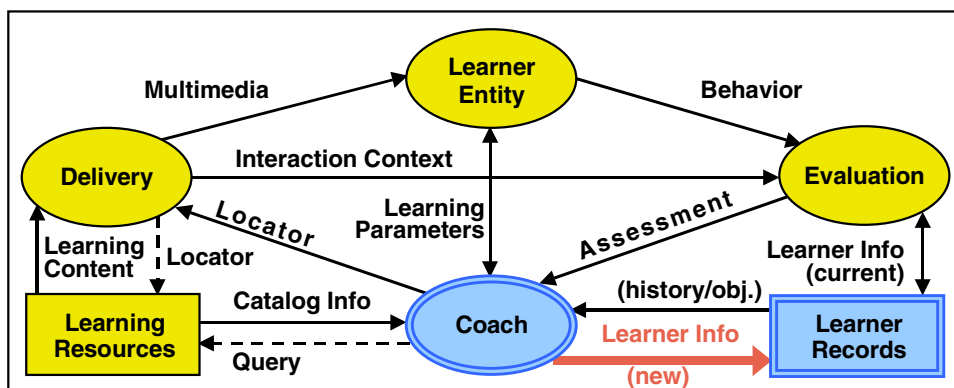


Figure 9—Learner information (a data flow): coach stores assessments and certifications in learner records

5.9.1 Definition

A 1-way data flow from the coach process to the learner records data store that represents the coach process' requests to store learner information.

5.9.2 Information type

Performance, preference, and other learner information.

Example 1: The coach may store performance information, such as assessment information and certifications, in the learner records.

Example 2: The coach may store “suspends” (bookmarks) as performance information for saving the learner entity’s session and resumption at some future time.

Example 3: The coach may store preferences in the learner records data store.

5.10 Assessment information

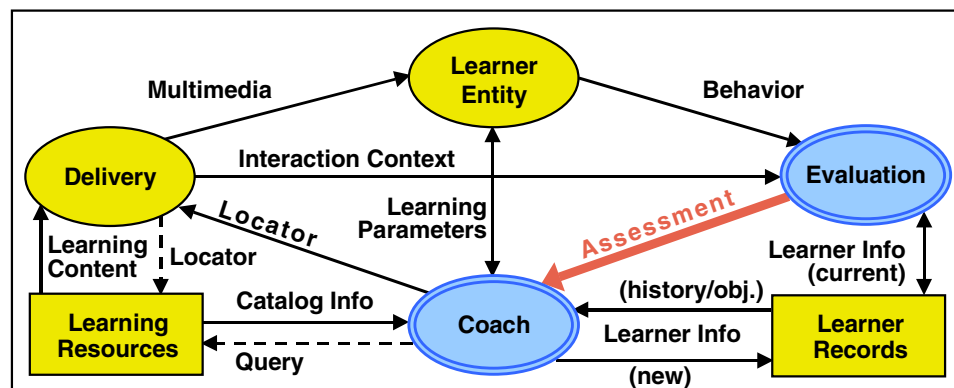


Figure 10— Assessment information (a data store): output from evaluation, represents learner entity’s “current state”

5.10.1 Definition

A data flow from the evaluation process to the coach process that represents information about the learner’s current state, which may be used by the coach process to determine optimal learning experiences.

5.10.2 Information type

Assessment information data flow may include performance information and other learner information.

NOTE—The distinction between the assessment information data flow and the flows from learner records is: the assessment information data flow is primarily intended to convey information about “where the learner is at” (an implementation-specific notion) and initiated by the evaluation process, while the flows to/from the learner records data store represent information flows that the coach has initiated (to/from the learner records data store). Implementations may use none, any, or all of these flows.

5.11 Coach

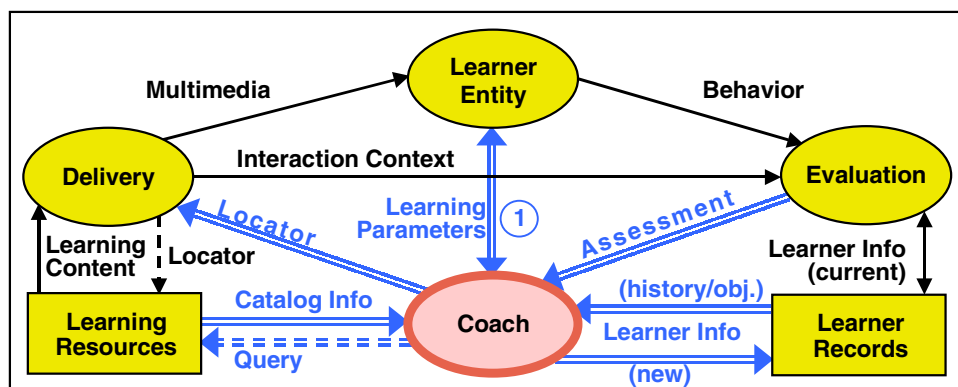


Figure 11—Coach (a process), Step 1: negotiates/exchanges learning parameters for optimum learning experience

5.11.1 Inputs/outputs

- *2-Way*: The learning parameters (learning parameters data flow) may be negotiated/exchanged with the learner entity.

NOTE—Learning styles, strategies, etc., may be chosen by either the learner entity (one-way negotiation, i.e., an assertion or inquiry), the coach (one-way negotiation, i.e., an assertion or inquiry), both the learner entity and the coach process (two-way negotiation), or an external authority (e.g., parent, teacher, institution, or content developer).

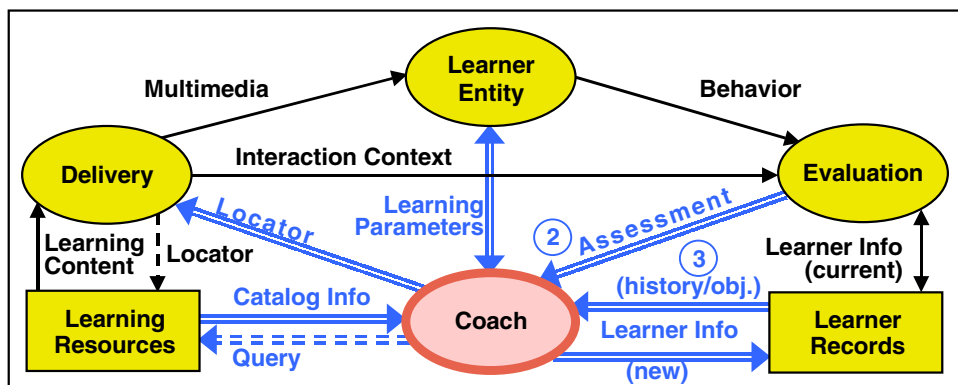


Figure 12—Coach (a process), Steps 2 and 3: receives current assessment information from evaluation. Searches and retrieves learner information relevant to the current learning experience

5.11.2 Inputs/outputs

- *Inputs*: The current assessment information (assessment information data flow) from the evaluation process.
- *Inputs*: Learner information (learner information data flow) from the learner records data store.

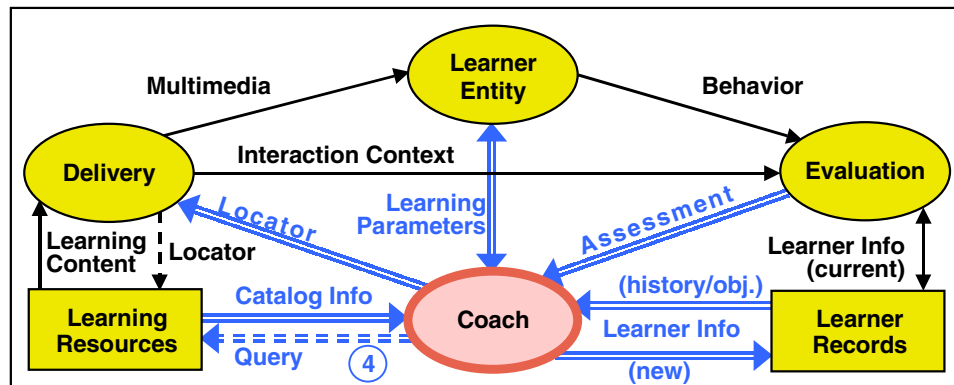


Figure 13—Coach (a process), Step 4: searches learning resources via queries for appropriate learning content. Learning resources returns “found” catalog info (a.k.a., learning object metadata) that matches the query

5.11.3 Inputs/outputs

- *Output:* Queries, a control flow, may be sent to the learning resources data store to search for (appropriate) learning materials.
- *Input:* The learning resources data store may return catalog information (catalog information data flow), e.g., a list of locators that match the search query.

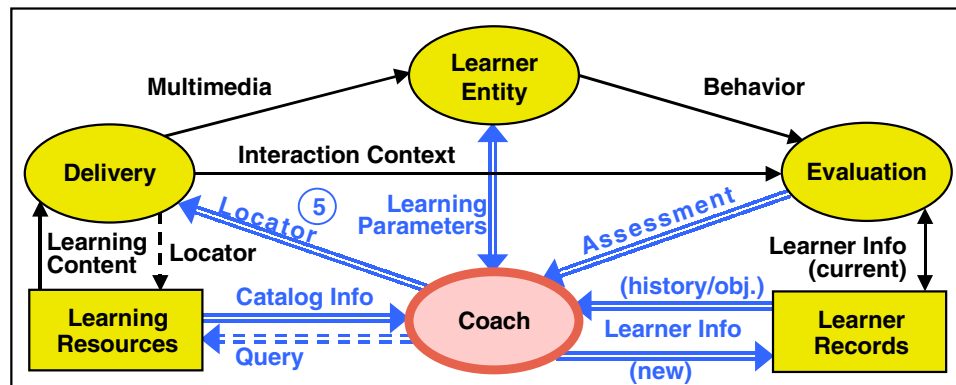


Figure 14—Coach (a process), Step 5: extracts the locators (e.g., URLs) from the “found” catalog info (learning object metadata). Sends the locators to the delivery process to direct the learning experience

5.11.4 Inputs/outputs

- *Output:* Locators (e.g., a lesson plan, pointers to content) may be sent, via the locator data flow, to the delivery process.

5.12 Query

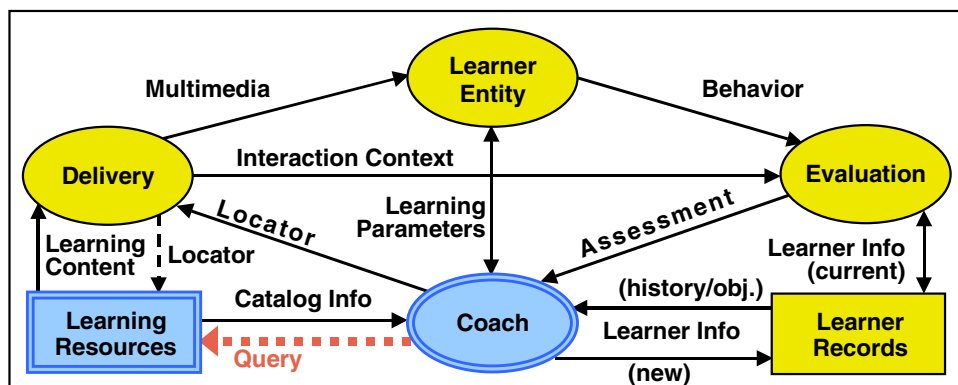


Figure 15—Queries (a control flow): the “search request” when looking for appropriate learning content in the learning resources

5.12.1 Definition

A 1-way control flow from the coach process to the learning resources that represents search requests for learning content.

NOTE—This flow (search requests) is a control flow and not a data flow because the (LTSA queries) flow does not represent inputs or data that are stored in the data store.

5.12.2 Information type

A query (via control flow) is a set of search criteria.

NOTE—The queries may specify search criteria based on, in part, learning parameters, assessment information, and learner information. Typically, the queries are searching for learning content that is appropriate for the learner entity.

5.13 Learning resources

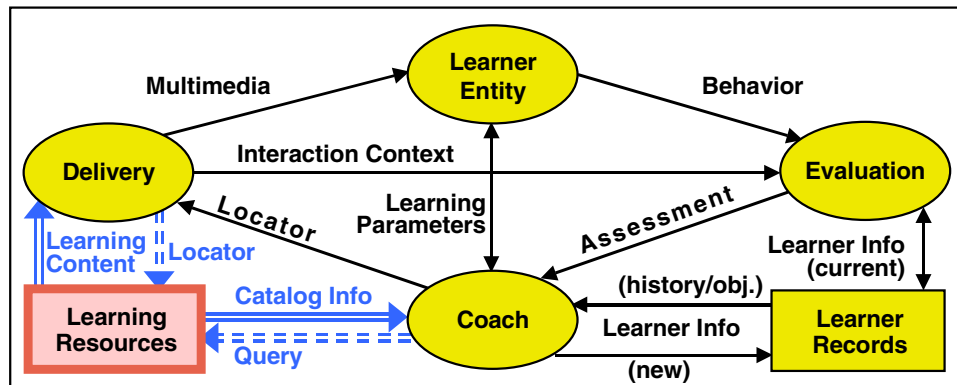


Figure 16—Learning resources (a data store): a database that represents “knowledge,” information, and other resources uses in the learning experiences. The learning resources may be represented as presentations, tutorials, experiments, lessons, etc.

5.13.1 Definition

A data store that may include representations of knowledge, presentations, tutorials, tutors, tools, experiments, laboratories, and other learning materials.

5.13.2 Store/retrieve

- *Retrieve:* May be searched by queries (via query control flow).
- *Retrieve:* The matching information is returned via the catalog info data flow, i.e., a set of content tags that are, conceptually, “card catalog” entries (also known as “learning object metadata”). The locators (conceptually, “call numbers” on the bindings of the “books in the digital library,” e.g., URLs) may be extracted from the catalog information.
- *Retrieve:* The locators may be used by the delivery process to retrieve learning content.
- *Retrieve:* The retrieved learning content is returned via the learning content data flow.

Example: A query on a topic in chemistry (specified as via a query control flow) might return a set of catalog information (via catalog info data flow) that includes a laboratory experiment simulating the behavior of solids, liquids, and gases, a presentation on Boyle’s Law, a bibliography of related materials, a tutorial, a chemistry tutor (the tutor may be human or surrogate; geographical nearness is irrelevant), and an ontology (a conceptual model of the subject represented as generic learning content) for temperature.

5.14 Catalog information

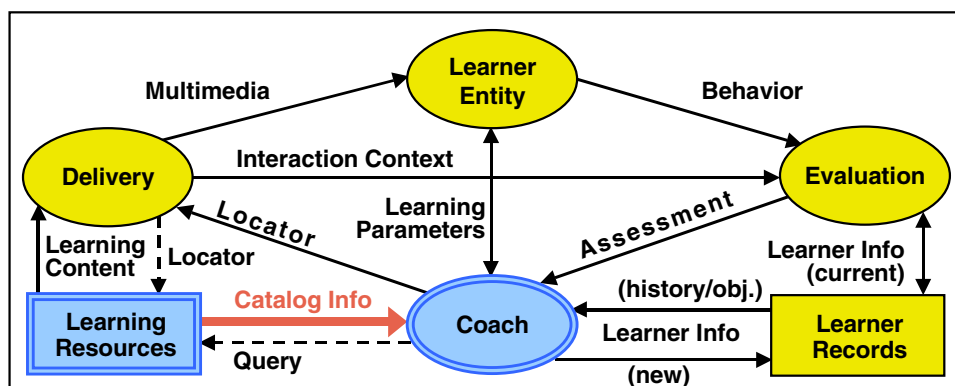


Figure 17—Catalog information (a data flow): represents “card catalog” information about learning content in the learning resources. Catalog info is also known as “learning object metadata”

5.14.1 Definition

A 1-way data flow from learning resources to the coach process that represents the result of searches of the learning resources data store, as directed by the query control flow (queries).

5.14.2 Information type

Information that describes learning resources.

NOTES:

1—The catalog information is also known as “learning object metadata.” Catalog information is similar to “card catalog” entries in a library.

2—In response to the query, it is possible to return the full learning content embedded within the catalog information (via catalog info data flow).

5.15 Locator sent by coach

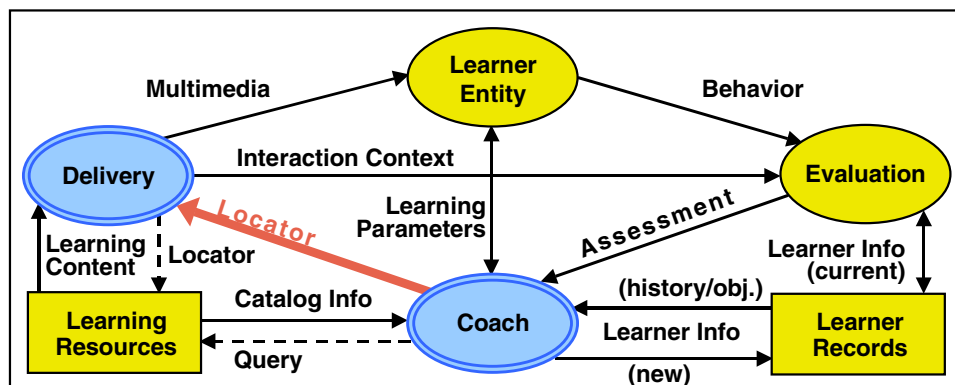


Figure 18—Locator (a data flow): represents the “call number” of the learning content in the learning resources. Web-based systems typically use URLs for locators

5.15.1 Definition

A 1-way data flow from the coach process to the delivery process that identifies or points to learning content.

5.15.2 Information type

Identifiers or pointers.

NOTES:

1—Using the library analogy, locators are similar to “call numbers” in a card catalog system.

2—This locator data flow (from coach process to delivery process) is the same information type as the locator control flow from the delivery process to the learning resources.

Examples: A URL, a URN, a URI, a pathname.

5.16 Locator sent by delivery

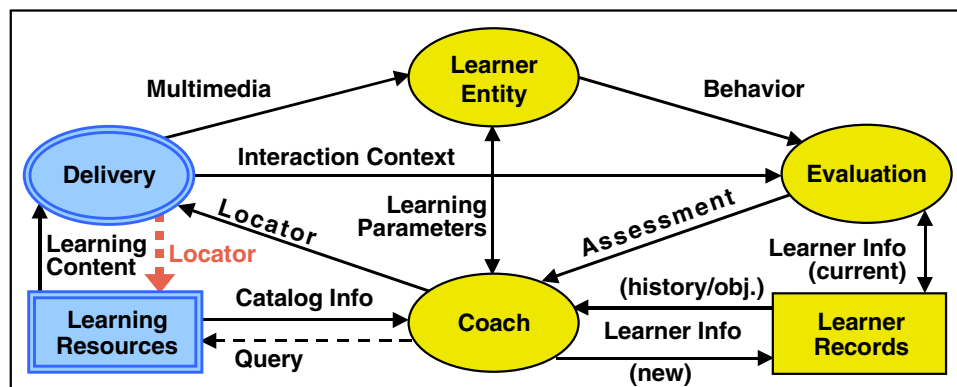


Figure 19—Locator (a control flow) sent by delivery

5.16.1 Definition

A 1-way control flow from the delivery process to the learning resource store, is a control flow containing locators identifying or pointing to learning content.

NOTE—This flow (locators that identify learning content to be retrieve) is a control flow and not a data flow because the flow (this locator flow) does not represent inputs or data that are stored in the data store. In 5.15, the locator is a data flow in the context of coach to delivery. The notion of data flow vs. control flow is contextual (see H.2.5).

5.16.2 Information type

Identifiers or pointers.

Example: A web URL.

5.17 Learning content

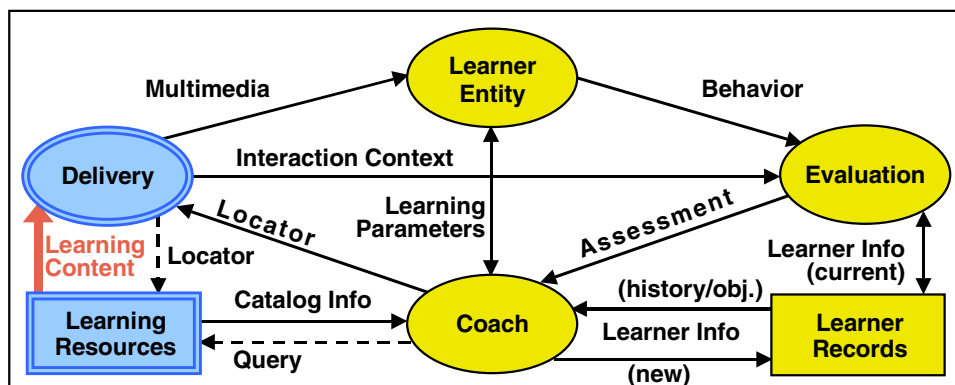


Figure 20—Learning content (a data flow): the coding of materials from the learning resources. Learning content may be lessons, presentations, tutorials, tutors, experiments, etc.

5.17.1 Definition

A 1-way data flow that represents materials that help create, coach, suggest, deliver, etc., the learning experience(s).

NOTE—The learning content data flow may be identified or referenced by the locator control flow, retrieved by the learning resources data store, and transformed by the delivery process into an interactive multimedia learning experience (e.g., delivered via the multimedia data flow, interaction context data flow, and behavior data flow).

5.18 Delivery

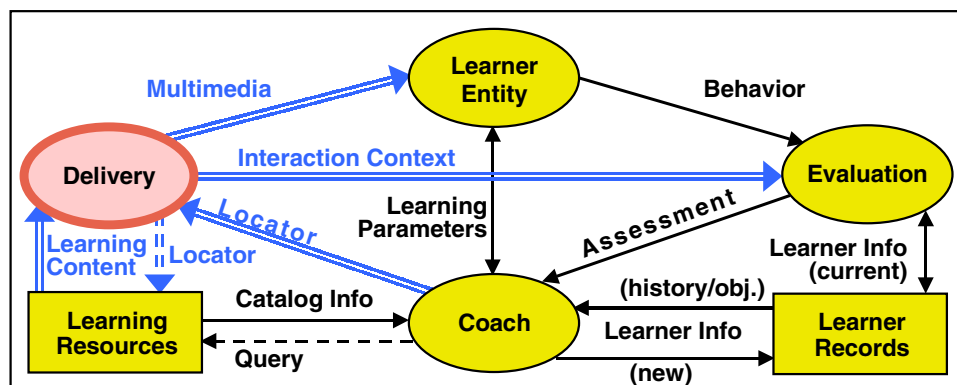


Figure 21—Delivery (a process): retrieves learning content from the learning resources based on the locator. Transforms the learning content into a multimedia presentation

5.18.1 Definition

An abstracted process that may transform information obtained via learning content into a presentation, which may be transferred to the learner entity via a multimedia data flow.

NOTE—The presentation may be static, interactive, collaborative, involve experiments and discovery, etc.

5.18.2 Inputs/outputs

- *Input*: May receive locators (via locator data flow) from the coach process.
- *Input*: May receive retrieved learning content (via learning content data flow) and may transform the learning content into a multimedia presentation for the learner entity.
- *Output*: May use locators (via locator control flow) to retrieve learning content from the learning resources data store.
- *Output*: May send multimedia (via multimedia data flow) to the learner entity.
- *Output*: May send the interaction context (via interaction context data flow) to the evaluation process.

NOTES:

1—Within an actual system implementation, the delivery process may be combined with the evaluation process to achieve the tight coupling necessary for responsive, interactive learning experiences.

2—The methods of implementing the delivery process may vary widely, e.g., presentation and questions, an intelligent tutoring system, video conferencing with a human tutor, and transforming an ontology (a conceptual model of the subject represented as generic learning content) into a presentation, among many other possibilities. The transformation methods of learning content to multimedia, are unspecified.

5.19 Interaction context

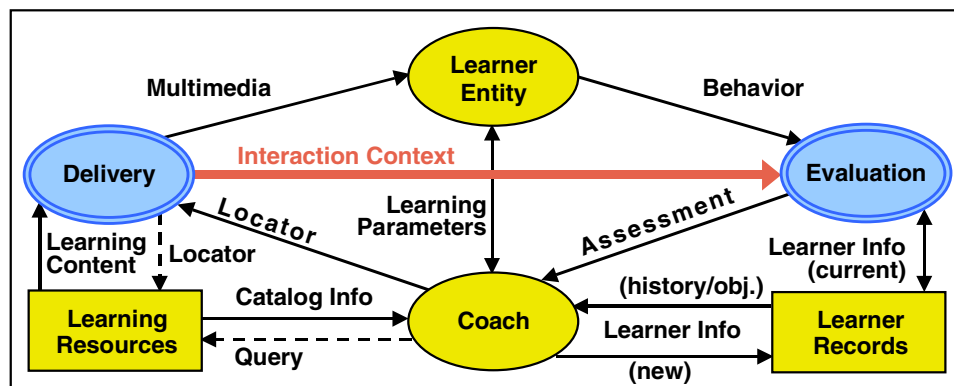


Figure 22—Interaction context (a data flow): the context of the learning content is sent to evaluation to correlate multimedia presentations with behavior responses

5.19.1 Definition

A 1-way data flow from the delivery process to the evaluation process that may provide information (e.g., a framework) necessary for the evaluation process to interpret the information supplied by the behavior data flow.

5.19.2 Information type

Contextual information, such as context names and pattern-responses.

NOTE—When the delivery process sends interactive multimedia to the learner entity, the evaluation process is expecting some behavioral response to the multimedia. The evaluation process may be unable to interpret the behavior without context, so the delivery process sends contextual information (e.g., possibly, the learning content itself) to the evaluation process so that the context of the learner entity's interactions may be understood.

Example: The learner entity is expected to select from a multiple choice question and the correct answer is “#2.”

5.20 Multimedia

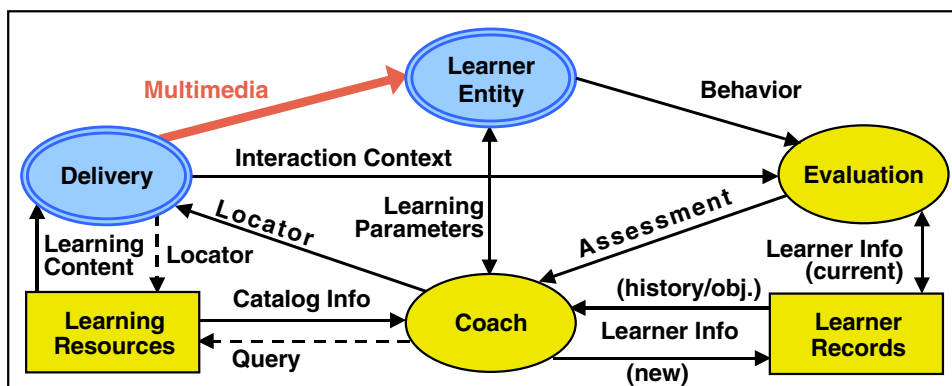


Figure 23—Multimedia (a data flow): the information (audio, video, graphics, text, etc.) sent to the learner entity

5.20.1 Definition

A 1-way data flow that represents the simultaneous presentation of several types of media, such as video, audio, and graphics from the delivery process to the learner entity.

5.20.2 Information type

Multimedia files and streams.

NOTE—In some learning technology systems, the implementation of the multimedia data flow is closely tied to the implementation of the behavior data flow to improve the responsiveness of the learning technology system.

5.21 Multiple roles for humans (informative)

This subclause is informative.

Humans may play one or more simultaneous roles in a learning technology system, i.e., not all components need be automated. Additionally, infrastructure may have multiple purposes. The following is a brief sampling of some possible mappings:

- *Learner (an individual) may map to:* learner entity (process), evaluation (process), learner records (data store), coach (process), learning resources (data store), and delivery (process).
- *Other learners may map to:* collaboration as a collective learner entity (process), and role playing in team learning as a collective learner entity (process).
- *Parent(s) may map to:* learner entity (process) as a surrogate or via collaboration or mentoring, evaluation (process), and coach (process).
- *Teacher(s) may map to:* learner entity (process) as a surrogate or via collaboration or mentoring, evaluation (process), coach (process), learning resources (data store), learner records (data store), and delivery (process).
- *Mentor(s) may map to:* learner entity (process) via collaboration or mentoring, evaluation (process), coach (process), and learning resources (data store).

- *Institution(s) may map to:* evaluation (process), learner records (data store), coach (process), learning resources (data store), and delivery (process).
- *Libraries may map to:* learning resources (data store).
- *Librarian(s) may map to:* query (control flow), catalog info (data flow), locator (both data flow and control flow), and learning resources (data store).
- *Classroom(s) may map to:* learning resources (data store) via experimentation and discovery, and delivery (process).
- *Web browser(s) may map to:* delivery (process), observable behavior (data flow), multimedia (data flow), and locator (data flow) via search engines and web retrieval.

5.22 Conceptual vs. actual implementations (informative)

This subclause is informative.

An important feature of the LTSA is the mapping of the “conceptual” system to the “actual” system. Actual systems, typically, are not organized as the individual LTSA components—there are commercial, business, and technical reasons for combinations or splittings of components. This is similar to the “architecture” of stereo component systems, e.g., a tuner, pre-amplifier, and amplifier are separate components but, typically, they are manufactured together as a “stereo receiver.” Annex F contains examples of mapping conceptual implementations (e.g., LTSA system components) to actual implementations.