IEEE P1484.1/D9-pre, 2001-11-15
Draft Standard for Learning Technology — Learning Technology Systems Architecture (LTSA)

Sponsored by the Learning Technology Standards Committee
of the IEEE Computer Society

IEEE Standards Activities Department
Standards Licensing and Contracts
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA

[Note: Information about IEEE LTSC P1484.1 can be found at:

http://ieee.ltsc.org/wg1

This document is also available at:

http://edutool.com/ltsa

This note will be removed upon reaching the final draft of this IEEE document.]

Introduction

(This introduction is not part of IEEE P1484.1, Learning Technology Systems Architecture.)

** TO BE SUPPLIED **

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

John Tyler, *Chair*
Brant Cheikes, *Chair (1996-1999)*
Frank Farance and Joshua Tonkel, *Technical Editors*

Carlos C. Amaro
Debbie Brown
Brant Cheikes
Mike Collett
Erik Duval
Eddy Forte
Jack Hyde
Igor Aizenberg
Anastasios Koutoumanos
Rolf Lindner
Bill McDonald
Bill Olivier
Claude Ostyn
Mike Petit
Dan Rehak
Tyde Richards
Kevin Riley
Jim Schoening
Kathy Sinitsa
Dan Suthers
John Tyler
Tom Wason
Eamonn Webster
The following persons were on the balloting committee: (To be provided by IEEE editor at time of publication.)

CONTENTS

1 Introduction

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, metadata, etc.. This Standard is pedagogically neutral, content-neutral, culturally neutral, and platform-neutral. This Standard (1) provides a framework for understanding existing and future systems, (2) promotes interoperability and portability by identifying critical system interfaces, and (3) incorporates a technical horizon (applicability) of at least 5-10 years while remaining adaptable to new technologies and learning technology systems. This Standard is neither prescriptive nor exclusive.

1.2 Purpose

In general, the purpose of developing system architectures is to discover 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 isn't a blue print 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., *critical interoperability interfaces and services are identified*.

The architectural framework developed in this standard should not address the specific details of implementation technologies (e.g., programming languages, authoring tools, or operating systems) necessary to create the system components, or the management systems (e.g., learning material lifecycle, quality assurance, access control, or user administration) necessary to manage a learning technology system, i.e., *the standard should facilitate the development of configuration guidelines for general learning technology systems*.

The standard shall identify the objectives of human activities and computer processes and their involved categories of knowledge, i.e., *it is possible to identify protocols and methods of cooperation and collaboration*.

1.3 Normative wording vs. informative wording

Note: This subclause is informative and not normative.

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, informative annexes, and sections labeled "*this clause/ subclause/ annex is informative and not normative*".

- **Informative wording.** This wording is helpful, but not required, for understanding this document. Clauses 1, 5, 7, and 8, and Annexes A-C and E-G are informative. Other informative wording is identified individually. 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)

This subclause is informative and not normative.

This Standard consists of 8 clauses and 8 annexes.

The following is an overview of each Clause and Annex.

- **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 [Architecture Overview]:** an overview of the architecture described in terms of refinement layers. Each layer is summarized.
- **Clause 6 [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.
- **Clause 7 [Implementation Perspectives and Priorities]:** the next refinement layer, learning systems from a variety of perspectives.
- **Clause 8 [Operational Components and Interoperability]:** the bottom refinement layer, generic components and building blocks of information technology systems as applied to learning technology.
- **Annex A [Bibliography, informative]:** references to related documentation.
- **Annex B [Learner-Related Influences, informative]:** the top two refinement layers of the architecture. This Annex describes the learning experience as interactions between the learner and his/her environment, and the learner-related design issues. This layer only applies to information technology — no pedagogy is implied.
- **Annex C [Illustrations of Stakeholder Mappings]:** examples of the fourth refinement layer, learning systems from a variety of perspectives.
- **Annex D [Pro Forma Implementation Conformance Statement, normative]:** the form that must be completed so that implementations can claim conformance to this Standard.
- **Annex E [Illustrations of Conforming Implementations, informative]:** a small collection of implementations that illustrate how Clause 4, Conformance, is applied.
- **Annex F [Conformance to IEEE 1471, informative]:** how LTSA conforms to IEEE 1471 for architectural description.

- **Annex G [Methodology, informative]:** the process used to develop this Standard. Note: This Annex will be removed prior to the publishing of this Standard.
- **Annex H [Document Development, informative]:** a revision history and list of all outstanding issues with respect to this document. Note: This Annex will be removed prior to the publishing of this Standard.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. IEEE and members of ISO and IEC maintain registers of currently valid Standards.

- IEEE 1484.3, Learning Technology Glossary.

3 Definitions

3.1 Definitions incorporated via normative reference

Note: The following is a brief glossary of terms used in this Standard. This glossary is in the process of being harmonized with the IEEE LTSC 1484.3 Glossary document, available at:

http://ltsc.ieee.org/wg3

This glossary serves the purpose of defining certain key terms within the context of this Standard. This Standard will be improved as terminology in education research, training, learning technology, and information technology converges.

For the purposes of this standard, the following terms and definitions apply. The *IEEE Dictionary of Electrical and Electronics Terms* should be referenced for terms not defined in this clause.

3.2 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**.

3.3 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.**

3.4 abstraction-implementation layer

See: **refinement layer.**

3.5 actual implementation

See: **implementation (A).**

3.6 administrator

This subclause is informative and not normative.

A person responsible for purchasing systems, managing systems, or managing institutions.

3.7 binding

An application or mapping from one framework or specification to another.

3.8 bus

A collection of subsystems that communicate with a common control protocol within a common namespace but, possibly, varying data protocols among members.

3.9 bus notation

A descriptive technique that decomposes systems into members (subsystems) connected to a bus. Bus notation is useful when there are a large number of subsystems, the functions or roles are undefined or changing over time, and the connections between subsystems are dynamic connections or on-demand connections. *See also:* **system notation; text description.**

3.10 coding

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

3.11 collaboration

Communication among the individual learners of the learner entity.

3.12 conceptual implementation

See: **implementation (B).**

3.13 control flow

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

3.14 control protocol

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

3.15 control store

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

3.16 data flow

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

3.17 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. *See also:* **control process**.

3.18 data protocol

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

3.19 data transformation process

See: **data process**.

3.20 data store

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

3.21 design priority

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

3.22 developer

This subclause is informative and not normative.

A creator of learning content and/or software.

3.23 distance learning system

This subclause is informative and not normative.

A learning technology system that (1) has at least one LTSA system component flow (learning preferences, 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: network delays, reliability, bandwidth, responsiveness, availability.

3.24 distributed system

This subclause is informative and not normative.

A collection of geographically diverse components that act as a single system.

3.25 encoding

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

3.26 flow

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

3.27 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.**

3.28 implementation under test

In conformance testing, the system that is being tested.

3.29 interaction (learning technology)

An information exchange between a learner and a system.

3.30 learner

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

3.31 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.**

3.32 learning experience

This subclause is informative and not normative.

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.

3.33 nomadic learning

This subclause is informative and not normative.

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.

3.34 nomadicity

This subclause is informative and not normative.

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

3.35 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**.

3.36 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**.

3.37 process

An active system component that transforms its inputs into outputs.

3.38 secondary design issue

After the primary design issue(s), the next significant focus of technical development. *See also:* **design priority; primary design issue**.

3.39 stakeholder

This subclause is informative and not normative.

(**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.

3.40 store

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

3.41 system component

In system notation, a process, a store, or a flow.

3.42 system notation

A descriptive technique that decomposes systems into subsystems of processes and stores connected by flows. *See also:* **bus notation; text descrption.**

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.

3.43 text description

A method of identifying system and subsystem functionality that uses words instead of graphics. *See also:* **bus notation; 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.

3.44 two-way flow

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

3.45 Acronyms and abbreviations

- 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 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, Pro Forma Implementation Conformance Statement, contains a printed copy of the ICS form. Annex E, Illustrations of Conforming Implementations, 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 (LE)
- LTSA behavior data flow from learner entity to assessment (B)
- LTSA evaluation process (E)
- LTSA assessment data flow from evaluation to coach (A)
- LTSA learner information data flow between evaluation and learner records (L1)
- LTSA learner records data store (R)
- LTSA learner information data flow from learner records to coach (L2)
- LTSA learner information data flow from coach to learner records (L3)
- LTSA learning preferences between learner entity and coach (L4)
- LTSA coach process (C)
- LTSA query control flow from coach to learning resources (Q)
- LTSA learning resources data store (LR)
- LTSA catalog information data flow from learning resources to coach (QI)
- LTSA locator data flow from coach to delivery (L5)
- LTSA delivery process (D)
- LTSA locator control flow from delivery to learning resources (L6)
- LTSA learning content data flow from learning resources to delivery (LC)
- LTSA interaction context data flow from delivery to evaluation (IC)
- LTSA multimedia data flow from delivery to learner entity (M)

The ICS shall identify 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 Clause 7, Stakeholder Perspectives and Priorities.

4.2 Conformance label

A conformance label may be used as a shorthand for an ICS. A conformance label, if used, shall be in one the following formats:

- "**`Conforms to IEEE 1484.1:2001 LTSA all components`**"
- "**`Conforms to IEEE 1484.1:2001 LTSA components X1-X2-X3-…`**"
- "**`Conforms to IEEE 1484.1:2001 LTSA components except X1-X2-X3-…`**"

where

- The first format indicates that all LTSA system components conform in the implementation.
- The second format indicates only certain LTSA system components conform in the implementation.
- The third format indicates all components except certain LTSA system components conform in the implementation.
- The list **`X1-X2-X3-…`** is a list of hyphen-separated LTSA system component abbreviations, as described above.

Example 1: The conformance label "**`Conforms to IEEE 1484.1:2001 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 "**`Conforms to IEEE 1484.1:2001 components E-A-C-L5-D-IC`**", see also Annex E, subclause 13.2, Tight Component Integration: Web Browser.

Example 3: The conformance label for a learning media "player", e.g., just insert a CD-ROM, might be "**`Conforms to IEEE 1484.1:2001 except components LR`**" — the learning resources are the only component missing.

5 Architecture overview

This Clause is informative and not normative.

Five refinement layers of architecture are specified, but only layer 3 (system components) is normative in this Standard. This architecture is applicable to a broad range of learning scenarios. These refinement layers are called, from highest to lowest levels:

- **Learner and Environment Interactions (informative):** Concerns the learner's acquisition, transfer, exchange, formulation, discovery, etc. of knowledge and/or information through interaction with the environment.
- **Learner-Related Design Features (informative):** Concerns the effect learners have on the design of learning technology systems.

- **System Components (normative):** Describes the component-based architecture, as identified in human-centered and pervasive features.
- **Implementation Perspectives and Priorities (informative):** Describes learning technology systems from a variety of perspectives by reference to subsets of the system components layer.
- **Operational Components and Interoperability — codings, APIs, protocols (informative):** Describes the generic "plug-n-play" (interoperable) components and interfaces of an information technology-based learning technology architecture, as identified in the stakeholder perspectives.

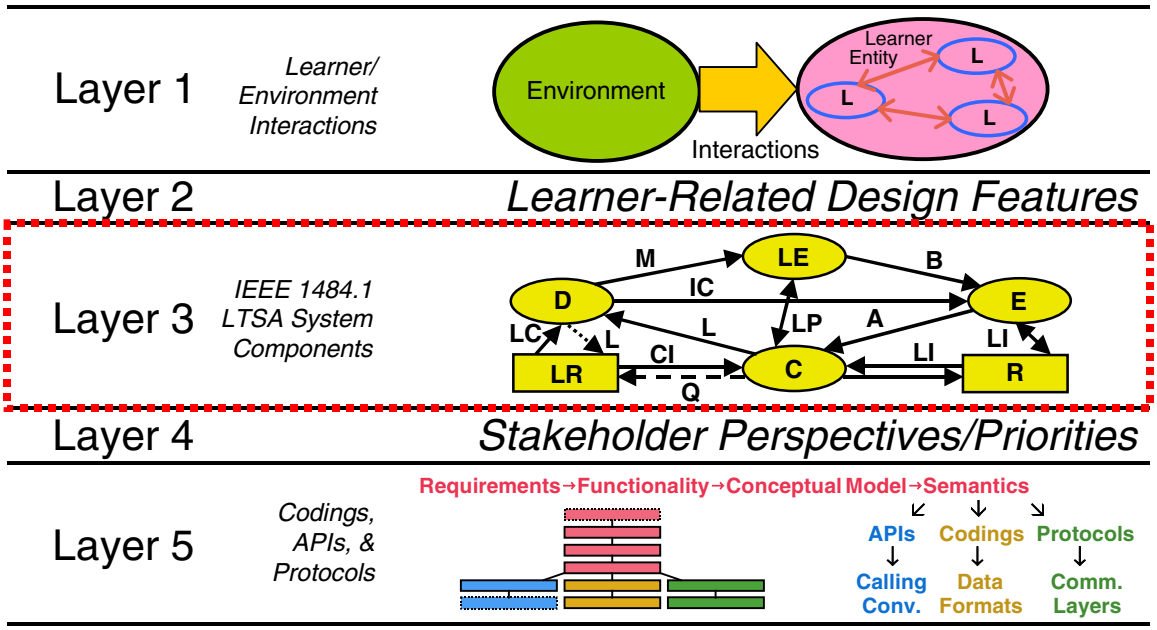


Figure 1. The LTSA abstraction-implementation layers. Only layer 3 (system components) is normative in this Standard.

Developer overview (informative)

The five abstraction-implementation layers identify design priorities, i.e., the ordering of design issues from most important to least important. Intuitively, developers understand that, for example, the human features of learning technology systems (layer 2) have a more pervasive effect on system design than, for example, the particular multimedia format (layer 5) — multimedia formats are small-scoped, "swappable" features. This Clause explains how the methodology was applied.

Layer 3 (system components) may be used to analyze interoperability requirements among major subsystems in learning technology systems.

Administrator overview (informative)

The five layers represent five independent areas of technical analysis. For example, it is possible to discuss an abstraction (e.g., the LTSA system components — layer 3), independently of an implementation (e.g., the coding, APIs, and protocols of an actual implementation —layer 5). In other

words, even though layer 3 contains components such as "evaluation" and "coach", these components are "conceptual" in that there is no requirement for separable, identifiable components called "evaluation" and "coach" in actual implementations.

Layer 3 (system components) is the only "normative" (required) layer within this Standard.

Teacher and learner overview (informative)

The five layers of the LTSA help separate the "big picture" from the "details". The use of layers helps the reader understand the problem "step by step".

Layer 3 (system components) is the only "normative" (required) layer within this Standard.

5.1 Refinement layers

The Learning Technology Systems Architecture is described in five successive refinement layers from highest to lowest. Each layer describes a system at a different level. The lower layers are implementations of the higher layers; the higher layers are abstractions of the lower layers.

5.2 Learner and environment interactions

This refinement layer focuses on the highest level (most generic) functionality from an information technology perspective: the learner has new or different knowledge after a learning experience. In information technology, this is diagrammed as one subsystem (environment) transferring information to another subsystem (the learner), i.e., an interaction. See Annex B, Learner-Related Influences, for additional explanation. The Learner-Environment Interactions diagram is *not* intended to represent current theories of learning or a learning process. It represents the *information technology issues* of learning technology systems and is useful for common, well-understood software engineering analysis and design techniques. *For the purposes of this Standard, the primary focus is information technology.*

5.3 Learner-related design features

This refinement layer concerns the learners' effect on the design of learning technology systems. See Annex B, Learner-Related Influences, for additional explanation.

5.4 System components

The LTSA identifies four processes: learner entity, evaluation, coach, and delivery process; two stores: learner records and learning resources; and thirteen information flows among these components: behavioral observations, assessment information, learner information (three times), query, catalog info, locator (twice), learning content, multimedia, interaction context, and learning preferences. See Annex G, Methodology, for an explanation of the notation used and for illustrations of the concepts of "processes", "stores", and "flows".

Briefly, the overall operation has the following form: (1) the learning styles, strategies, methods, etc., are negotiated among the learner and other stakeholders and are communicated as learning prefer-

ences; (2) the learner is observed and evaluated in the context of multimedia interactions; (3) the evaluation produces assessments and/or learner information; (4) the learner information is stored in the learner history database; (5) the coach reviews the learner's assessment and learner information, such as preferences, past performance history, and, possibly, future learning objectives; (6) the coach searches the learning resources, via query and catalog info, for appropriate learning content; (7) the coach extracts the locators from the available catalog info and passes the locators to the delivery process, e.g., a lesson plan; and (8) the delivery process extracts the learning content from the learning resources, based on locators, and transforms the learning content to an interactive multimedia presentation to the learner.

Multiple roles

In a given learning situation, there is not necessarily a one-to-one correspondence between system components and individuals. An individual might represent more than one system component in a given learning situation, e.g., the individual who represents the learner entity might also represent the coach in a self-paced learning environment. Likewise, more than one individual might represent a single system component in a given learning situation, e.g., the learner entity might be represented by several individuals learning collaboratively or as a team.

Multiple learning experiences

Although a single set of components is described, there might be several different kinds of learning experiences occurring simultaneously in a given learning situation. For example, a course offered jointly by the mathematics and computer science departments may foster two different kinds of learning experience, yet there is only one "physical" course that the learner attends. Another example would be the student teacher in the classroom: the "learners" are learning (through their own learning experiences), and the teacher is a "learner", too (a different learning experience).

5.5 Stakeholder perspectives

Annex C, Illustrations of Stakeholder Mappings, contains an informative summary of over 120 stakeholder perspectives — they are formulated, consolidated, and reviewed from the standpoint of the LTSA. The list of perspectives isn't exhaustive. The results of this analysis have been: (1) verification and validation of the LTSA components in significant systems, stakeholders, and industries, (2) discovery of which LTSA components are emphasized and de-emphasized in different systems, stakeholders, and industries, and (3) indication of varying priorities among higher-level and lower-level design issues.

The stakeholder perspectives layer is considered a separate refinement layer because this layer of granularity addresses a particular design issue: which perspective, view, or subset is relevant to the lower-level design.

5.6 Operational components and interoperability

The major areas of operational components and interoperability are identified via several notations, but generically described as codings, APIs, and protocols. Knowing which interoperability standards (codings, APIs, and protocols) are in use can increase our understanding of a system and help us know

about its potential interoperability, but systems must be integrated and configured properly to achieve true interoperability among themselves. Clause 8, Operational Components and Interoperability, provides an overview of how technical standards can be related to LTSA and the development process that creates and harmonizes the technical work. The specification of actual coding, API, protocol, etc., standards is outside the scope of LTSA.

6 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, Methodology, describes this notation.

The LTSA system components identify the *critical* 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 6.22, Conceptual vs. Actual Implementations, and Annex E, Illustrations of Stakeholder Mappings, 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".

Developer overview (informative)

Each LTSA system component is described in technical detail: processes and stores are described by their functionality and their interfaces; flows are described by their connectivity (the components they connect) and the type of information across the flow.

Administrator overview (informative)

Each LTSA system component is described. This Clause contains the technical description of the critical interfaces for learning technology systems. System and component interoperability is greatly enhanced by the identification and establishment of the critical interfaces.

Teacher and learner overview (informative)

This Clause describes the technical details of each LTSA system component. This Clause provides a precise technical specification.

6.1 Component organization

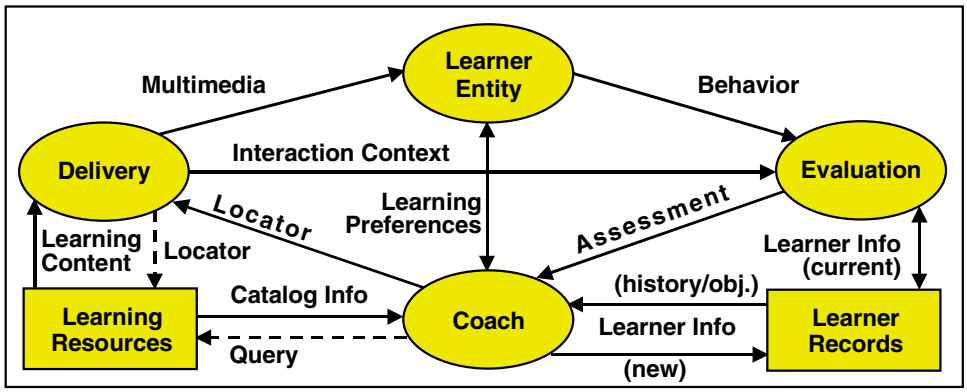


Figure 2. The LTSA system components.

The LTSA system components are:

- **Processes**: learner entity, evaluation, coach, delivery.
- **Stores**: learner records, learning resources.
- **Flows**: learning preferences, 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").

6.2 Learner entity

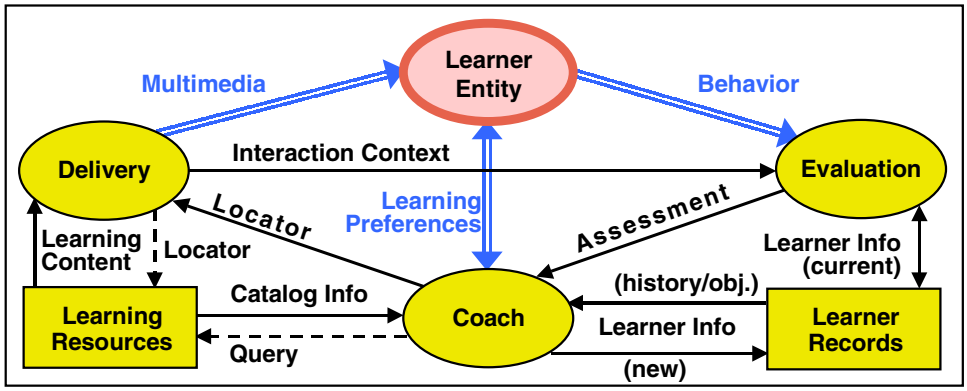


Figure 3. Learner Entity (a process): The abstraction of the human learner.

Definition

A conceptual 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.

Inputs/Outputs

- (Input) The learner entity may receive a multimedia presentation.
- (Output) The learner's behavior may be observed.
- (2-Way) The learning preferences are negotiated with the coach.

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

Note 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.

6.3 Learning preferences

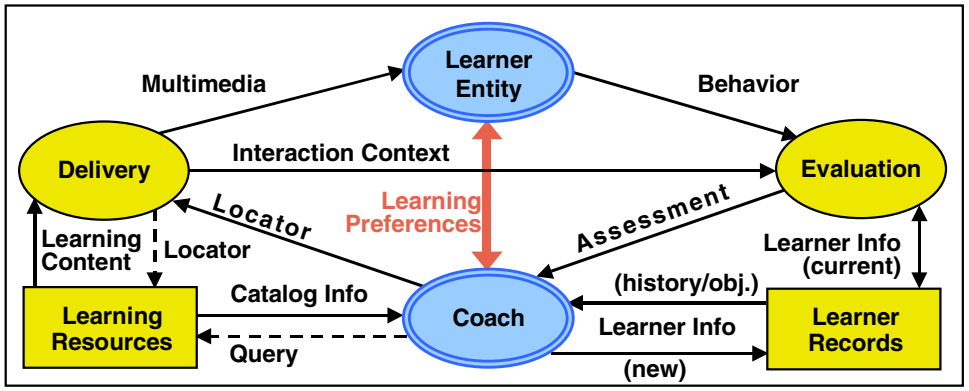


Figure 4. Learning Preferences (a data flow): All interested parties contribute to learning preferences.

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 preferences.

Information Type

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

6.4 Behavior

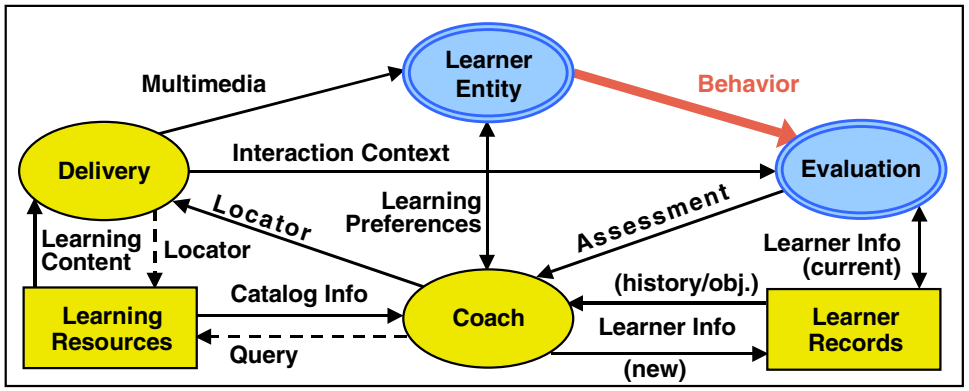


Figure 5. Behavior (a data flow): The coding and encoding of learner entity's actions.

Definition

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

Information Type

Behavior information 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.

6.5 Evaluation

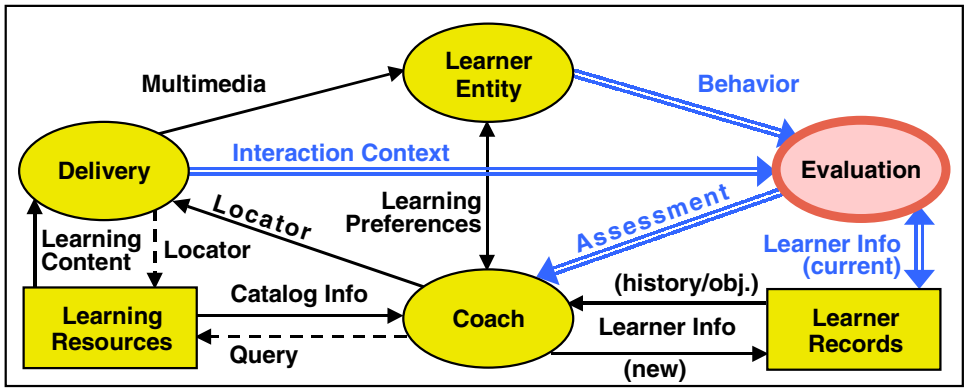


Figure 6. Evaluation (a process): The processing of behavior information to produce assessment and learner information.

Definition

A conceptual process that may produce measurement(s) of the learner entity.

Inputs/Outputs

- (Input) The learner entity's observable behavior
- (Input) The interaction context may provide context to the learner entity's behavior to determine the appropriate evaluation.
- (Output) Assessment information may be sent to the coach.
- (Input/Output) Learner information may be retrieved and stored 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 keystrokes "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).

6.6 Learner information stored/retrieved by evaluation

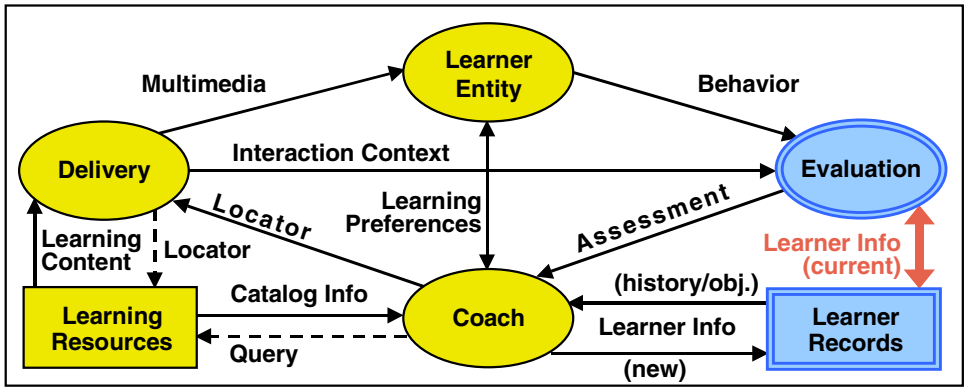


Figure 7. **Learner information (a data flow): Output from evaluation, to be stored as learner entity history information in learner records.**

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.

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".

6.7 Learner records

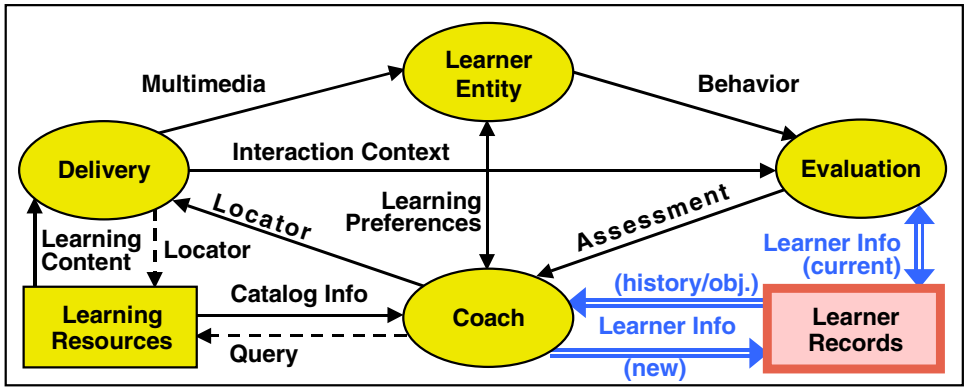


Figure 8. **Learner Records (a data store): Storage and retrieval of learner information of the past (e.g., history), present (e.g., "suspends", current assessment), and future (e.g., objectives).**

Definition

A data store of learner information, such as performance, preference, and other types of information. The learner records may store/retrieve information 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).

Store/Retrieve

- (Store/Retrieve) The evaluation process may store/retrieve learner information (e.g., grades on lessons).
- (Store/Retrieve) The coach process may store/retrieve learner information (e.g., performance, preference, and other types of learner information).

6.8 Learner information received by system coach

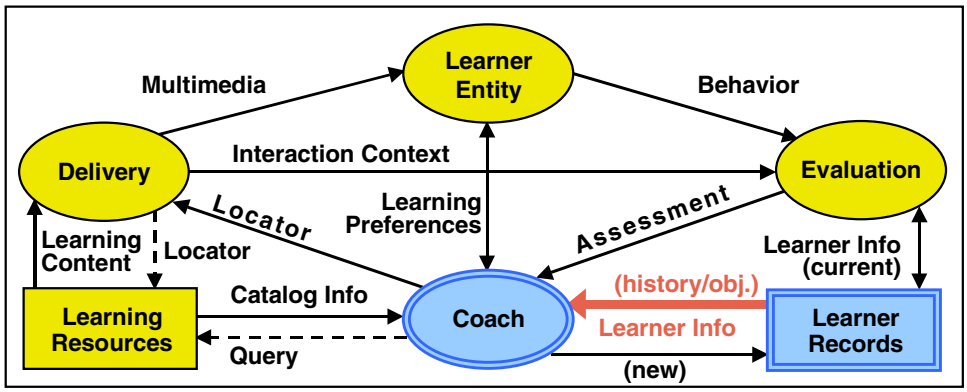


Figure 9. Learner information (a data flow): Coach retrieves learner records, returned as learner information.

Definition

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

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.

6.9 Learner information stored by system coach

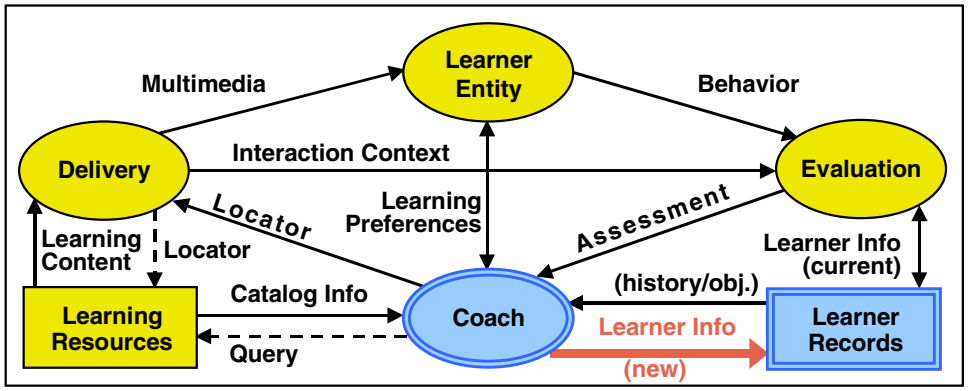


Figure 10. Learner information (a data flow): Coach stores assessments and certifications in learner records.

Definition

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

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 learner records.

6.10 Assessment information

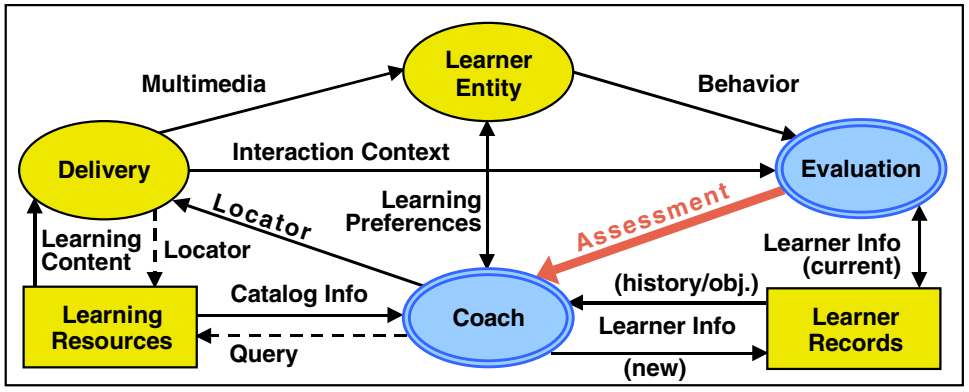


Figure 11. Assessment information (a data store): Output from evaluation, represents learner entity's "current state".

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 component to determine optimal learning experiences.

Information Type

Assessment information may include performance information and other learner information.

6.11 Coach

Definition

A conceptual process that may incorporate information from several sources, such as the learner (learning preferences), evaluation process (assessment information), learner records (performance, preference, and other learner information), and learning resource (query and catalog information), and may use this information to search (query) and select (locator) learning content (via the delivery process and multimedia) for learning experiences. The coach process is defined in 5 steps. These steps may be performed in any order. Steps may be omitted during the learning experiences.

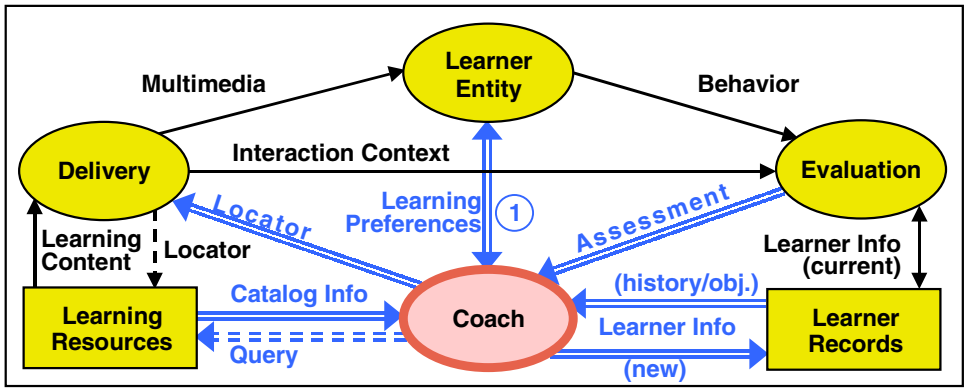


Figure 12. Coach (a process), Part 1: Negotiates/exchanges learning preferences for optimum learning experience.

Inputs/Outputs

- (2-Way) The learning preferences 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 coach (two-way negotiation), or an external authority (e.g., parent, teacher, institution, or content developer).

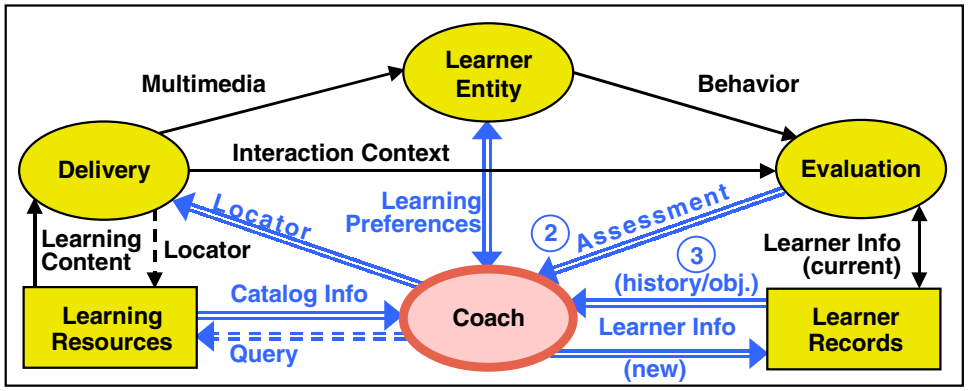


Figure 13. Coach (a process), Parts 2 and 3: Receives current assessment information from evaluation. Searches and retrieves learner information relevant to the current learning experience.

Inputs/Outputs

- (Inputs) The current assessment information from the evaluation process.
- (Inputs) Learner information from the learner records.

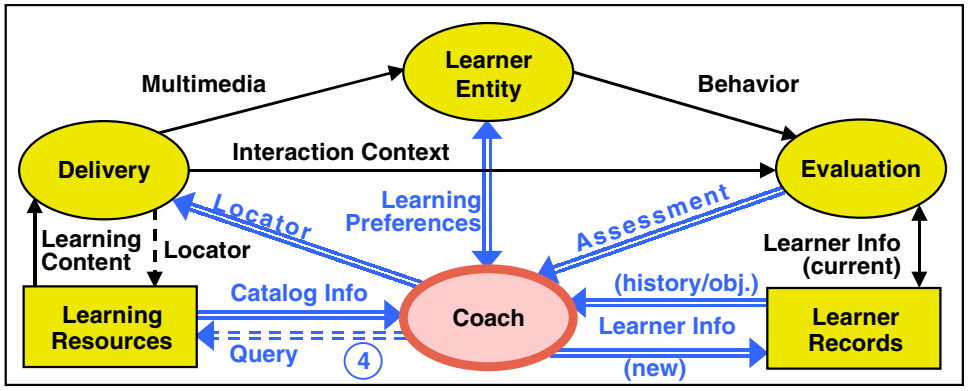


Figure 14. Coach (a process), Part 4: Searches learning resources via queries for appropriate learning content. Learning resources returns "found" catalog info (a.k.a., learning object metadata) that matches the query.

Inputs/Outputs

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

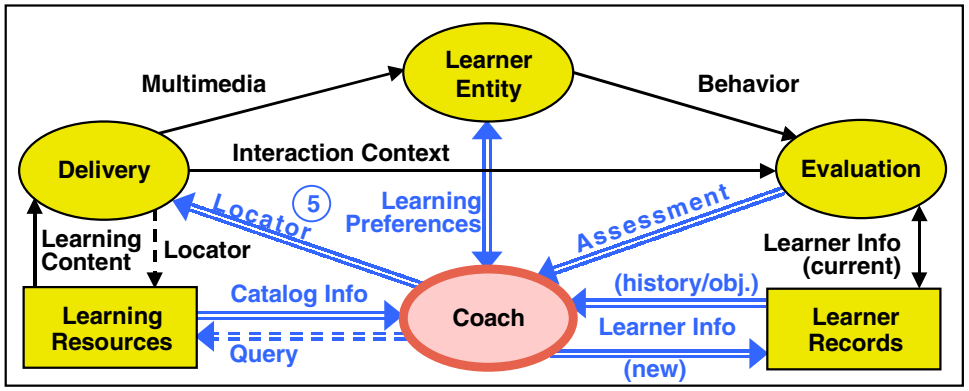


Figure 15. **Coach (a process), Part 5: Extracts the Locators (e.g., URLs) from the "found" catalog info (learning object metadata). Sends the Locators to the delivery process to direct the learning experience.**

Inputs/Outputs

- (Output) Locators (e.g., a lesson plan, pointers to content) may be sent to the delivery process.

6.12 Query

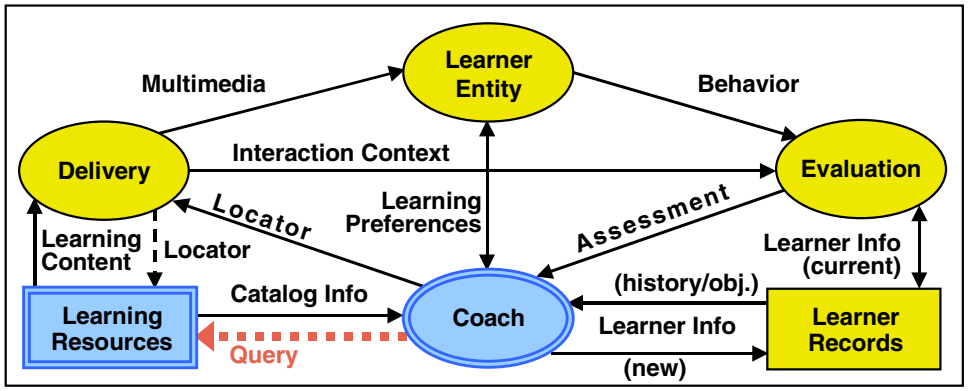


Figure 16. **Queries (a control flow): The "search request" when looking for appropriate learning content in the learning resources.**

Definition

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

Note 1: 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.

Information Type

A query is a set of search criteria.

Note 2: The queries may specify search criteria based on, in part, learning preferences, assessment information, and learner information. Typically, the queries are searching for learning content that is appropriate for the learner entity.

6.13 Learning resources

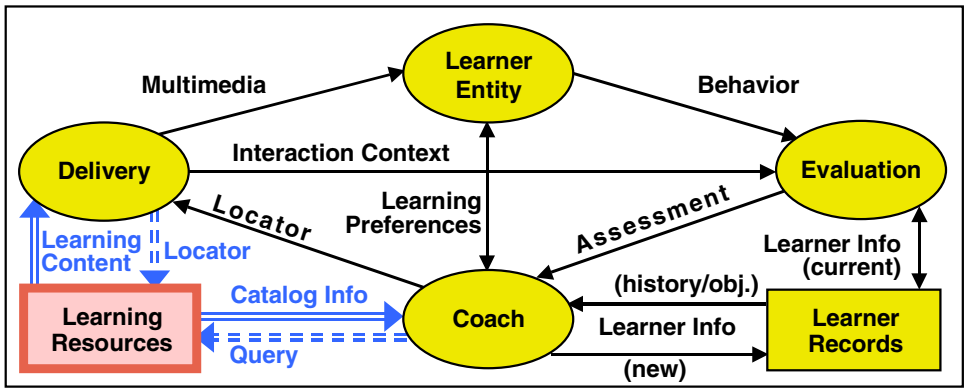


Figure 17. Learning Resources (a data store): A database that represents "knowledge", information, and other resources uses in the learning experiences. The learning resources may be represented as presentations, tutorials, experiments, lessons, etc..

Definition

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

Store/Retrieve

- (Retrieve) May be searched by queries.
- (Retrieve) The matching information is returned as catalog info, 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) are extracted from the catalog info.
- (Retrieve) The locators may be used by the delivery process to retrieve learning content.

Example: A query on a topic in chemistry (specified as a query) might return a set of catalog info that include 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.

6.14 Catalog information

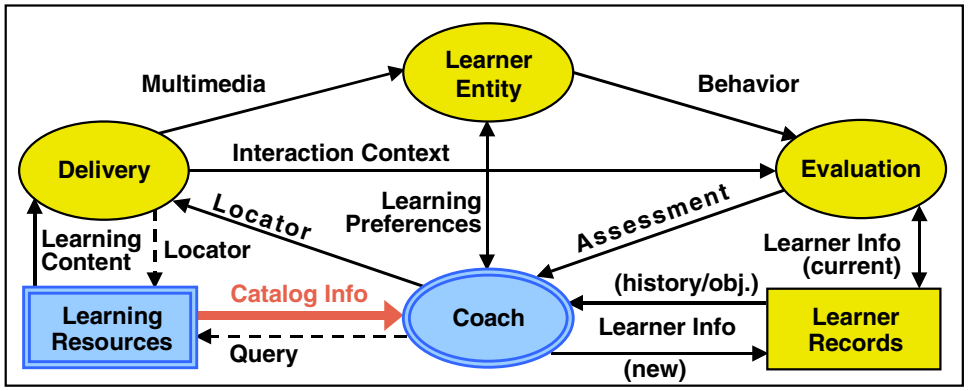


Figure 18. Catalog Information (a data flow): Represents "card catalog" information about learning content in the learning resources. Catalog info is also known as "learning object metadata".

Definition

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

Information Type

Information that describes learning resources.

Note 1: The catalog information is also known as "learning object metadata". Catalog information is similar to "card catalog" entries in a library.

Note 2: Metadata is may be used in web content for facilitating searches. However, web content metadata is inadequate for learning content because learning content requires more search criteria (e.g., pre-requisites, co-requisites, learning style) than what is provided for in web content (e.g., title, subject, author, keywords).

6.15 Locator sent by coach

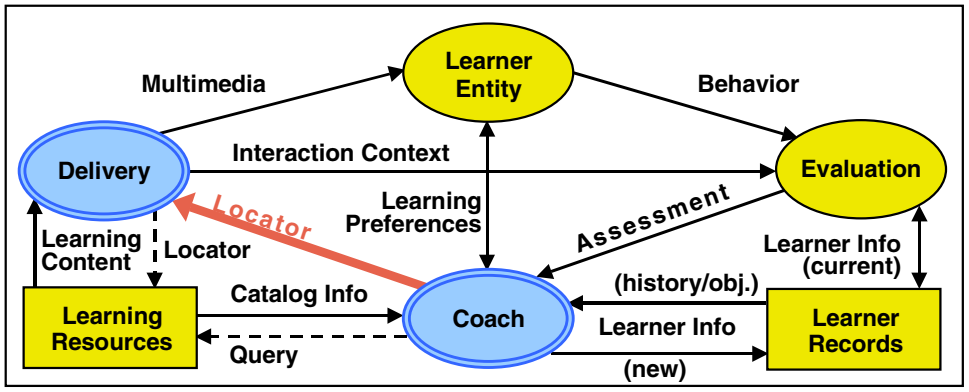


Figure 19. Locator (a data flow): Represents the "call number" of the learning content in the learning resources. Web-based systems typically use URLs for locators.

Definition

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

Information Type

Identifiers or pointers.

Note 1: Using the library analogy, locators are similar to "call numbers" in a card catalog system.

Note 2: This locator (from coach process to delivery process) is the same information type as the locator from the delivery process to the learning resources.

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

6.16 Locator sent by delivery

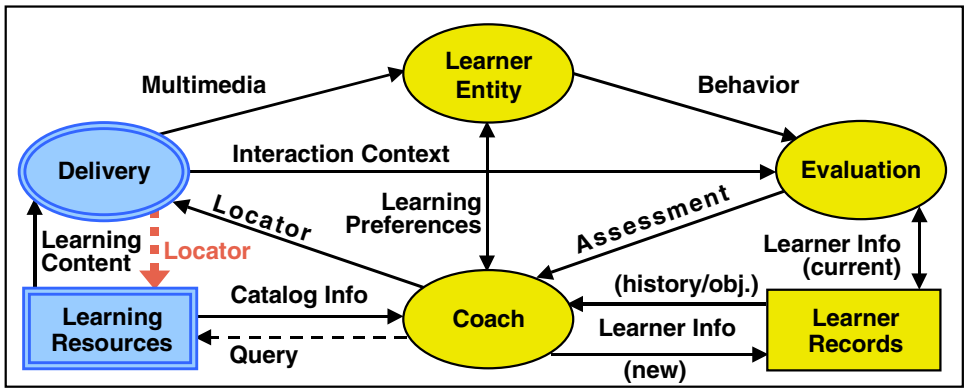


Figure 20. Locator (a control flow) sent by delivery.

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 the previous subclause, the locator is a data flow in the context of coach to delivery. The notion of data flow vs. control flow is contextual (see Annex G, Methodology, subclause 15.7.2.6, Data vs. Control).

Information Type

Identifiers or pointers.

Example: A web URL.

6.17 Learning content

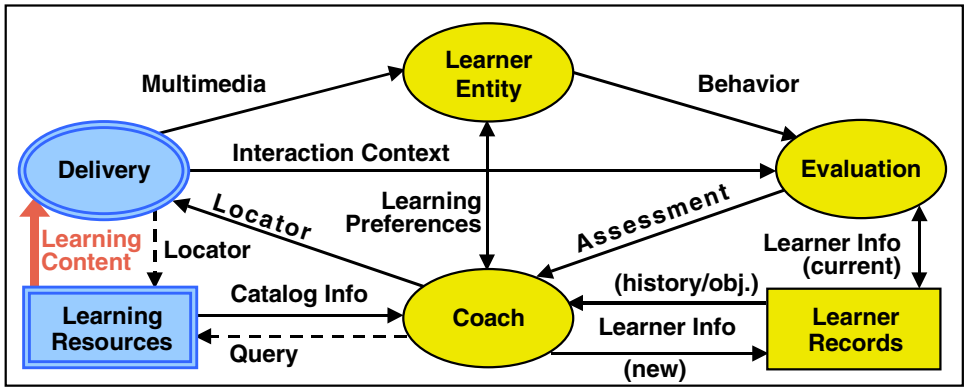


Figure 21. Learning Content (a data flow): The coding of materials from the learning resources. Learning content may be lessons, presentations, tutorials, tutors, experiments, etc..

Definition

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

Note: The learning content may be identified by the locator, retrieved by the learning resources, and transformed by the delivery system into an interactive multimedia learning experience.

6.18 Delivery

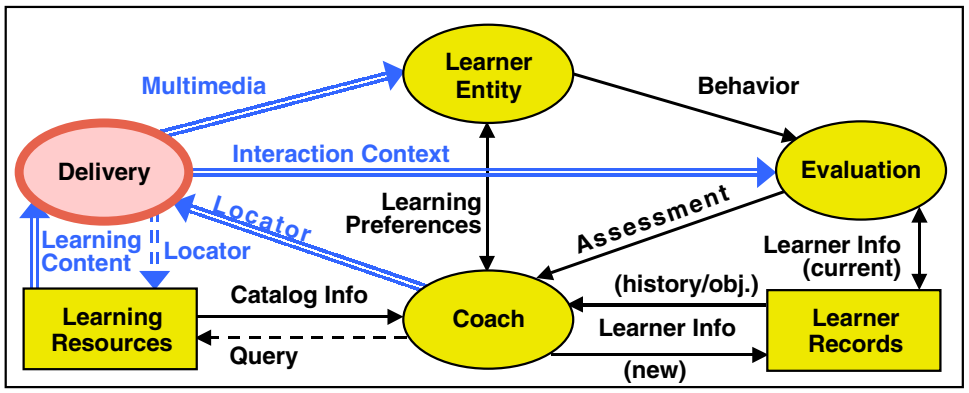


Figure 22. Delivery (a process): Retrieves learning content from the learning resources based on the locator. Transforms the learning content into a multimedia presentation.

Definition

A conceptual process that may transform information obtained via learning content into a presentation, which may be transferred to the learner entity via a multimedia.

Note 1: The presentation may be static, interactive, collaborative, involve experiments and discovery, etc..

Inputs/Outputs

- (Input) May receive locators from the coach.

- (Output) May use locators to retrieve learning content from the learning resources.
- (Input) May receive retrieved learning content and may transform the learning content into a multimedia presentation for the learner entity.
- (Output) May send multimedia to the learner entity.
- (Output) May send the interaction context to the evaluation process.

Note 2: 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.

Note 3: 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.

6.19 Interaction context

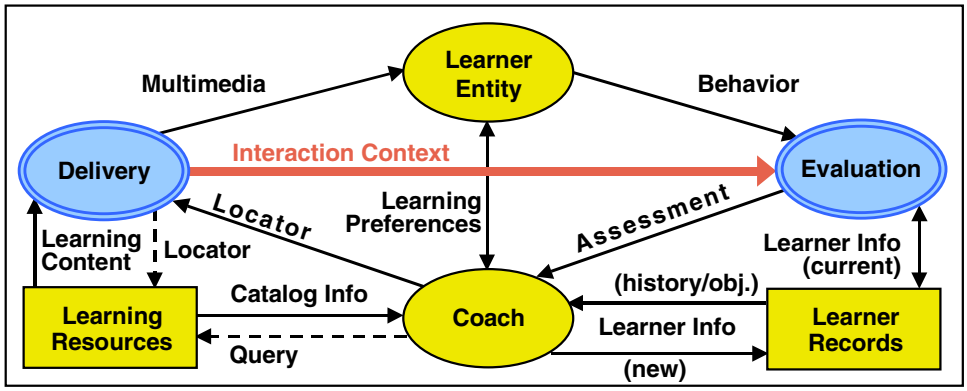


Figure 23. Interaction Context (a data flow): The context of the learning content is sent to evaluation to correlate multimedia presentations with behavior responses.

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.

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".

6.20 Multimedia

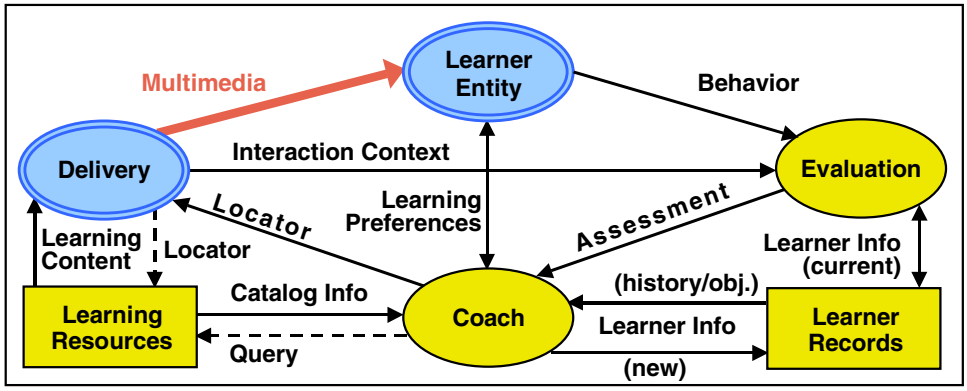


Figure 24. Multimedia (a data flow): The information (audio, video, graphics, text, etc.) sent to the learner entity.

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.

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.

6.21 Multiple roles for humans

This subclause is informative and not normative.

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.

6.22 Conceptual vs. actual implementations

This subclause is informative and not normative.

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 E, Illustrations of Conforming Implementations, contains examples of mapping conceptual implementations (e.g., LTSA system components) to actual implementations.

7 Stakeholder perspectives and priorities

This Clause is informative and not normative.

Overview

Each stakeholder has an important, legitimate perspective. However, each stakeholder has a different perception of learning technology systems. The stakeholder perspectives and priorities are diagrammed using notation that is specific to LTSA.

7.1 Abstraction-implementation boundary

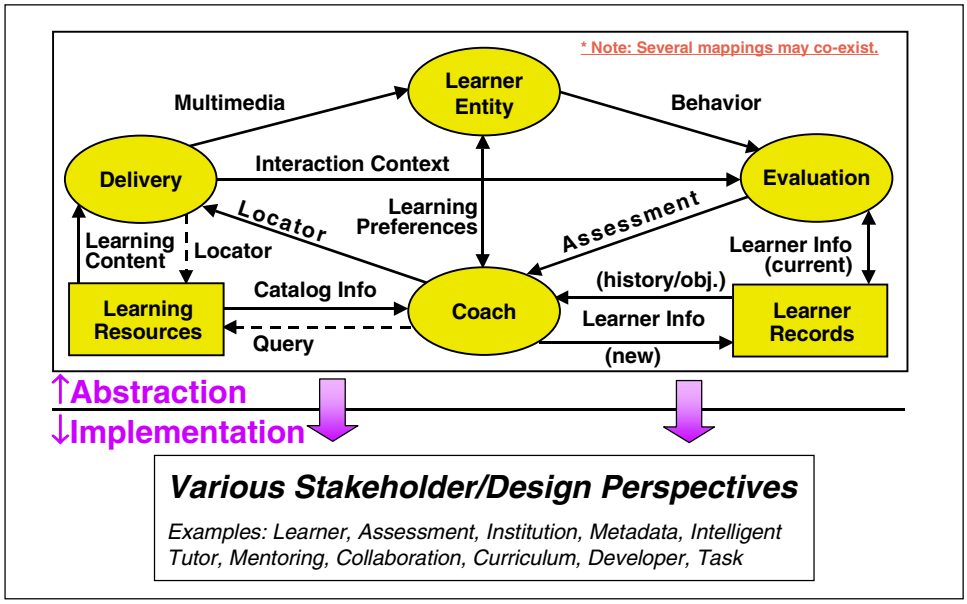


Figure 25. LTSA system components (the abstraction) are implemented as various stakeholder perspectives.

The LTSA system components are an abstraction that is implemented in various stakeholder perspectives. The stakeholder's perspective (layer 4) is a subset of LTSA system components that represents an implementation of the LTSA layer 3.

7.2 Notation for perspectives and priorities

The identification of stakeholder perspectives and their priorities requires an analytical method. The results of the analysis may identify:

- The LTSA system components that are of interest to the stakeholder.
- The relative importance of the LTSA system components.
- The critical interoperability interfaces of the stakeholder.

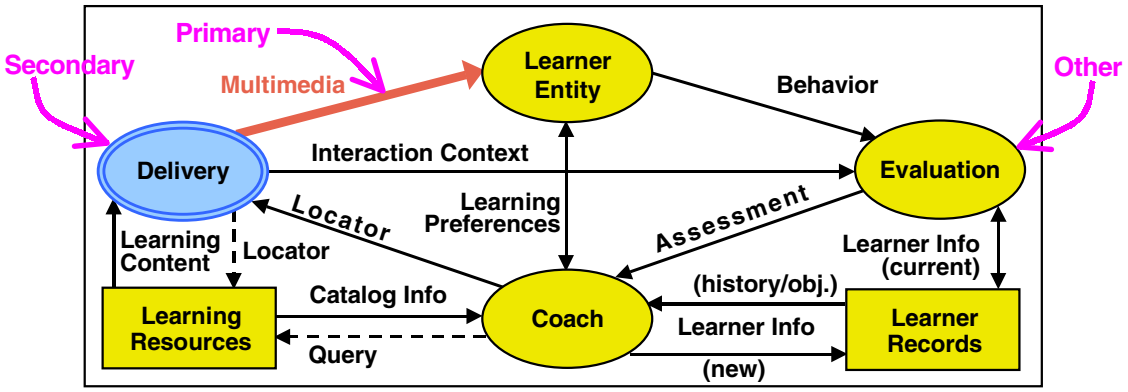
The LTSA uses a particular notational convention for identifying the stakeholder perspectives and priorities: each stakeholder is represented by a diagram employing a subset of the LTSA components, each with its own emphasis and de-emphasis on particular components. The *emphasis and de-emphasis* (primary, secondary, and other design issues) *reflect the technology issues, not the pedagogy.*

The primary design priorities are shown in red and **bold**. The secondary priorities are shown in blue and double lines. LTSA components that are not primary or secondary, or are not applicable are shown without distinction, with normal weight, or in olive.

Example stakeholder mapping diagram and table: digital audio and digital video

The following is a sample stakeholder called "digital audio and video". This stakeholder is mostly concerned about the use of digital audio and video in learning technology systems. The following diagram and table summarize this particular stakeholder.

Note: The priorities for this stakeholder are different than other stakeholders, e.g., LTSA delivery is a secondary design priority for this stakeholder, but might be a primary design priority for other stakeholders.



Summary	The digital audio and digital video components of multimedia.
LTSA Design Priorities	Primary: The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia.
Non-LTSA Focus	Primary: Delivery of multimedia across networks of varying capabilities. Multimedia presented on hardware with varying capabilities.
	Secondary: Common presentation of varying multimedia lessons and learning content.
Other Issues	Multi-platform portability.

In the sample mapping above, the following features are highlighted:

- **Primary (diagrammed):** Components that are primary design priorities are colored in red and use bold lines. Each primary design component has an additional explanation in the entry labeled "LTSA Design Priorities, Primary". These components may be related to the entry labeled "non-LTSA issues".
- **Secondary (diagrammed):** Components that are secondary design priorities are colored in blue and are use double lines. Each secondary design component has an additional explanation in the entry labeled "LTSA Design Priorities, Secondary". These components may be related to the entry labeled "non-LTSA issues".
- **Other (diagrammed):** Components that are tertiary design priorities or not applicable are colored in olive or black and are not emphasized (no bold lines, no double lines).
- **Summary:** A one or two line summary of the stakeholder.

- **LTSA Design Priorities:** This entry is divided into two subentries: primary and secondary. The primary design issues identify the main engineering concerns. In this example only one LTSA system component (multimedia) is a primary design priority, but this component has two primary design issues (QoS; protocols and formats). Similarly, the secondary design issues identify the next priority of engineering concerns. Again, in this example a single secondary design priority (delivery) corresponds to more than one secondary design issue.
- **Non-LTSA Focus:** This entry describes design issues but not necessarily in the terms of the LTSA system components. The purpose of this entry is to describe the stakeholders' design issues in their own terms rather than LTSA terminology.
- **Other Issues:** This entry is for other stakeholder issues not described elsewhere in the stakeholder mapping.

8 Operational components and interoperability

This Clause is informative and not normative.

This Clause identifies the main operational components that are common to many learning technology systems, such as codings, APIs, protocols, interchange specifications, processes, stores (databases), information flows, and human interfaces. Not all interoperability components are incorporated into all learning technology systems.

Developer and administrator overview (informative)

In an actual learning technology system, knowing all the codings, APIs, protocols, and other interoperability interfaces is necessary but not sufficient. Several other compatibility features are required, such as connectivity, security, nomadicity, and administration. However, the LTSA provides a framework for analyzing of and planning for integrated, interoperable systems.

Teacher and learner overview (informative)

The operational and interoperable components identified at this level represents the subsystems, codings, APIs (application programming interfaces), protocols, etc., of actual systems. Having common functionality does not guarantee interoperability: cellular telephones and walkie-talkies have similar functionality but do not interoperate. Similarly, having interoperable interfaces does not imply functionality: two systems can use the internet TCP/IP protocols, but they may be disconnected from each other so there is no functionality. Thus, a complete description of functions, codings, APIs, protocols, and other interfaces is necessary.

8.1 Abstraction-implementation boundary

The following diagram shows stakeholder perspectives and priorities.

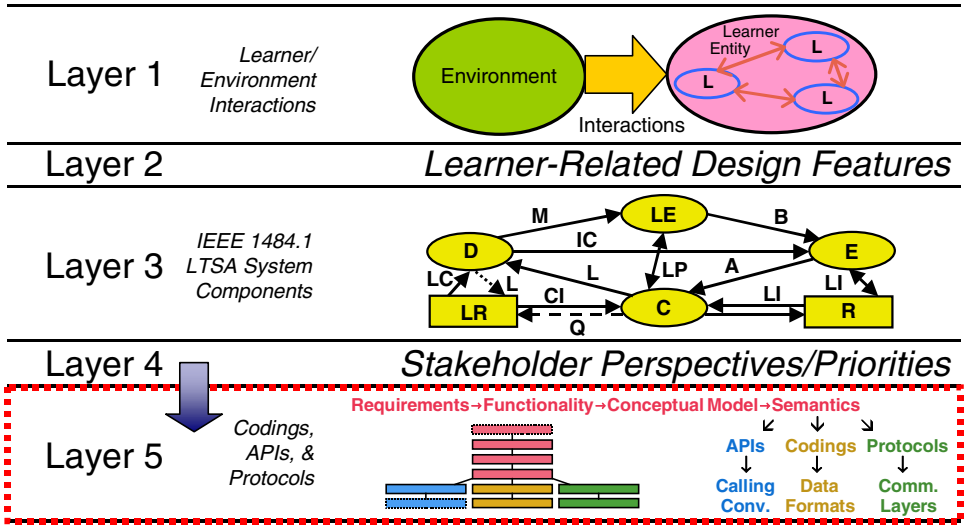


Figure 26. Layer 4 ⇒ Layer 5: Stakeholder perspectives (abstractions) implemented as interoperability components in various domains, e.g., codings, APIs, protocols.

Although there are varying stakeholder perspectives, there are common operational and interoperable components within each of the stakeholders' systems.

8.2 Standards/specifications development process

This Standard was developed using the techniques described in Annex G, Methodology. However, the following description provides much more detail about the standards development process applied towards information technology interoperability standards for codings, APIs, and protocols.

The following steps are not required by the formal standard process, but they are *a best practice for developing high quality standards and specifications with a 5-10 year technical horizon.*

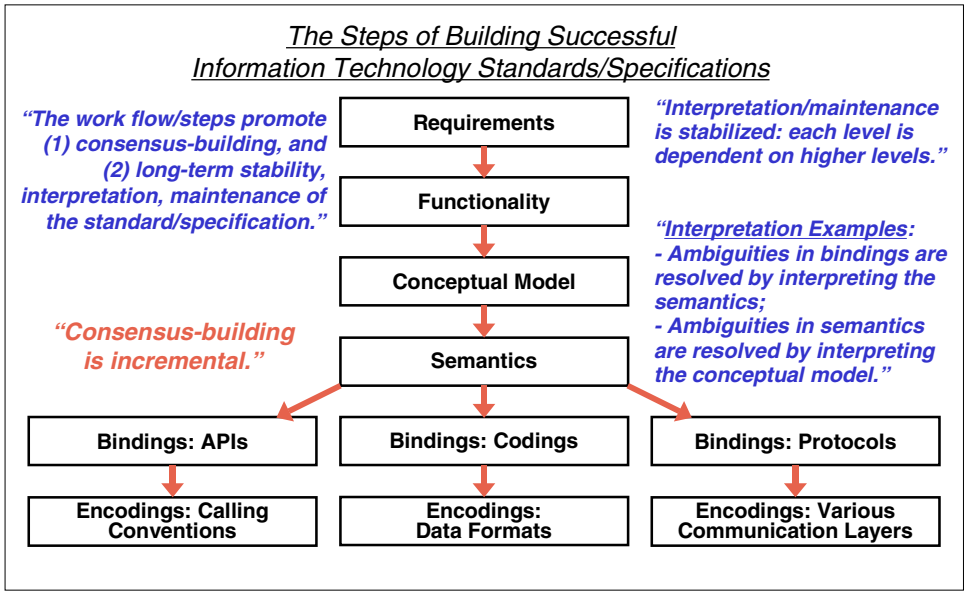


Figure 27. Major steps for developing interoperability standards in information technology for many application areas.

The following are the major steps in the development of IT interoperability standards. These standards are "lower level" when compared to architecture standards.

Note: These steps are not development phases, but areas of development risk and technical description.

- **Requirements:** Allows the standard or specification, upon completion, to be validated by reaffirming the satisfaction of the original requirements. *Result: The standard or specification remains useful according to its original intent. Note: The formal standards process is not requirements-based, i.e., the identification of requirements is a useful "best practice".*
- **Functionality:** Helps delimit and "contain" the scope of the standard, which minimizes "feature creep" — an undesirable hazard of standards development. *Result: The consensus-building process will stay focused on its technical goals.*
- **Conceptual Model:** Describes a virtual implementation that models the theory of operation. For the maintenance phase of the standards lifecycle, the conceptual model may be used to resolve ambiguities in semantics that were unforeseen or overlooked in the formal consensus-building. *Result: The standard or specification can adapt to changes in technology.*
- **Semantics:** Describes the precise meanings of interoperability and are described separately from conceptual model and bindings. Semantics are not tied to or influenced by a particular binding. *Result: Semantics are binding-independent so more (future) bindings and applications are possible, thus, a longer lifetime for the standard or specification, and increased interoperability.*
- **Bindings:** Describes the mappings to particular codings, file formats, APIs, commands, protocols, transaction sets, and so on, and allow them to be separated from the "standard behavior" (semantics). *Result: The standard or specification can have common functionality across many operating environments (languages, operating systems, syntaxes, file types, protocol stacks, service methods, etc.), thus the standard or specification will have wider applicability and adoption.*

- **Encodings:** Describes the bit/byte representation and allows them to be separated from the information structure. *Result: The standard or specification can be transformed into "native" representations that are optimal for individual, specific operating environments, thus the standard or specification will have wider adoption.*

By separating the standards or specification development into several steps, certain higher-risk issues can be addressed earlier (e.g., conceptual model and conformance), while certain lower-risk issues (e.g., API signatures and character sets) can be postponed. *Result: Resources are best utilized and scheduling can be more predictable.*

Note: The remaining diagrams of development steps illustrate how technical sources and the workflow process can be used to create IT interoperability standards for learning technology. Many standards committees use workflow techniques like these.

8.3 Harmonizing technical activities

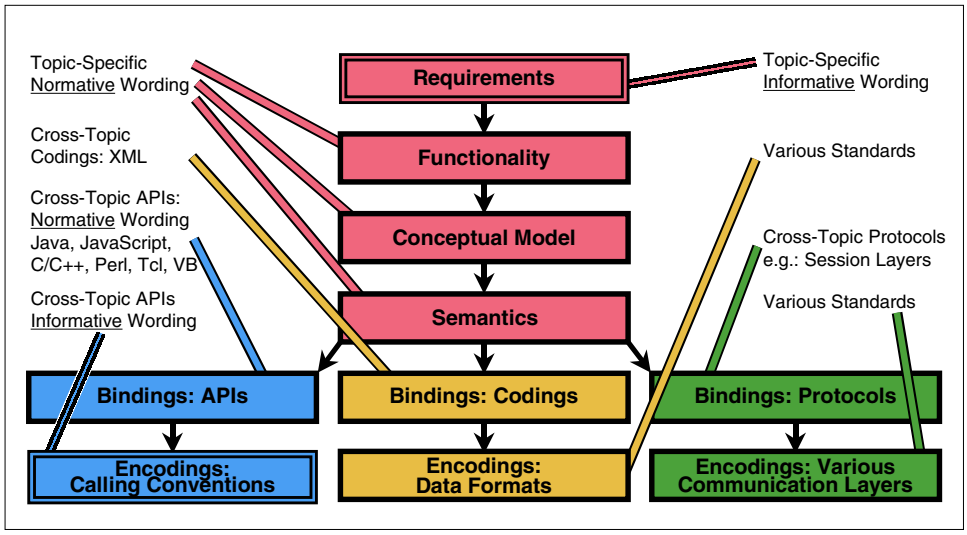


Figure 28. Generic harmonization of IT interoperability standards, as represented in the workflow. Note: The coloring used in this diagram is unrelated to other notations defined and used in this Standard.

Generically, harmonization of IT interoperability standards may be possible (1) by discovering common technology areas, (2) by separating bindings from the functionality of the standard, and (3) by collaborating and liaising with related technical activities.

9 Annex A: Bibliography (informative)

This Annex is informative and not normative.

The following is a collection of supporting documents and web resources that were used in the development of this Standard.

- http://web.ansi.org/iisp, American National Standards Institute, Information Infrastructure Standards Panel (ANSI IISP)
- http://www.eoe.org/, Apple Computer's Educational Object Economy (EOE)
- http://advlearn.lrdc.pitt.edu/its-arch/p1484/ARM.html, "Architecture Abstraction Hierarchy Reference Model", by Frank Belz, Dan Suthers, Tom Wheeler
- http://ariadne.unil.ch, ARIADNE Project of European Union
- http://aicc.org, Aviation Industry Computer-Based Training (CBT) Committee (AICC)
- http://www.omg.org/corbamed, Common Object Request Broker Architecture of Object Management Group (OMG), Medical Infomatics (CORBAMED)
- http://www.adlnet.org, DoD Advanced Distributed Learning (ADL)
- http://imsproject.org, Educause's Instructional Management Systems Project (IMS)
- http://edutool.com, Edutool specifications and documents on learning technology infrastructure
- http://ltsc.ieee.org, Institute of Electrical and Electronics Engineers, Learning Technology Standards Committee (IEEE LTSC)
- http://jtc1sc36.org, International Organization for Standardization - International Electrotechnical Committee, Joint Technical Committee 1, Subcommittee 36 — Information Technology for Learning, Education, and Training (ISO-IEC JTC1 SC36)
- http://www.itscj.ipsj.or.jp/caw, International Organization for Standardization - International Electrotechnical Committee, Joint Technical Committee 1 — Cultural Adaptability Workshop (CAW)
- http://ssdo.org/jtc1/gii-roadmap, International Organization for Standardization - International Electrotechnical Committee, Joint Technical Committee 1 — Global Information Infrastructure Standards Roadmap (ISO-IEC JTC1 GII)
- http://ssdo.org, International Organization for Standardization - International Electrotechnical Committee, Joint Technical Committee 1 — Standards Operations Roundtable (SORT):

10 Annex B: Learner-related influences (informative)

This Annex is informative and not normative.

10.1 Learner-environment interactions

The top refinement layer of the LTSA is a *very* generalized architecture refinement layer called "Learner-Environment Interactions" (see below).

Important: There is often much confusion about this level of abstraction. The aim at this level is to view the system from *an information technology perspective* (e.g., in terms of the flow of information). Many readers misread this refinement layer as a description of some theory of learning. *This description is not a diagram of any theory of learning.* The purpose of describing learning technology at this level of abstraction is to relate it to software engineering methodology in order to create lower levels of abstraction.

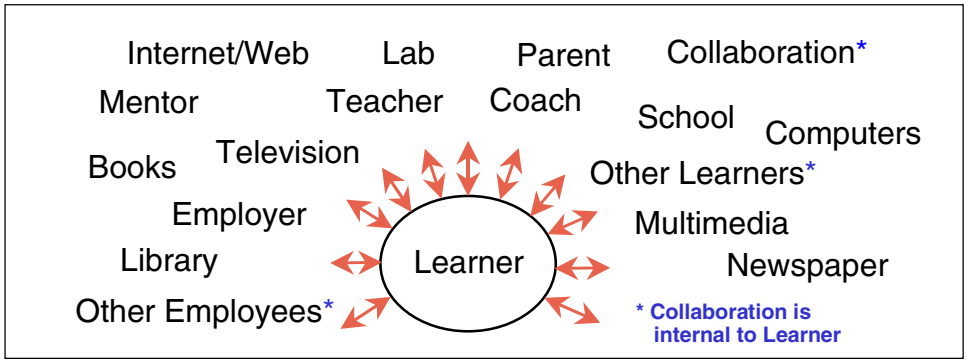


Figure 29. The learner's view of the learning environment.

10.2 System description

The learner-environment interactions diagram (see below) only represents the Learner Entity and their Environment from a systems engineering perspective of information technology, i.e., this diagram doesn't portray current research on theories of learning. The reason for using this diagramming technique is to simplify certain engineering aspects of technology design: the focus is on the overall view of information flow and the system is diagramed as a one-way arrow (flow) of interactions from the environment to the learner entity. The implementations of concepts (lower level abstractions or systems themselves) may focus on pedagogical issues or other technical issues.

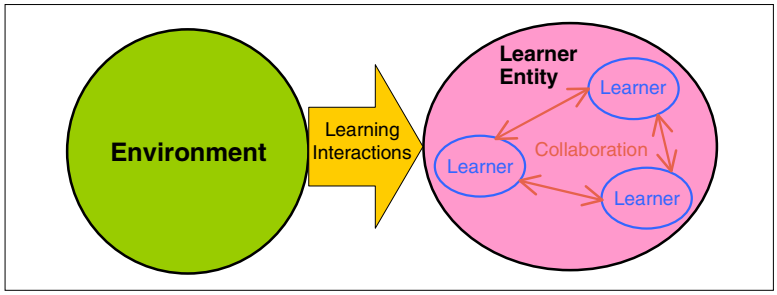


Figure 30. A system view of learner-environment interactions. This diagram is equivalent to the previous diagram. Note: Collaboration among the learners is internal to the collective learner entity.

10.3 Learner entity

The Learner Entity (process) represents an abstract learner, e.g., an individual human learner, several learners working collaboratively, or members of a team operating in different roles.

Collaboration among learners is *internal* to the collective Learner Entity. An analogy is a distributed database system: several individual databases "collaborate" to give the appearance of a single database. The LTSA notion of learner collaboration, i.e., internal to the Learner Entity rather than as a separate component, is an important simplifying feature of the LTSA.

10.4 Environment

The Environment (process) represents the environment with which the Learner Entity interacts. The Learning Interactions (flow) may be correlated to learning experiences.

10.5 Learner-related design issues

The design of the lower layers of the architecture are affected by the needs of learners and, in particular, the nature of human (in contrast to machine) learning. These details of the learners' effects on system design are outside the scope of this Standard.

11 Annex C: Illustrations of stakeholder mappings (informative)

This Annex is informative and not normative.

This Annex contains a sample of many different stakeholders' perspectives in learning technology systems. This Annex shows how each perspective is represented, is relevant, and is included within the framework of the Learning Technology Systems Architecture (LTSA). Clause 7, Stakeholder Perspectives and Priorities, describes the LTSA layer 4 and this stakeholder diagramming and notation.

Each stakeholder has an important, legitimate perspective. However, each stakeholder has a different perception of learning technology systems.

First, the generic stakeholders are presented in order of complexity (isolated, overlapping, parallel). Second, related industries are correlated to learning technology. Third, the work of various standards and specification development organizations is presented as stakeholder perspectives themselves.

Note 1: Each perspective is represented by a diagram employing a subset of the LTSA components, each with its own emphasis and de-emphasis on particular components. The *emphasis and de-emphasis* (primary, secondary, and other design issues) *reflect the technology issues, not the pedagogy*.

Note 2: The primary design priorities are shown in red and **bold**. The secondary priorities are shown in blue and double lines. LTSA components that are not primary or secondary, or are not applicable are shown without distinction, with normal weight, or in olive.

11.1 Building consensus among stakeholders

Building consensus among such a large and diverse group of stakeholders is difficult. This observation is based upon much standards and specification development experience within the Global Information Infrastructure (GII) that spans diverse industries, including many information and communication technologies (see "http://web.ansi.org" and "http://ssdo.org/jtc1/gii-roadmap"). One discovery has been the limitations of technical analysis itself, as applied to cross-industry solutions: *in some cases technical consensus may be impossible due to conflicting business priorities — in other words, technical consensus is impossible because of non-technical issues*. For example, one stakeholder group may require high security while another stakeholder group requires high usability — typically, security and usability are conflicting requirements. In a single organization or enterprise it may be possible to strike a compromise — all organizational structures provide a common manager, executive, or board to resolve disputes. However, the scope of the LTSA spans many organizations and many stakeholders, so such compromises may be very difficult or even impossible.

The purpose of labeling primary and secondary design issues is to identify the business priorities — typically, different among stakeholders that share the *same* subset of LTSA system components.

Consensus is built around a common component architecture (LTSA layer 3) by identifying the stakeholders' perspectives (LTSA layer 4) and identifying the interoperability and/or interchange protocols that meet the stakeholders' needs (LTSA layer 5).

Note: Only LTSA layer 3 (system components) is normative in this Standard. The descriptions of the remaining four layers are informative.

11.2 Stakeholders ordered by complexity

Stakeholders are divided into five categories: isolated, overlapping, parallel, related industries, and LTSC standard activity.

The "isolated" stakeholders have relatively simple features concerning isolated and neighboring LTSA system components. The "isolated" stakeholders (1) have little overlap with other "isolated" stakeholders, and (2) can make use of isolated "component" standards. Typical examples of "simple" stakeholders are Learner Records, Metadata, and Multimedia.

The "overlapping" stakeholders are concerned with many, most, or all LTSA system components. The "overlapping" stakeholders are complex because they (1) overlap with other stakeholders, and (2) have differing, possibly conflicting, design priorities that can make standards and interoperability difficult. Typical examples of "overlapping" stakeholders are Experimentation, Intelligent Tutoring Tools, and Distance Learning.

The "parallel" stakeholders are concerned with the integration of multiple, active sessions of LTSA system components. The "parallel" stakeholders (1) must synchronize, start, and stop multiple sessions, (2) must integrate, collaborate, and synchronize the feedback, coaching, and user interfacing, and (3) may use recursive design. A typical example of a "parallel" stakeholder is the Student Teacher.

The "related industries" stakeholders concern industries that have substantial overlap with the LTSA, such as entertainment and multimedia, expert systems, and control and feedback systems.

The "LTSC standards activity" stakeholders concern each of the LTSC Working Groups and their technical activity. This description, effectively, is a roadmap to the LTSC WGs and shows relationships among activities.

The differing and conflicting design priorities can make standards setting difficult. An architecture like LTSA can help resolve that conflict. A typical example of conflicting priorities yet similar LTSA components may be seen by comparing the Metadata and Ontologies stakeholder perspectives.

11.3 Few, isolated components

The "isolated" stakeholders are characterized by addressing few and/or isolated LTSA system components. The "isolated" stakeholders, usually, can make good use of "component" standards for interacting with *other* LTSA system components, such as data interchange formats, protocols, application programming interfaces, and object-based components.

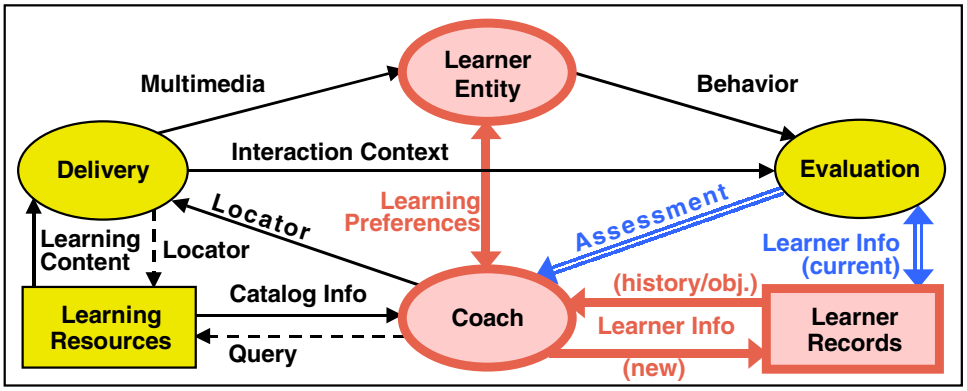
The following are the "isolated" stakeholders identified in the LTSA:

- **Learner-Centered:** The learner, learner-maintained recordkeeping, and learner-motivated coaching.
- **Assessment-Centered:** Educational standards, assessment, and recordkeeping.

- **Records, Certifications:** Recordkeeping and creating, maintaining, and validating certifications.
- **Learner Profiles:** learner information, such as performance, preference, and other important information.
- **Student Administration Systems:** Back office systems and related systems.
- **Task Model, School-To-Work:** Task descriptions to make learners more attractive to employers and school-to-work programs.
- **Institution-Centered:** K-12 and higher education learning environments, recordkeeping, large teaching staff, and large student bodies.
- **Content-Centered:** Designers, developers, and producers of learning content.
- **Learning Content Cataloging, Metadata:** Cataloging, searching, and indexing learning content.
- **Ontologies, Expert Systems:** Knowledge organization, engineering, and coding for retrieval as learning content.
- **Digital Libraries:** Automated and distributed libraries of electronic media.
- **Learning Objects:** Learning content integrated with course structure and sequencing.
- **Content Launch:** Launching content in distance, distributed, and nomadic multi-platform environments.
- **Content Objects:** Small media components structured and aggregated into larger components of learning content.
- **Content Packaging:** Aggregating and bundling learning content and related components.
- **Content Developer:** Developers of learning content and their supporting systems.
- **Digital Audio and Video:** The digital audio and digital video components of multimedia.
- **Multimedia Search and Retrieval:** Auditory, visual, and other sensory information, and physical interactions.
- **Peripheral devices:** Input/output devices attached to learning technology systems and related systems.
- **Collaboration, Asynchronous Learning:** Learners (of the collective learner entity) operating as teams in which learners have similar roles. Learners may access the learning environment and/or collaborate at different times.
- **Multiple Role Learning, Team Learning:** Learners (of the collective learner entity) operating as teams in which learners have different roles.
- **Icon conventions:** User interface conventions for icons in learning technology systems.

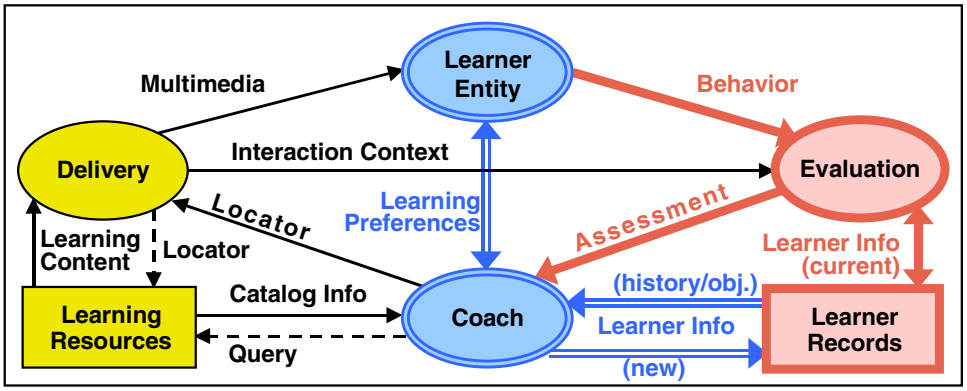
Typical examples of "isolated" stakeholders are Learner Records, Metadata, and Multimedia Search and Retrieval **—** all are characterized by addressing an isolated subset of LTSA system components.

11.3.1 Learner-centered



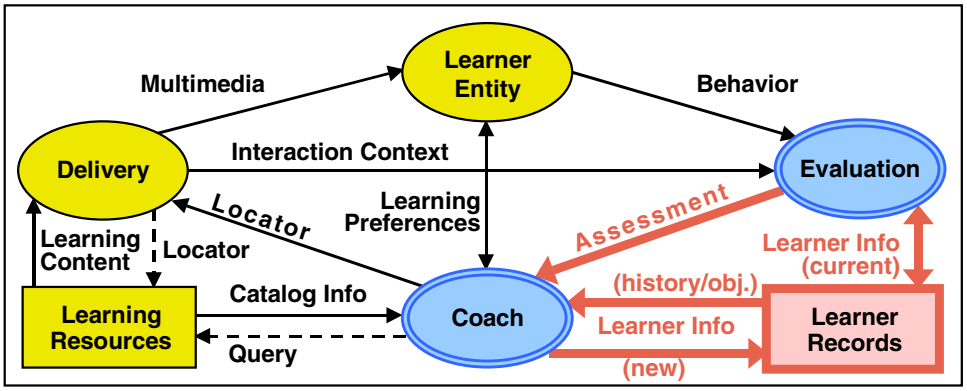
Summary	Learner, learner-maintained recordkeeping, learner-motivated coaching.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocol for communicating the learning preferences. The protocol and format of the learner information. The functionality of the coach as it supports the learner entity's objectives. The ability of the learner entity to maintain his/her learner records.
	Secondary: The protocol and format of assessment information.
Non-LTSA Focus	Primary: The learner keeps his/her records. The learner has influence on his/her learning methods, style, and strategies.
	Secondary: Assessment as feedback on the learner's progress. Learner information as the learner's history.
Other Issues	Nomadic (roaming, sometimes connected) access to learner records by the evaluation process and the coach. Distributed (separated) access to learner records by the evaluation process and the coach.

11.3.2 Assessment-centered



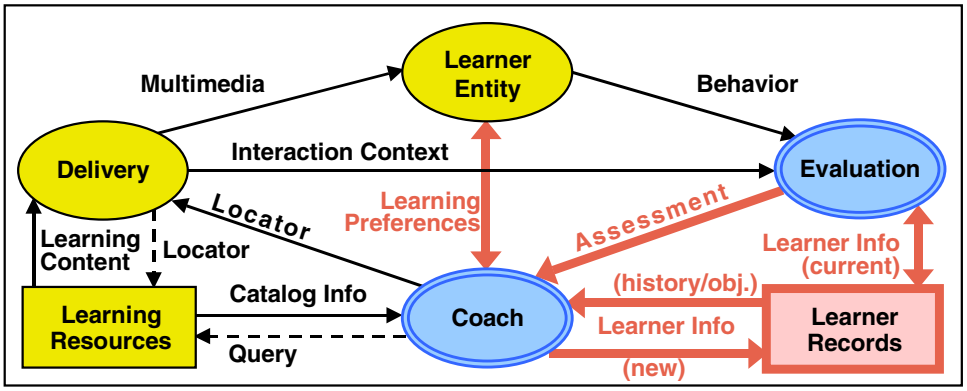
Summary	Educational standards, assessment, and recordkeeping.
LTSA Design Priorities	Primary: The standards, procedures, methods, protocols, and formats of behavior observation. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records.
	Secondary: The interface to the learner entity. The protocol and format of learning preferences. The scope, functionality, and interfaces of the coach.
Non-LTSA Focus	Primary: Evaluation and assessment of learners. Education standards. Maintenance of the learner's records. Reporting systems.
	Secondary: Adapting the system's teaching methods based on the assessment of the student body.
Other Issues	Aggregation of the learner entity's comprehensive learner records and other records.

11.3.3 Records, Certifications



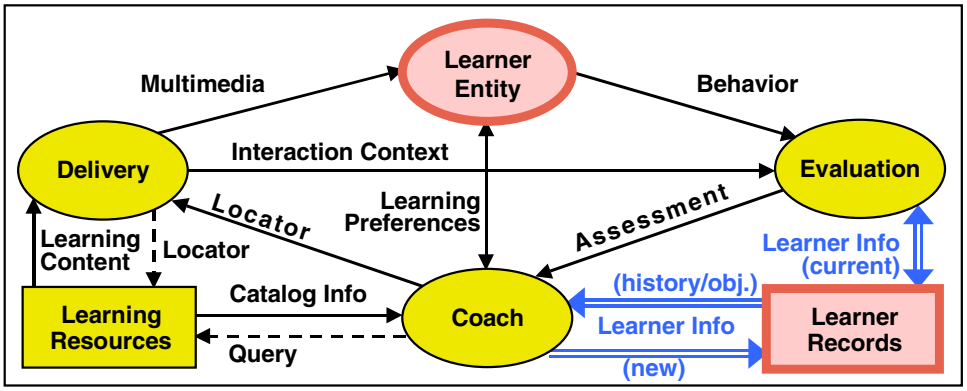
Summary	Recordkeeping and creating, maintaining, and validating certifications.
LTSA Design Priorities	Primary: The protocols and formats of the learner information. The protocols, semantics, and formats of assessment information. The formats, indexing, storage, and retrieval of information in the learner records. Secondary: The scope, functionality, and interfaces of the evaluation process. The scope, functionality, and interfaces of the coach.
Non-LTSA Focus	Primary: Learner entity's records storage and management. Secondary: Common semantics and formats for generation and use of assessment and grades.
Other Issues	Common reporting tools for learner information.

11.3.4 Learner profiles



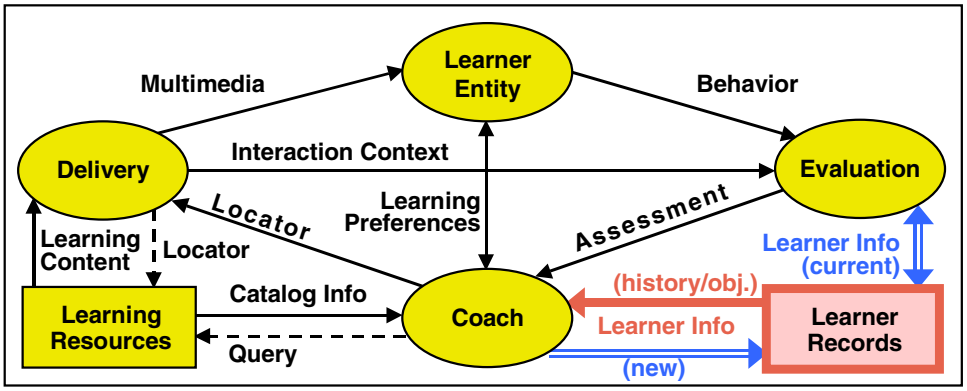
Summary	Learner information, such as performance information, preference information, and other important information.
LTSA Design Priorities	Primary: The protocol for communicating the learning preferences. The protocols and formats of the learner information. The protocols, semantics, and formats of assessment information. The formats, indexing, storage, and retrieval of information in the learner records.
	Secondary: The scope, functionality, and interfaces of the evaluation process. The scope, functionality, and interfaces of the coach.
Non-LTSA Focus	Primary: Learner entity's records storage and retrieval. Supporting learners with special needs, special preferences, and information technology adaptation (e.g., deafness or blindness).
	Secondary: Common semantics and formats for use in related learning technology tools.
Other Issues	Common tools for storing and retrieving learner information.

11.3.5 Student administration systems



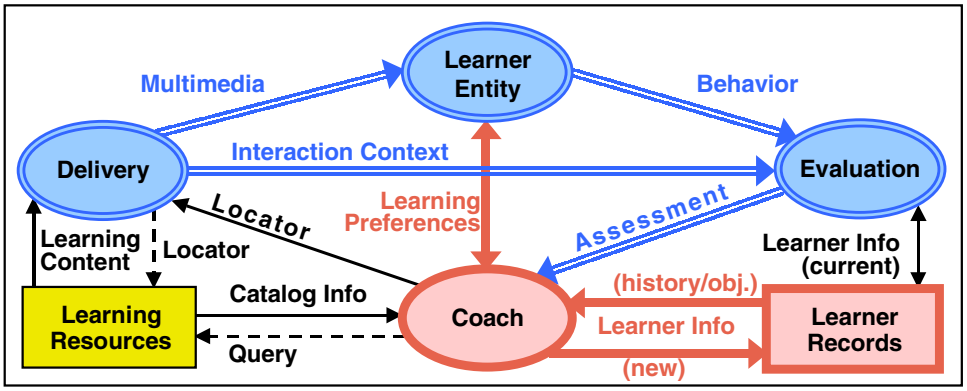
Summary	Back office systems and related systems.
LTSA Design Priorities	Primary: The interface to the learner entity. The formats, indexing, storage, and retrieval of information in the learner records.
	Secondary: The semantics, protocols, and formats of the learner information.
Non-LTSA Focus	Primary: Administering back office systems in learning technology systems and learning institutions. Security and authentication methods and mechanisms.
	Secondary: Communication with related learning technology tools, subsystems, and components.
Other Issues	Collaboration and communication with other back office systems.

11.3.6 Task model, School-to-work



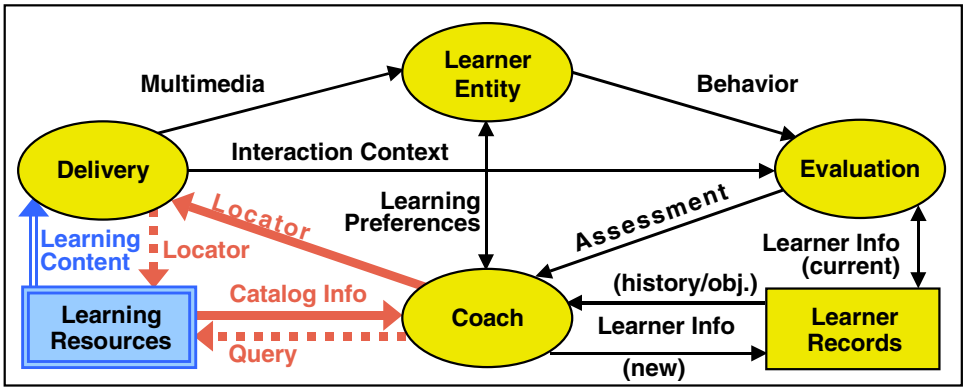
Summary	Task descriptions to make learners more attractive to employers and school-to-work programs.
LTSA Design Priorities	Primary: The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of (historical and future) learner information. Secondary: The protocols and formats of the learner information.
Non-LTSA Focus	Primary: Accessing the learner's history database for qualified learners that meet the needs of some employer, project, or "task". Secondary: Common and interoperable formats for distributed databases.
Other Issues	The semantics of matching learners' skills and capabilities to the needs of employers, projects, and/or "tasks". Supporting school-to-work programs, and for task bidding (job bidding), task buying, and task selling. Security methods to control access to the learner entity's history.

11.3.7 Institution-centered



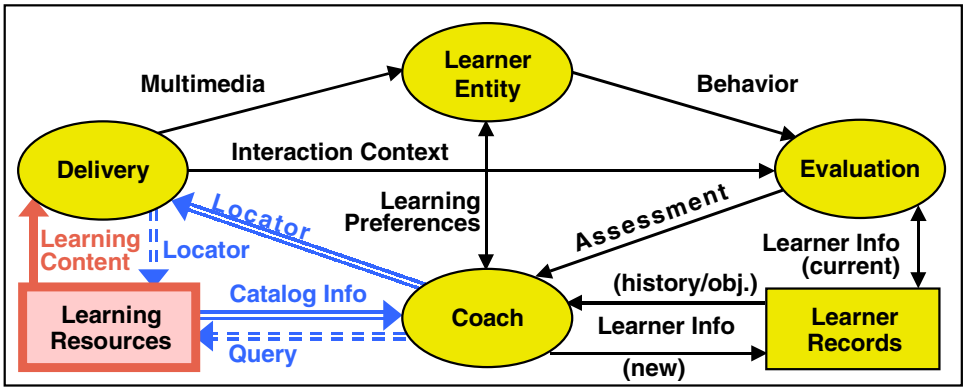
Summary	K-12 and higher education learning environments, recordkeeping, large teaching staff, large student bodies.
LTSA Design Priorities	Primary: The protocol and format of learning preferences. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records. The scope, functionality, and interfaces of the coach. Secondary: The interface to the learner entity. The standards, procedures, methods, protocols, and formats of behavior observation. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: The institution is responsible for recordkeeping and reporting. The institution may have a strong interest in selecting and negotiating learning styles. The institution may have thorough integration of learning styles, control of course delivery and pacing, records format, and common (not necessarily centralized) recordkeeping. Secondary: The institution must may a large student body. The infrastructure to support course delivery, evaluation, assessment, and grading.
Other Issues	Common security systems to maintain confidentiality and integrity.

11.3.8 Learning content cataloging, Metadata



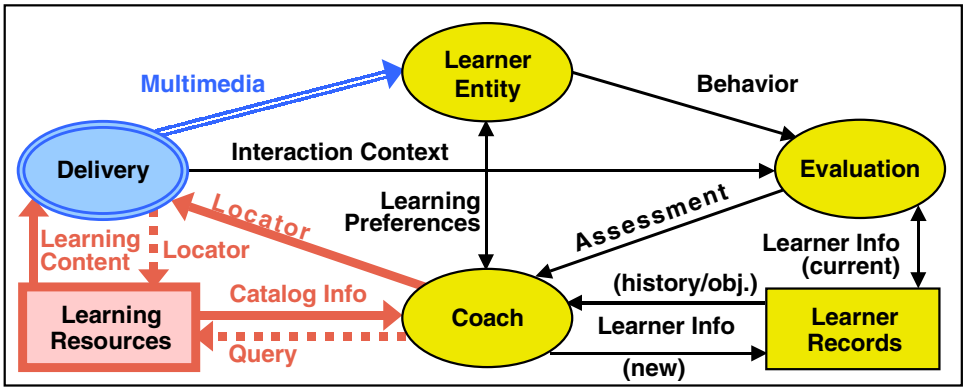
Summary	The cataloging, searching, and indexing of learning content and the attributes associated with learning content..
LTSA Design Priorities	Primary: The protocols and formats of the queries, catalog info, and locators of the learning resources. Secondary: The organization and structure of the learning resources. The protocols and formats of learning content generated from the learning resources.
Non-LTSA Focus	Primary: Searching, locating, and creating coherent lesson plan(s) of appropriate learning materials in a large, distributed library. Secondary: The protocols, semantics, and formats of learning content and learning materials.
Other Issues	A common method of invoking, initiating, or starting learning content, e.g., starting an intelligent tutoring system will probably be different from calling up a web page. Distributed and nomadic learning resources. Naming and taxonomic conventions.

11.3.9 Ontologies, Expert systems



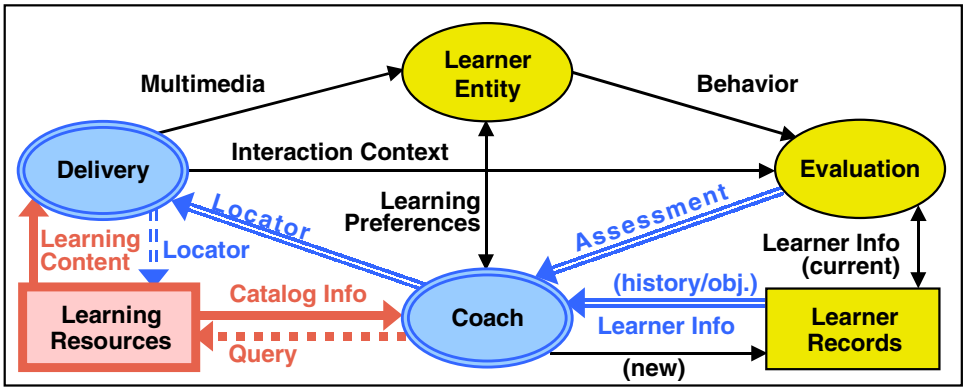
Summary	Knowledge organization, engineering, and coding for retrieval as learning content.
LTSA Design Priorities	Primary: The protocols, semantics, and formats of ontologies, expert systems, and knowledge systems that implement the learning resources. The protocols and formats of learning content generated from the learning resources.
	Secondary: The protocols and formats of the queries, catalog info, and locators of the learning resources.
Non-LTSA Focus	Primary: Ontologies to support learning content derived from knowledge libraries. Knowledge systems and knowledge libraries that represent expert knowledge.
	Secondary: The methods used to search and retrieve appropriate ontologies and knowledge resources.
Other Issues	Coordination and combination of various knowledge systems.

11.3.10 Digital libraries



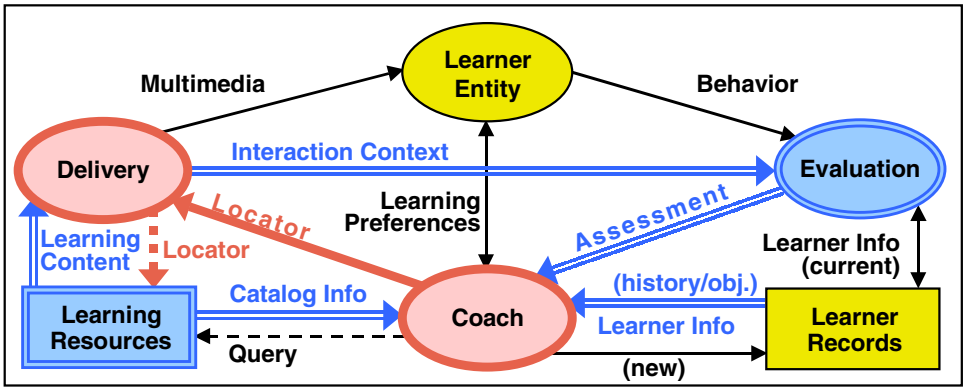
Summary	Automated and distributed libraries of electronic media.
LTSA Design Priorities	Primary: The protocols and formats of the queries, catalog info, and locators of the learning resources, e.g., a digital library of learning content. The organization and structure of units of knowledge and information, as organized by the learning resources. The protocols and formats of learning content generated by or extracted from the learning resources. Secondary: The scope, functionality, and interfaces of the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: A rich, diverse repository of electronic learning materials. Secondary: The infrastructure to support the delivery of learning materials.
Other Issues	Integration with other libraries and cataloging systems.

11.3.11 Learning objects



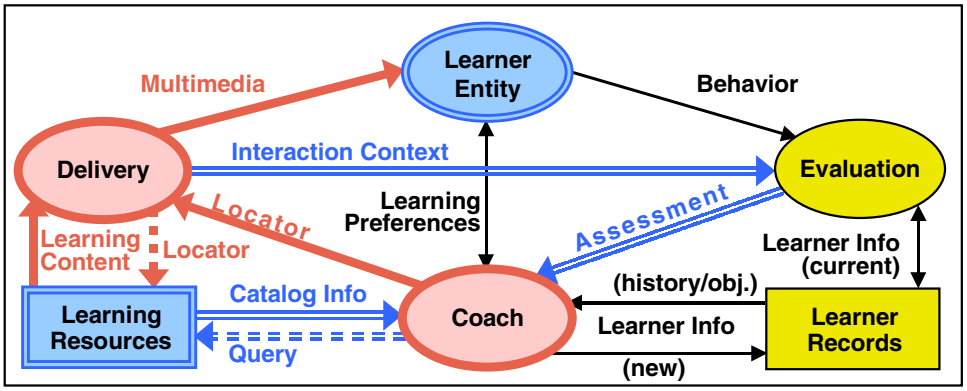
Summary	Learning content with a focus on reusability.
LTSA Design Priorities	Primary: The protocols and formats of the queries and catalog info of the learning resources. The organization and structure of units of knowledge and information, as organized by the learning resources. The protocols and formats of learning content generated from the learning resources. Secondary: The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The scope, functionality, and interfaces of the coach. The protocols and formats of locators. The scope, functionality, and interfaces of the delivery process.
Non-LTSA Focus	Primary: A rich, diverse, and reusable set of learning materials. The methods for searching for appropriate learning materials. The creation, packaging, or repackaging of learning materials with diverse learning resources. Creation of sharable, reusable learning content. Secondary: The infrastructure to support the learner's progress through a large digital library of small units of learning materials. The integration of the learning materials into a content structure and/or sequencing system.
Other Issues	Methods for determining pre-requisites and co-requisites of learning materials.

11.3.12 Content launch



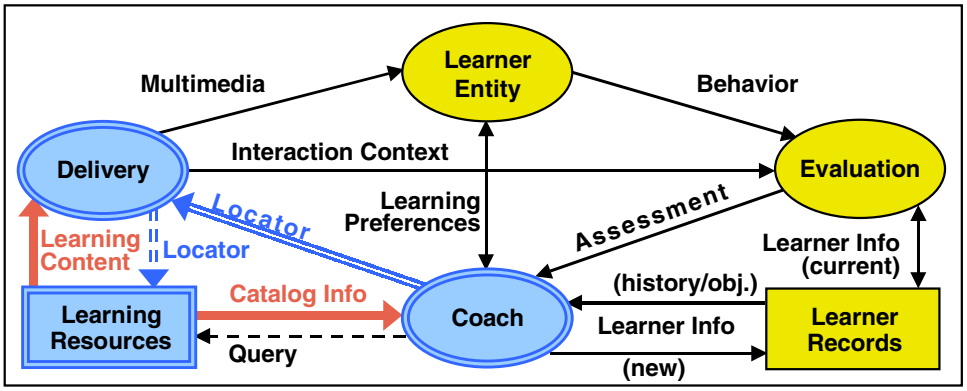
Summary	Launching content in distance, distributed, and nomadic multi-platform environments.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the coach. The protocols and formats of the locators of learning content. The transformation methods used to convert learning content to multimedia. The scope, functionality, user interface, and control inputs and outputs to the delivery process. Secondary: The protocols and formats of the interaction context generated from the learning resources. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The protocols and formats of catalog info. The organization and structure of the learning resources. The protocols and formats of learning content generated from the learning resources.
Non-LTSA Focus	Primary: Invocation or initiation of multimedia delivery. Secondary: Cataloging, searching, and retrieving learning content. Correlation of learning content to multimedia presentations and behavior responses. Assessment of the leaner.
Other Issues	Enrollment verification for students. Intellectual property rights management (IPRM), including copy-protection, security, and usage billing. Close coupling of the delivery process with the evaluation process for responsive, interactive learning experiences.

11.3.13 Content-centered



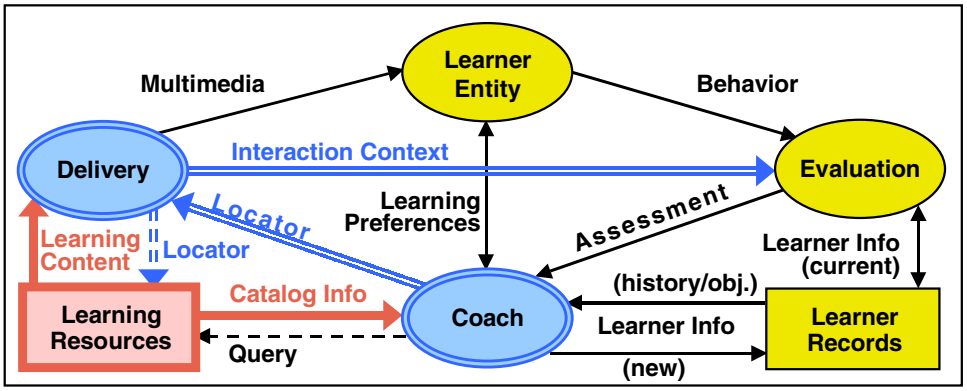
Summary	Designers, developers, and producers of learning content.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the coach. The protocols and formats of the locators of learning content. The protocols and formats of learning content generated from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia. The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The interface to the learner entity. The protocols, semantics, and formats of assessment information. The protocols and formats of queries and catalog info of the learning resources. The protocols and formats of the interaction context generated from the learning resources.
Non-LTSA Focus	Primary: The delivery of diverse, interactive multimedia to the learner. Invocation or initiation of multimedia delivery.
	Secondary: Cataloging, searching, and retrieving learning content. Correlation of learning content to multimedia presentations and behavior responses. Assessment of the learner.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing. Close coupling of the delivery process with the evaluation process for responsive, interactive learning experiences.

11.3.14 Content objects



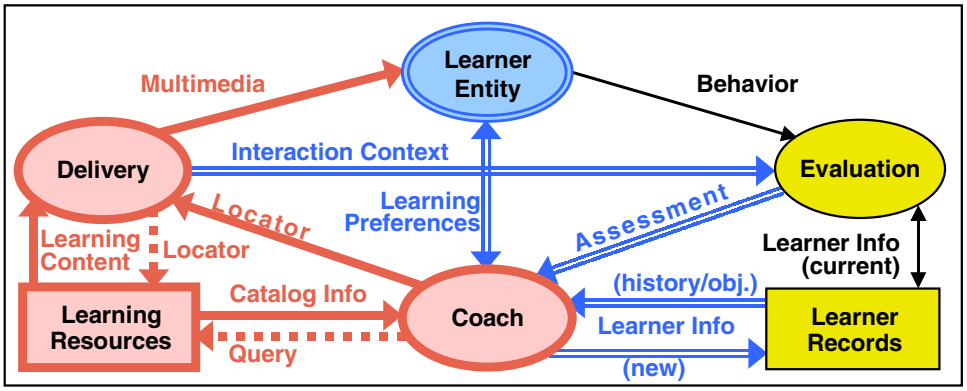
Summary	Small media components structured and aggregated into larger components of learning content.
LTSA Design Priorities	Primary: The protocols and formats of the catalog info associated with the learning resources. The protocols and formats of learning content generated from the learning resources.
	Secondary: The scope, functionality, and interfaces of the coach. The protocols and formats of locators. The organization and structure of units of knowledge and information, as organized by the learning resources. The scope, functionality, and interfaces of the delivery process.
Non-LTSA Focus	Primary: A rich, diverse set of media (content) resources to be used as learning resources. The cataloging methods for appropriate learning materials. Creation of sharable, reusable learning content.
	Secondary: The integration of the learning materials into a course structure and/or sequencing system. The combination, packaging, and use of content objects in a larger structure, such as a lesson or a course.
Other Issues	Methods for repackaging content objects as, say, learning objects.

11.3.15 Content packaging



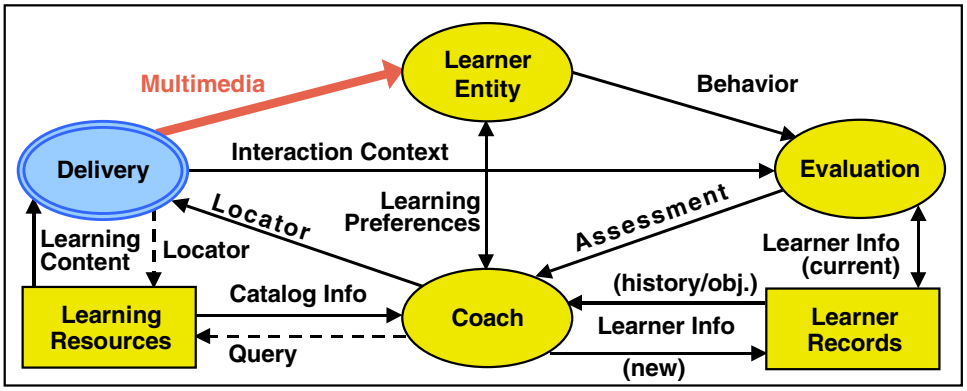
Summary	Aggregating and bundling learning content and related components.
LTSA Design Priorities	Primary: The protocols and formats of the catalog info associated with the learning resources. The organization and structure of units of knowledge and information, as organized by the learning resources. The protocols and formats of learning content generated from the learning resources.
	Secondary: The scope, functionality, and interfaces of the coach. The protocols and formats of the locators of learning content. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia. The protocols and formats of the interaction context generated from the learning resources.
Non-LTSA Focus	Primary: Combination, packaging, and labeling of learning content and related components. Sending and receiving content packages across a variety of platforms and configurations.
	Secondary: Cataloging and launching learning content. Correlation of learning content to multimedia presentations and behavior responses.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing.

11.3.16 Content developer



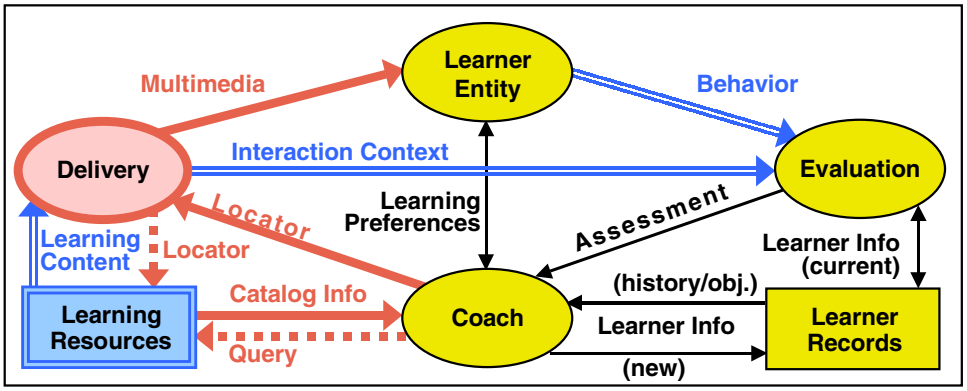
Summary	Developers of learning content and their supporting systems.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the coach. The protocols and formats of the queries, catalog info, and the locators of learning resources. The organization and structure of the learning resources. The protocols and formats of learning content generated from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia. The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The interface to the learner entity. The protocol and format of learning preferences. The protocols, semantics, and formats of assessment information. The protocols, semantics, and formats of learner information. The protocols and formats of the interaction context generated from the learning resources.
Non-LTSA Focus	Primary: The development of multi-platform learning content. The delivery of diverse, interactive multimedia to the learner. Invocation or initiation of multimedia delivery. Tagging and cataloging learning content so that it becomes searchable and widely available for use. Integration of various types of learning content and various curricula.
	Secondary: Progressing the learner through the learning experience. Correlation of learning content to behavior responses. Assessment of the learner.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing. Close coupling of the delivery process with the evaluation process for responsive, interactive learning experiences. Determining pre-requisites and co-requisites for learning content. The content development process is not depicted in LTSA, only the finished product: learning resources and learning content.

11.3.17 Digital audio and video



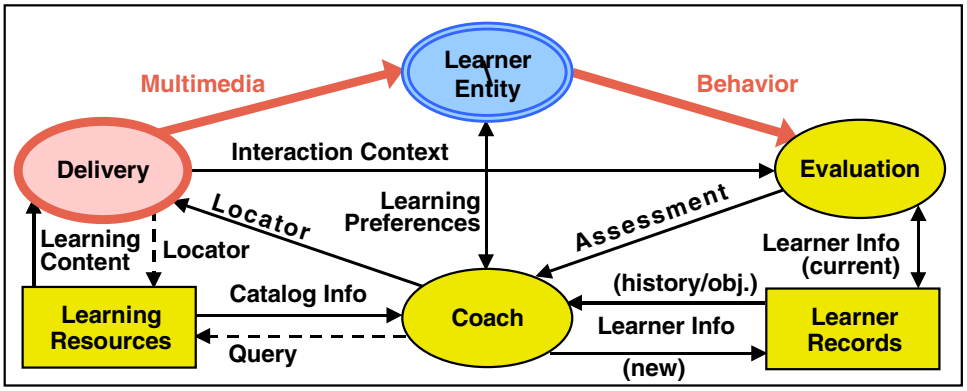
Summary	The digital audio and digital video components of multimedia.
LTSA Design Priorities	Primary: The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia.
Non-LTSA Focus	Primary: Delivery of multimedia across networks of varying capabilities. Multimedia presented on hardware with varying capabilities.
	Secondary: Common presentation of varying multimedia lessons and learning content.
Other Issues	Multi-platform portability.

11.3.18 Multimedia search and retrieval



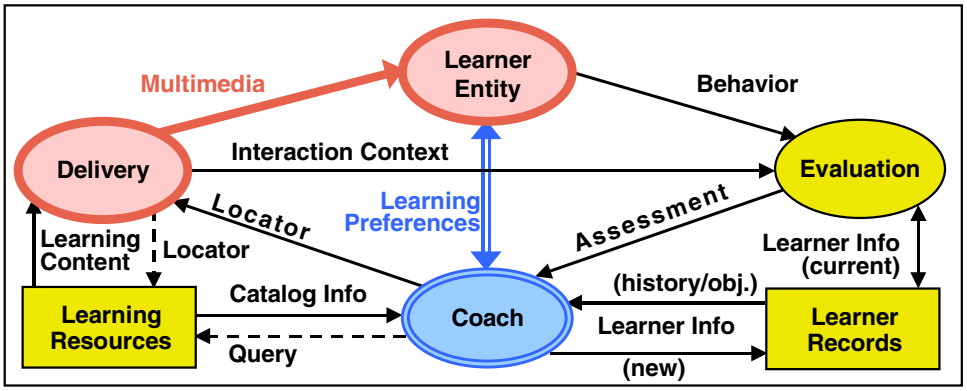
Summary	Auditory, visual, and other sensory information, and physical interactions.
LTSA Design Priorities	Primary: The protocols and formats of the queries and catalog info of the learning resources. The protocols and formats of the locators of learning content. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia. Secondary: The protocols and formats of learning content generated from the learning resources. The transformation methods used to convert learning content to multimedia. The protocols and formats of correlating multimedia by means of the interaction context to the evaluation of behavior. The standards, procedures, methods, protocols, and formats of observable behavior.
Non-LTSA Focus	Primary: Delivery of multimedia across networks of varying capabilities. Multimedia presented on hardware with varying capabilities. Locating and referencing multimedia lessons and learning content. Secondary: Controlling multimedia presentations by the coach as determined by the observable behavior. Seamless access to a large digital library of multimedia. Common presentation of varying multimedia lessons and learning content. Correlation of multimedia to the evaluation of behavior.
Other Issues	Interaction, integration, and close coupling between observing behavior and presenting multimedia.

11.3.19 Peripheral devices



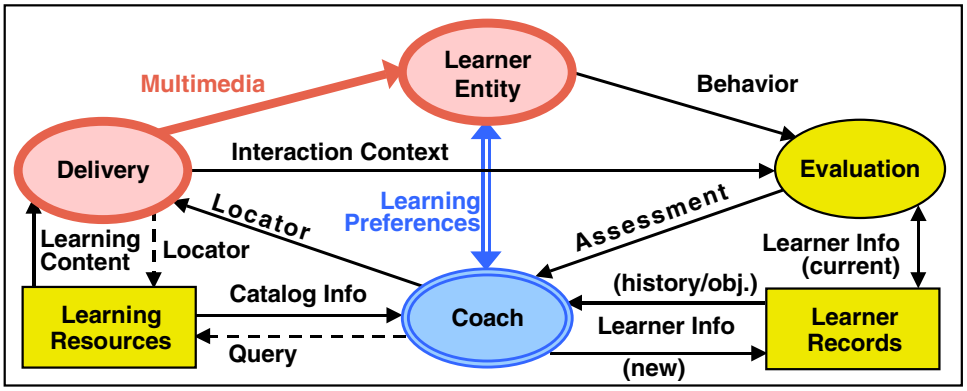
Summary	Input/output devices attached to learning technology systems and related systems.
LTSA Design Priorities	Primary: The standards, procedures, methods, protocols, and formats of observable behavior. The QoS of multimedia connections. The protocols and formats of multimedia. The scope, functionality, user interface, and control inputs and outputs to the delivery process. Secondary: The interface to the learner entity.
Non-LTSA Focus	Primary: Common operating system and environment interfaces for various input/output peripheral devices. Locating and referencing multimedia lessons and learning content. Secondary: Common methods for discovering system and platform capabilities.
Other Issues	Multi-platform packaging for device drivers and related software.

11.3.20 Collaboration, Asynchronous learning



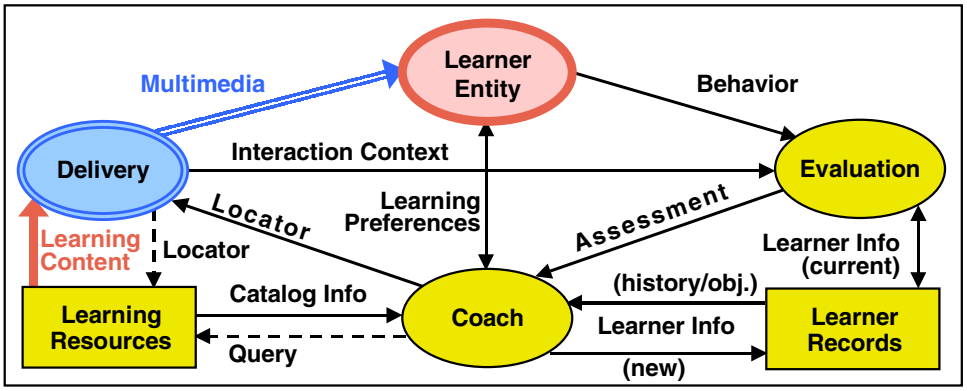
Summary	Learners (of the collective learner entity) operating as teams in which learners have similar roles; learners may access the learning environment and/or collaborate at different times.
LTSA Design Priorities	Primary: The interface to the learner entity. The communication among the learners that represent the collective learner entity. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia. The support tools accessible through the delivery process and sent via multimedia. Secondary: The protocol and format of learning preferences. A coach that supports collaboration activities.
Non-LTSA Focus	Primary: Collaboration among the learners that represent the collective learner entity. Collaboration among learners in different "time zones" or asynchronous access. Secondary: Controlling and/or assisting the learning experience via some coordinator. Using delivery tools to support N-way communication.
Other Issues	Security: Who can participate? Who can control? Who can speak? Who can listen? Logging and/or recording of a collaborative session. Compare to "Multiple role, Team learning" stakeholder.

11.3.21 Multiple role learning, Team learning



Summary	Learners (of the collective learner entity) operating as teams in which learners have different roles.
LTSA Design Priorities	Primary: The interface to the learner entity. The communication among the learners that represent the collective learner entity. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia. The support tools accessible through the delivery process and sent via multimedia. Secondary: The protocol and format of learning preferences. A coach that supports collaboration activities.
Non-LTSA Focus	Primary: Interactions among learners, but learners have different roles (e.g., captain, first officer, flight engineer). Collectively, the learners function as a single, conceptual learner entity. Parallel learning environments are operating simultaneously, so learner plays (at least) two roles: (1) part of the collective learner entity that represents the team, and (2) an individual learner that has his/her own learning environment. Collaboration among learners in different "time zones" or asynchronous access. The communication of multiple delivery processes to the learner entity. Secondary: A coordinator interacting with the coach to lead the team learning.
Other Issues	The communication among the parallel learning environments of the members of the team (learners) as they interact and represent the collective team learner entity. Compare to "Collaboration, Asynchronous Learning" stakeholder.

11.3.22 Icon conventions



Summary	User interface conventions for icons in learning technology systems.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocols and formats of learning content generated from the learning resources. The transformation methods used to convert learning content to multimedia.
	Secondary: The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: Commonality of user interfaces across a variety of platforms, presentation systems, and learning content conventions.
	Secondary: Controlling multimedia presentations by the system coach as determined by the learner's observable behavior. Seamless access to a large digital library of multimedia. Common presentation of varying multimedia lessons and learning content. Correlation of multimedia to the evaluation of behavior.
Other Issues	Interaction, integration, and close coupling among observing behavior and presenting multimedia.

11.4 Many, overlapping, dependent components

The "overlapping" stakeholders are characterized by addressing many and/or overlapping LTSA system components. The "overlapping" stakeholders, usually, have differing and conflicting design priorities that can make standards development and interoperability difficult or impossible. The "overlapping" stakeholders, typically, have difficulty defining internal boundaries (may reduce the applicability of component standards) and have differing system design priorities.

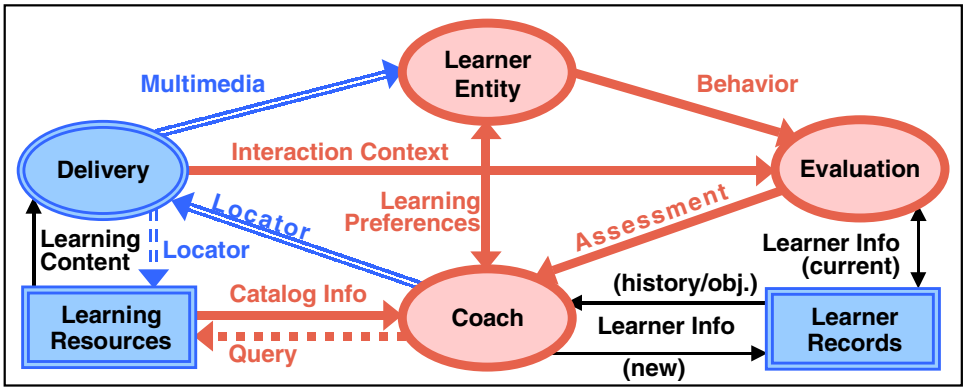
Significant, enterprise-wide, object-oriented solutions become impractical or difficult to integrate for "overlapping" stakeholders because object "inheritance" is closely tied to consensus of technical design priorities (⇒ there are differing design priorities), which is tied to common, shared business and political issues (⇒ no political, business, or technical consensus).

The following are the "overlapping" stakeholders identified in the LTSA:

- **Mentoring, Coaching:** Human and automated coaching of the learner.
- **Electronic Performance Support Systems:** On-site, on-demand learning support systems.
- **Interactive Environment:** Interactive tools and content.
- **Simulation:** Creating virtual worlds for training, experimentation, and instruction.
- **Learning Tool-To-Tool Communication:** Agents and tools communicating with each other in the learner's environment.
- **Sequencing, Pre-Requisites, Co-Requisites:** Course structure and sequencing of modules and learning experiences.
- **Curriculum-Centered:** Curriculum-driven content development.
- **Content Management Systems, Entertainment Systems:** Generic content management, entertainment, and content delivery systems.
- **Learning Management Systems:** Management systems for monitoring, motivating, and affecting the learner's progress.
- **Experimentation, Discovery:** Learning experiences based on learner-directed experiments and discovery.
- **Intelligent Tutoring Tools:** Specialized tools for specific subject areas that tutor and coach learners.
- **Distance Learning, Distributed Learning, Nomadic Learning:** Learning experiences and environments distributed over space (distance learning, sometimes connectedness) and time (asynchronous learning).

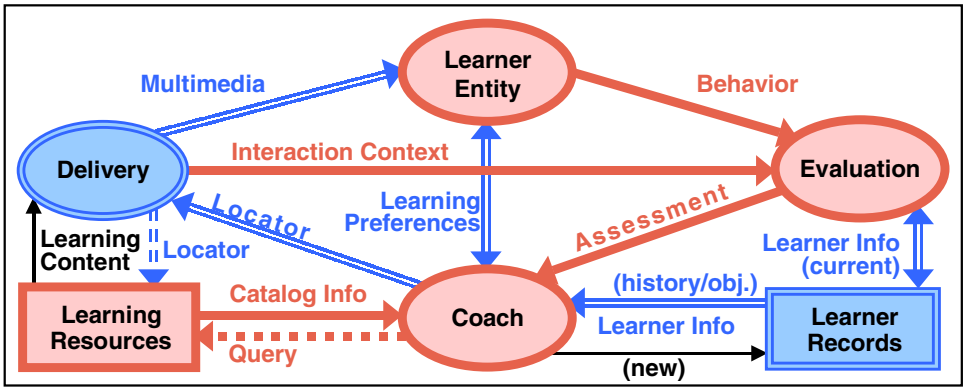
Typical examples of "overlapping" stakeholders are Experimentation, Intelligent, Tutoring Tools, and Distance-Distributed-Nomadic Learning — all address most LTSA system components and have much overlap with other stakeholders.

11.4.1 Mentoring, Coaching



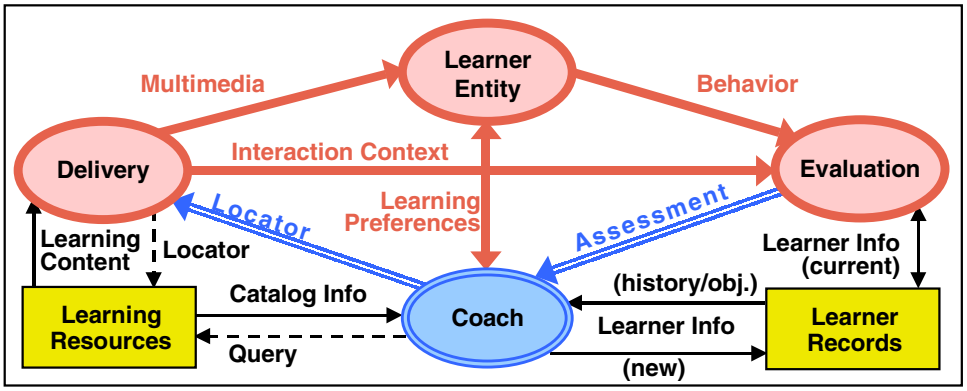
Summary	Human and automated coaching of the learner.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocol and format of learning preferences. The standards, procedures, methods, protocols, and formats of observable behavior. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries and catalog info of the learning resources. Secondary: The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of locators. The organization and structure of units of knowledge and information, as organized by the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: The learner and the mentor jointly evaluate the learner entity. The learner and the mentor collaborate on the learner entity's learning and its direction. Secondary: The tools to support mentoring and experimentation. The delivery mechanisms to support mentoring, collaboration, and experimentation.
Other Issues	The learning resources must be rich enough to support the needs of the learner and mentor.

11.4.2 Electronic performance support systems



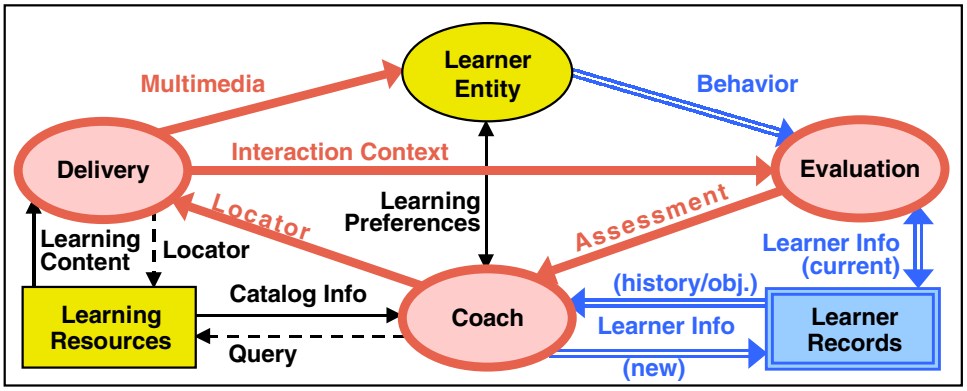
Summary	On-site, on-demand learning support systems.
LTSA Design Priorities	Primary: The interface to the learner entity. The standards, procedures, methods, protocols, and formats of observable behavior. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries and catalog info of the learning resources. The design and organization of the learning resources. Secondary: The protocol and format of learning preferences. The protocols and formats of learner information. The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of the locators of available knowledge, information, tools, and learning environments. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: The learning technology system provides on-demand support to the learner during the performance of his/her work. Secondary: The multimedia to support the user interface.
Other Issues	The human interface that supports the learners while they perform their work.

11.4.3 Interactive environment



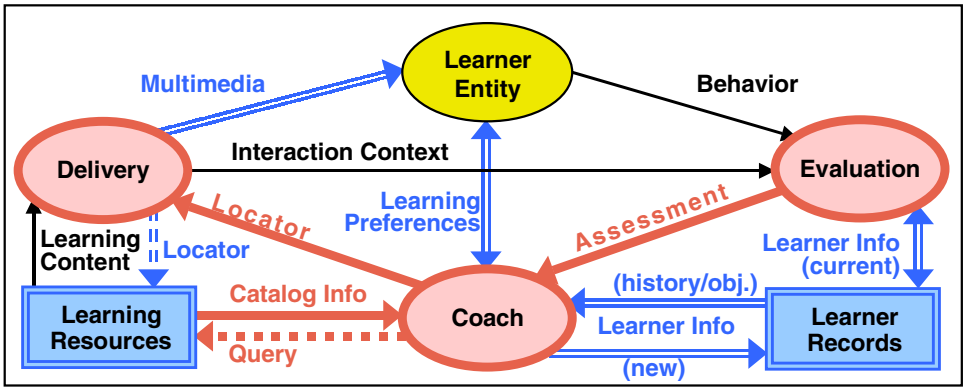
Summary	Interactive tools and content.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocol and format of learning preferences. The standards, procedures, methods, protocols, and formats of observable behavior. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the coach. The protocols and formats of the locators.
Non-LTSA Focus	Primary: The learner has influence over styles and methods of learning. The interaction between the learner and the human-machine interface is very responsive (e.g., response times of less than 3 seconds). Strong coupling among evaluation, system control, and multimedia delivery.
	Secondary: Responsive (quick) evaluation and assessment of the learner. Responsive (quick) direction of learning experience.
Other Issues	Interaction, integration, and close coupling among behavior observations, evaluation process, coach, delivery process, and multimedia.

11.4.4 Simulation



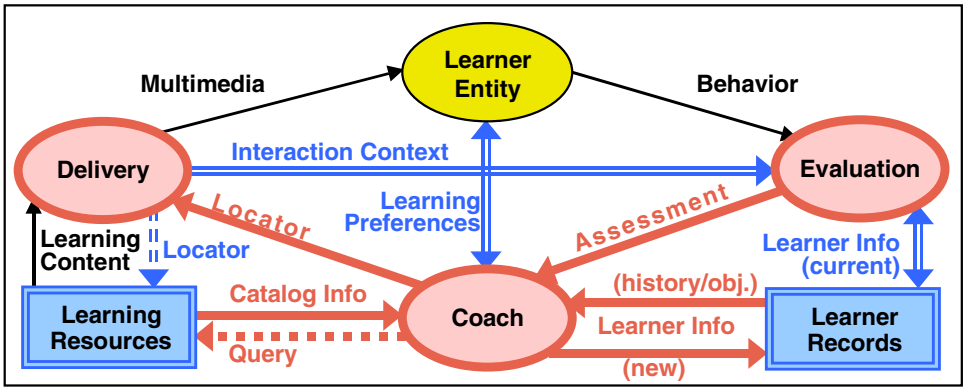
Summary	Creating virtual worlds for training, experimentation, and instruction.
LTSA Design Priorities	Primary: The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the coach. The protocols and formats of the locators of learning content from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia. Secondary: The standards, procedures, methods, protocols, and formats of behavior observation. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records.
Non-LTSA Focus	Primary: Responsive, dynamic, and realistic evaluation of learners actions and their effects on the simulation environment. Hardware limitations to simulation. Simulation fidelity vs. system cost. Limitations to the simulation environment, e.g., what happens when the learner "crashes" a flight simulator?. Secondary: Integrating the learner's actions, usually in fine granularity, within the simulation environment. Recording and analyzing learner behavior for real-time and post-session analysis.
Other Issues	Interaction, integration, and close coupling among behavior observation, evaluation process, coach, delivery process, and multimedia.

11.4.5 Learning tool-to-tool communication



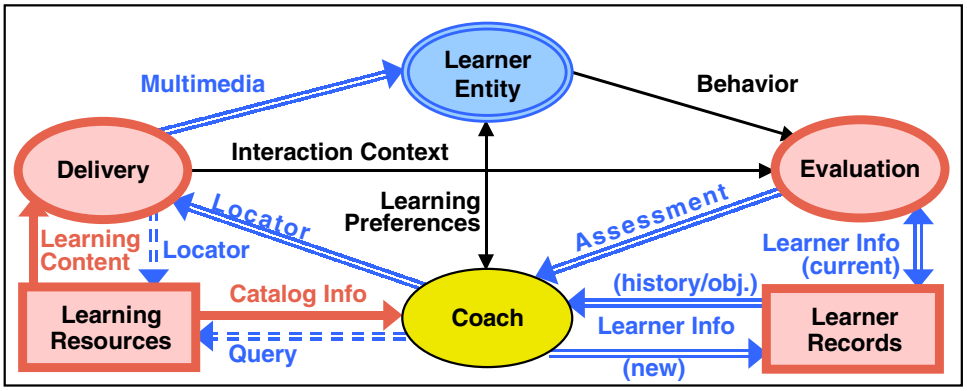
Summary	Agents and tools communicating with each other in the learner's environment.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries and catalog info of the learning resources, e.g., a digital library of tools, laboratories, tutors, and other learning materials. The protocols and formats of the locators of learning content from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. Secondary: The protocol and format of learning preferences. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records. The organization and structure of units of knowledge and information, as organized by the learning resources. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: A strong coupling of evaluation, assessment, system coach, query and planning of learning content, and delivery. Common communication among the various tools and learning materials. Secondary: Access to decision-support information: the learner's history and the available learning re-sources. The learner's access and control over the tools to support various learning styles. Network performance to support varying degrees of multimedia presentation.
Other Issues	The communication among the tools, laboratories, tutors, and other learning materials of the learning resources.

11.4.6 Sequencing, Pre-requisites, Co-requisites



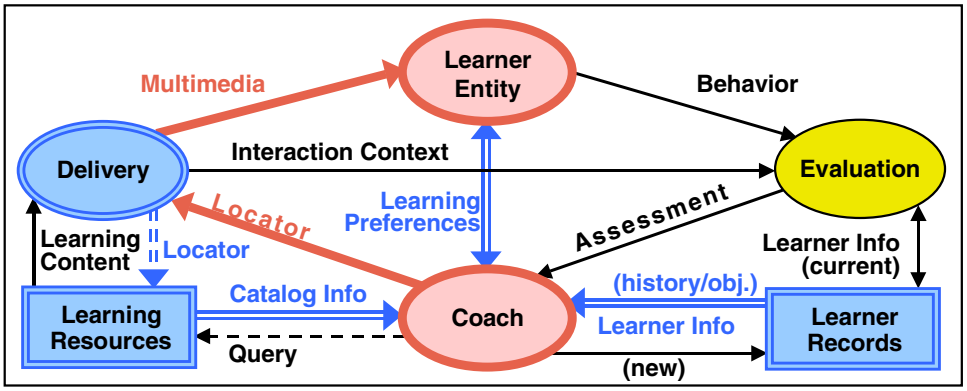
Summary	Course structure and sequencing of modules and learning experiences.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The scope, functionality, and interfaces of the coach. The certification and motion protocols the coach uses to advance the learner through learning materials. The protocols and formats of the queries and catalog info of the learning resources. The protocols and formats of the locators of learning content from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process.
	Secondary: The protocol and format of learning preferences. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The formats, indexing, storage, and retrieval of information in the learner records. The structure, design, and organization of the learning resources.
Non-LTSA Focus	Primary: Common methods for controlling the sequencing and flow of learning materials.
	Secondary: Access to the learner records database to support appropriate content sequencing. Access to the learning styles and negotiation with the learner — supports appropriate content sequencing.
Other Issues	Determining the granularity of the learning material that is sequenced.

11.4.7 Curriculum-centered



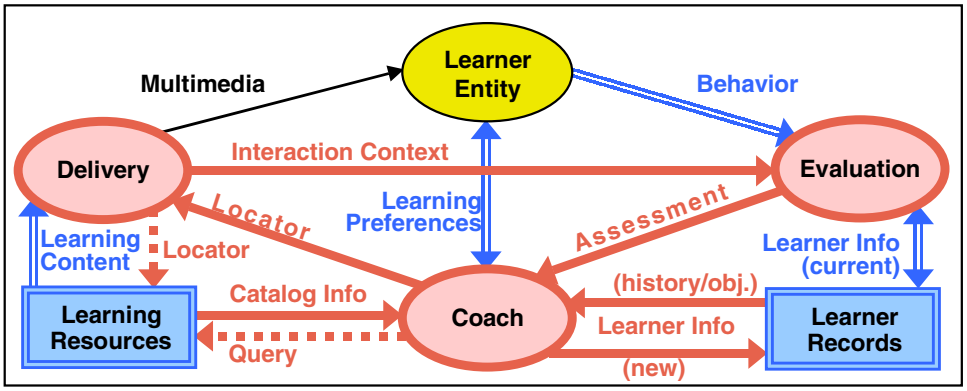
Summary	Curriculum-driven content development.
LTSA Design Priorities	Primary: The scope, functionality, and interfaces of the evaluation process. The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of the catalog info associated with the learning resources. The protocols and formats of learning content generated from the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The transformation methods used to convert learning content to multimedia. Secondary: The interface to the learner entity. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The protocols and formats of the queries of the learning resources. The protocols and formats of the locators of learning content. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: Curriculum development and learning content based on curricula. Secondary: Cataloging, searching, and retrieving learning content. Correlation of learning content to multimedia presentations and behavior responses. Assessment of the learner.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing. Close coupling of the delivery process with the evaluation process for responsive, interactive learning experiences. The tools that support the development process. The content and curricula development processes are not depicted in LTSA, only the finished product: learning content and learning resources.

11.4.8 Content management systems, Entertainment systems



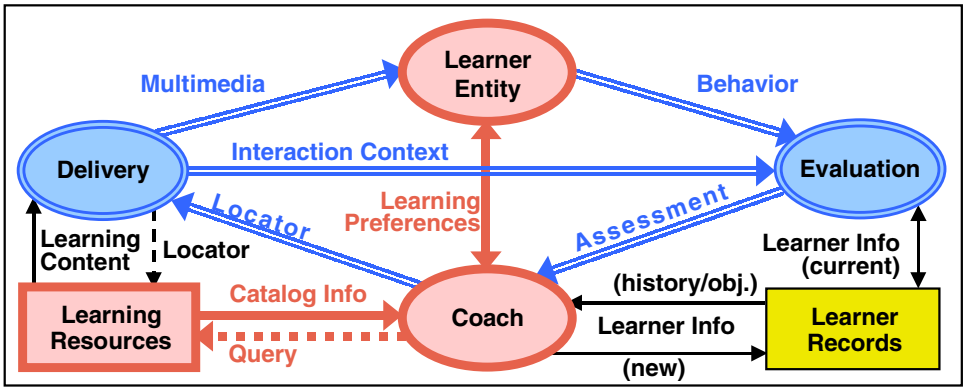
Summary	Generic content management, entertainment, and content delivery systems.
LTSA Design Priorities	Primary: The interface to the learner entity. The scope, functionality, and interfaces of the coach. The protocols and formats of the locators of available laboratories. The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The protocol and format of learning preferences. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of the catalog info of the learning resources. The structure, design, and organization of the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process.
Non-LTSA Focus	Primary: Providing a variety of on-demand content to users.
	Secondary: Catalog selection capabilities.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing.

11.4.9 Learning management systems



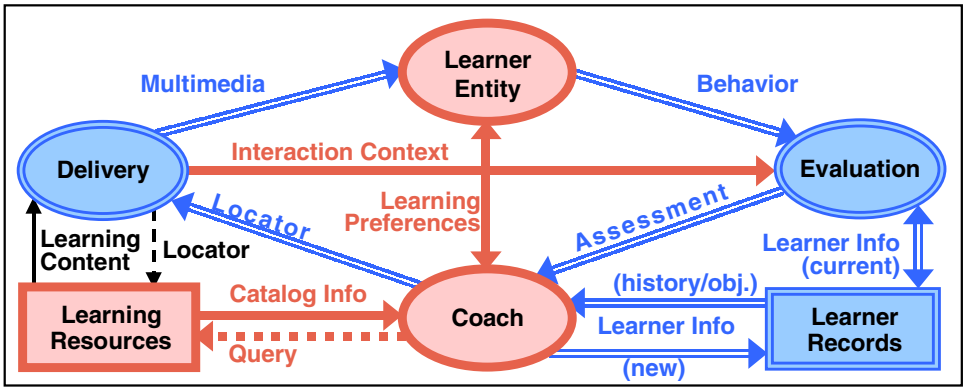
Summary	Management systems for monitoring, motivating, and affecting the learner's progress.
LTSA Design Priorities	Primary: The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries and catalog info of the learning resources. The protocols and formats of the locators of learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process.
	Secondary: The protocol and format of learning preferences. The standards, procedures, methods, protocols, and formats of the learner's observable behavior. The formats, indexing, storage, and retrieval of information in the learner records. The structure, design, and organization of the learning resources.
Non-LTSA Focus	Primary: Progressing learners through units of learning content (e.g., lessons and courses). To launch learning content and to provide the necessary tracking and recordkeeping.
	Secondary: Management system for learning content from a variety of sources.
Other Issues	Intellectual property rights management (IPRM), including copy-protection, security, and usage billing.

11.4.10 Experimentation, Discovery



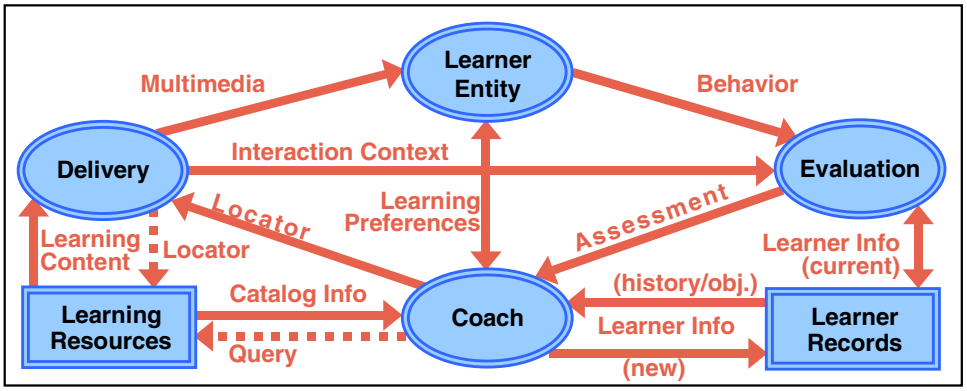
Summary	Learning experiences based on learner-directed experiments and discovery.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocol and format of the learning preferences. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries and catalog info of the learning resources, e.g., a digital library of experimentation and discovery laboratories. The structure, design, and organization of the learning resources.
	Secondary: The standards, procedures, methods, protocols, and formats of the learner's observable behavior. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the locators of available laboratories. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: Permitting an environment that allows the learner to progress by experimentation and discovery. Supporting "what if" exploration by allowing access to wide range of experimentation and discovery environments (laboratories).
	Secondary: The learner does self-evaluation and assessment.
Other Issues	Collaboration and mentoring features to support experimentation and discovery. The learning resources must be rich enough to support experimentation and discovery.

11.4.11 Intelligent tutoring tools



Summary	Specialized tools for specific subject areas that tutor and coach learners.
LTSA Design Priorities	Primary: The interface to the learner entity. The protocol and format of learning preferences. The scope, functionality, and interfaces of the coach. The protocols and formats of the queries of the learning resources, e.g., a digital library of tutoring tools. The structure, design, and organization of the learning resources. The protocol and format of the interaction context to correlate the multimedia to the behavior for the evaluation process. Secondary: The standards, procedures, methods, protocols, and formats of behavior observation. The scope, functionality, and interfaces of the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The formats, indexing, storage, and retrieval of information in the learner records. The protocols and formats of the locators of tutoring tools. The scope, functionality, user interface, and control inputs and outputs to the delivery process. The QoS of multimedia connections. The protocols and formats of multimedia.
Non-LTSA Focus	Primary: The learner chooses the direction, discovery, and experimentation. Tutoring tools are specialized for specific subject areas. Learners learn by using tools. Secondary: Tutoring tools perform specialized behavior observation, evaluation, assessment, and delivery.
Other Issues	The protocols, semantics, and formats of communication among tutoring tools. The tutoring support must support a rich set of learning resources.

11.4.12 Distance learning, Distributed learning, Nomadic learning



Summary	Learning experiences and environments distributed over space and (e.g., distance learning, sometimes connectedness, asynchronous learning).
LTSA Design Priorities	Primary: The protocol and format of learning preferences. The standards, procedures, methods, protocols, and formats of behavior observation. The protocol and format of the interaction context to correlate the multimedia with the behavior for the evaluation process. The protocols, semantics, and formats of assessment information. The protocols and formats of the learner information. The protocol and formats of queries and catalog info of the learning resources. The protocols and formats of the locators of learning materials. The QoS of multimedia connections. The protocols and formats of multimedia.
	Secondary: The interface to the learner entity. The scope, functionality, and interfaces of the evaluation process. The scope, functionality, and interfaces of the coach. The formats, indexing, storage, and retrieval of information in the learner records. The structure, design, and organization of the learning resources. The scope, functionality, user interface, and control inputs and outputs to the delivery process.
Non-LTSA Focus	Primary: Distance (low bandwidth, long latency), distributed (separated), and nomadic (sometimes disconnected) communication of learning preferences, behavior, assessment, and learner information, query, catalog info, locator, multimedia, and interaction context.
	Secondary: Distance, distributed, and nomadic operation of the evaluation process, learner records, coach, learning resources, and delivery process.
Other Issues	Security, learner identification, and collaboration within a distance, distributed, and nomadic environment. Note: This implementation perspective is of interest because its primary design concerns are the LTSA flows, while its secondary design concerns are the LTSA processes and stores.

11.5 Multiple, parallel, and/or recursive components

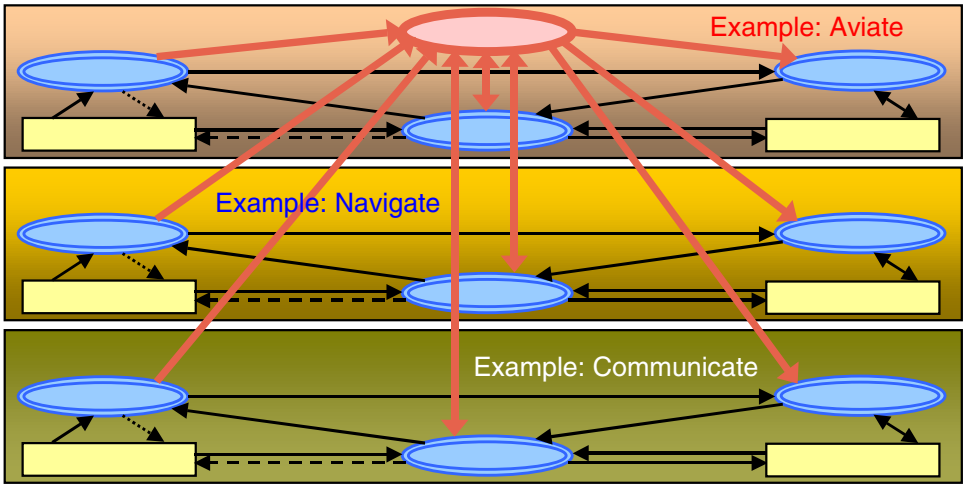
The "parallel" stakeholders are characterized by the integration of multiple, active sessions of LTSA system components. The "parallel" stakeholders must synchronize, start, and stop multiple sessions. The "parallel" stakeholders must integrate, collaborate, and synchronize the feedback, coaching, and user interfacing. The LTSA components may be recursive by having one learner entity play the role of a "non-learner" (e.g., coach) for another learner entity.

The following are the "parallel" stakeholders identified in LTSA:

- **Parallel Sessions for the Same Learner:** Multiple, simultaneous learning experiences.
- **Student Teacher:** Student teacher in his/her role as both "teacher" (one learning experience) and "learner" (another learning experience).
- **Multi-Tiered Process Improvement:** Multiple, simultaneous learning experiences for individuals that support the learning experiences of a common learner(s). Multiple learning experiences, evaluations, assessments, coaching, and delivery processes operating among the student, teacher, principal, and school board as the student, teacher, and principal improve their performance via substantially different learning technology environments.

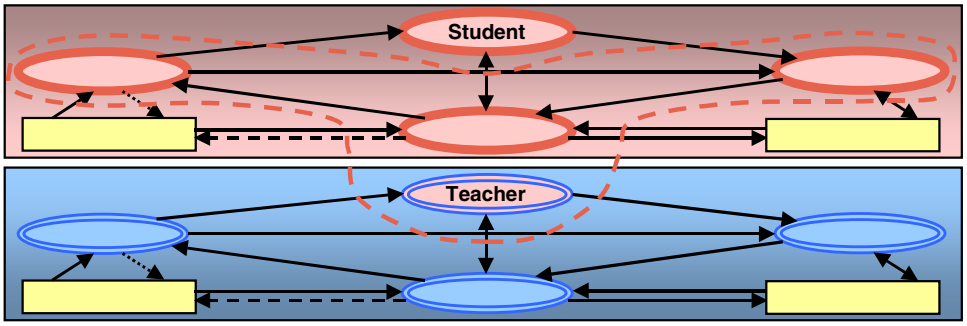
A typical example of a "parallel" stakeholder is the Student Teacher — dual, simultaneous learning experiences as the student teacher is both the "teacher" (coach) and the "learner". A good example of LTSA recursion is Multi-Tiered Process Improvement: the school board serves as coach to the principal as learner entity; the principal serves as coach to the teacher as learner entity; and the teacher serves as coach to the student as learner entity.

11.5.1 Parallel sessions for the same learner



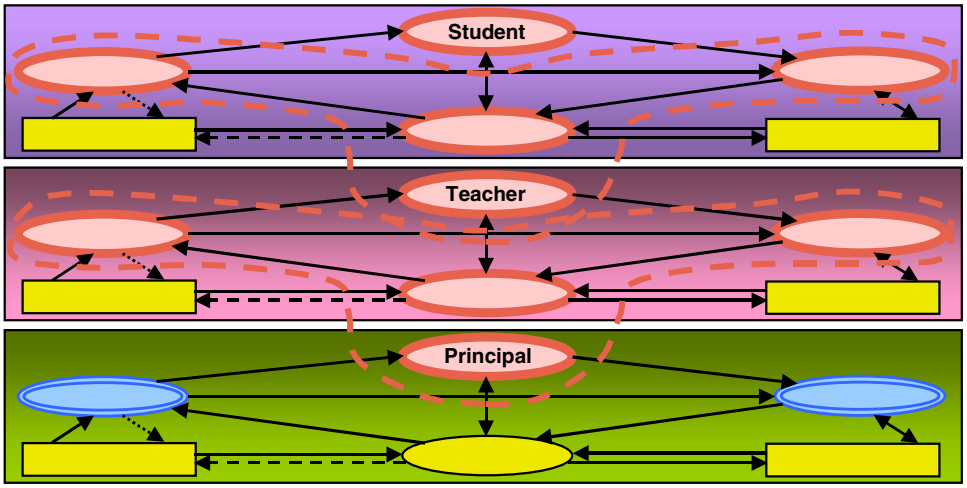
Summary	Multiple, simultaneous learning experiences.
LTSA Design Priorities	Primary: The integration of multiple behavior, interaction context, learner preferences, and multimedia components. The learner entity user interface to support parallel sessions in learning technology systems.
	Secondary: The integration of multiple evaluation, coach, and delivery processes.
Non-LTSA Focus	Primary: Multiple, simultaneous learning experiences. Multiple flows of behavior, learning preferences, and multimedia.
	Secondary: Multiple evaluation, system coach, and delivery processes.
Other Issues	Correlation of learner entity performance among the parallel LTSA sessions. Multiple, simultaneous learning experiences with multiple flows of behavior, learner preferences, and multimedia. Multiple sets of LTSA components support the parallel sessions in learning technology systems.

11.5.2 Student teachers



Summary	The student teacher in his/her role as both "teacher" (one learning experience) and "learner" (another learning experience).
LTSA Design Priorities	Primary: Synchronizing the behavior, interaction context, evaluation, learner information, learner records, assessment information, coach, locator, delivery, and multimedia of both learning experiences. Secondary: Integration of evaluation, assessment, learner records, and coach as one learning experience (e.g., learner information) affects the other learning experience (e.g., evaluation of the teacher's teaching abilities).
Non-LTSA Focus	Primary: The student teacher as a "teacher". Secondary: The student teacher as a "learner".
Other Issues	The correlation of aggregate performance to the teacher's teaching abilities.

11.5.3 Multi-tiered process improvement



Summary	Multiple, simultaneous learning experiences for individuals in different roles that support the learning experiences of a common learner(s).
LTSA Design Priorities	Primary: Synchronizing the behavior, interaction context, evaluation, learner information, learner records, assessment information, coach, locator, delivery, and multimedia of the multiple learning experiences. Secondary: Integration of evaluation, assessment, learner records, and coach as one learning experience (e.g., aggregate learner information) affects another learning experience (e.g., evaluation of teacher's teaching abilities or principal's administrative abilities).
Non-LTSA Focus	Primary: Multiple learning experiences, evaluations, assessments, coaching, and delivery processes operating among the student, teacher, principal, and school board. The teacher coaches and evaluates the student, the principal coaches and evaluates the teacher, and the school board coaches and evaluates the principal. Secondary: Student, teacher, and principal improve their performance via substantially different performance objectives and learning technology environments.
Other Issues	The number, type, and scope of "process improvement" (learning experience) feedback loops and coaching situations. The methods of aggregating the performance of one set of learners (e.g., learners' performance) and correlating them to the performance of another set of learners (e.g., teachers' performance).

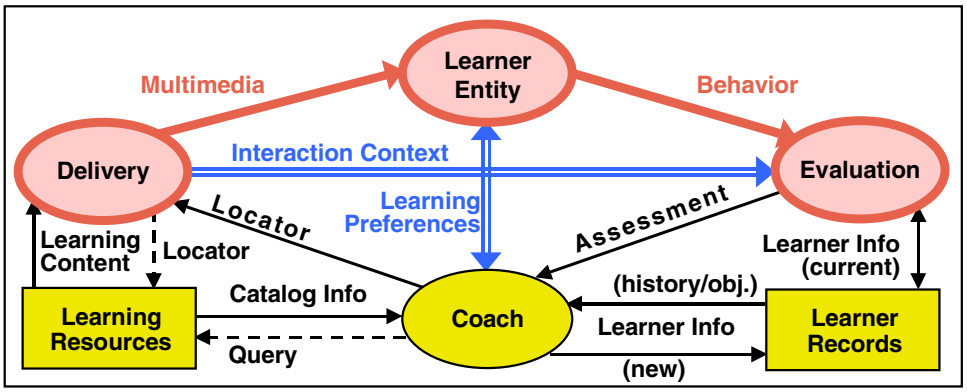
11.6 Related industries

Several industries have significant areas of technology overlap. The following are the "related industries" stakeholders identified in the LTSA:

- **Human Factors, User Interfaces:** The LTSA "top half".
- **Data Collection and Analysis:** The LTSA "bottom right quadrant".
- **IT Decision-Support Systems:** The LTSA "bottom half".
- **Expert Systems, Intelligent Systems:** The LTSA "bottom left quadrant".
- **Entertainment and Multimedia:** The LTSA "left half".
- **Control and Feedback Systems:** The LTSA "center".

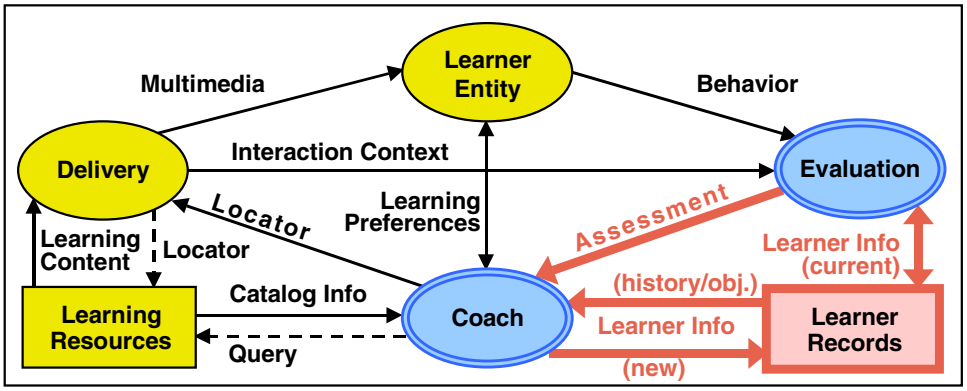
Note: Because other industries have significantly different business and architectural perspectives than the LTSA, only overlapping components and technologies are diagrammed — no mappings between industries are attempted, as per comments about the GII in the beginning of this Annex.

11.6.1 Human Factors, User Interfaces



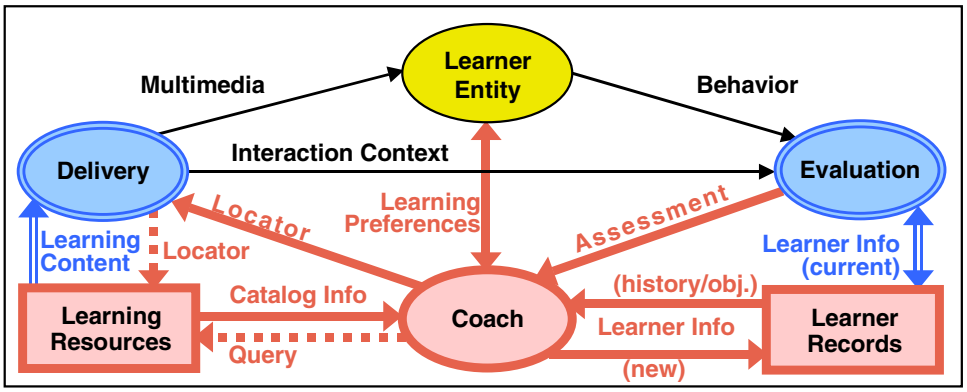
Summary	The human factors and user interfaces technology areas overlap with the "top half" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus.

11.6.2 Data collection and analysis



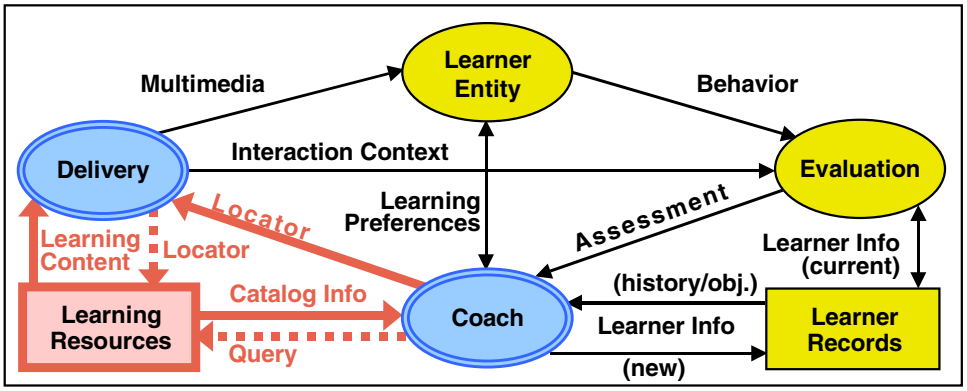
Summary	The data collection and analysis technology areas overlap with the "bottom right quadrant" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus.

11.6.3 IT decision-support applications



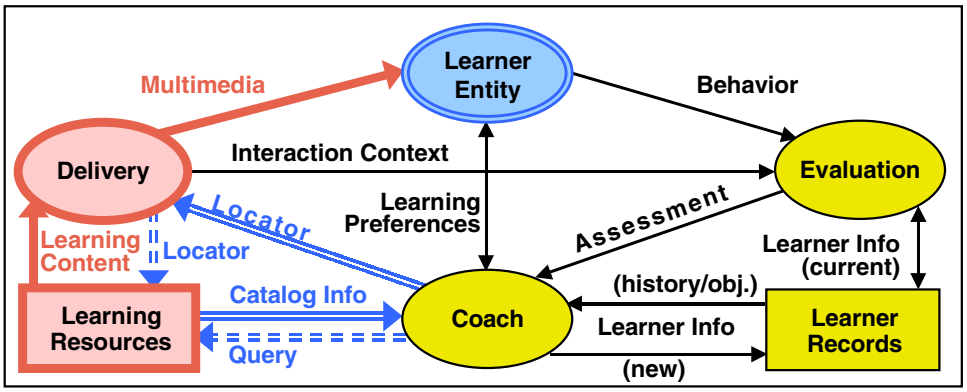
Summary	The IT decision-support technology areas overlap with the "bottom half" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus.

11.6.4 Expert systems, Intelligent systems



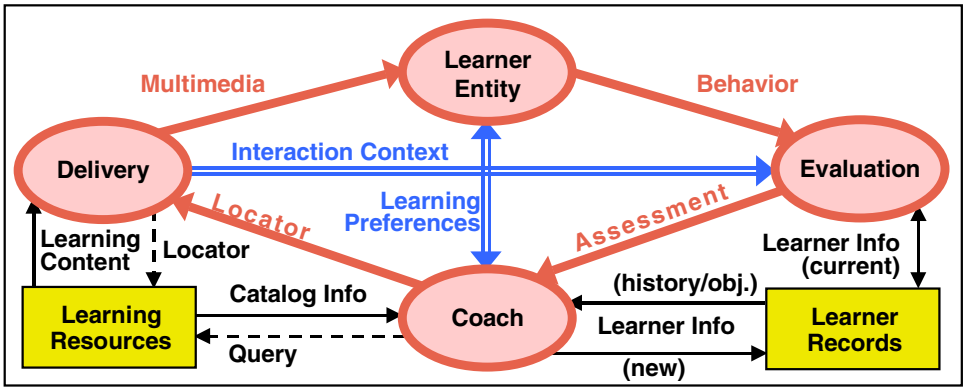
Summary	The expert systems and intelligent systems technology areas overlap with the "bottom left quadrant" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus.

11.6.5 Entertainment, Multimedia systems



Summary	The entertainment and multimedia technology areas overlap with the "left half" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus.

11.6.6 Control and feedback systems



Summary	The control and feedback systems technology areas overlap with the "center" of the LTSA system components. Several other LTSA stakeholders have a similar or overlapping technology focus. Note: The notion of learning as a "control and feedback system" provides a somewhat simplistic approach towards learning technology and the learning process. However, learning technology systems do share many technologies in common with control and feedback systems technology.

11.7 Summary

All of the stakeholders' perspectives can be represented via the LTSA system components with varying emphasis, de-emphasis, and prioritization of specific LTSA system components.

The LTSA system component organization represents a common abstraction of the various stakeholder perspectives. Thus, the LTSA system component organization is a framework for architecting and engineering learning technology systems. Using the automobile analogy again, the LTSA system components are similar to the components of an automobile architecture: depending upon the perspective of a particular stakeholder, certain components are emphasized (various automobile perspectives: power, maneuverability, maintainability, style, fuel efficiency, payload, configurability, ruggedness, number of passengers, etc.), but most cars have similar components.

12 Annex D: Pro Forma Implementation Conformance Statement (normative)

Implementations that claim conformance to this Standard shall complete the following form.

IEEE 1484.1 LTSA Pro Forma Implementation Conformance Statement (ICS)	
Claimant's Name, Organization, Address	
Date of Conformance Claim	
Name of Implementation	
Version of Implementation	
Digital Signature/Checksum	
URL for Related Resources	
Available LTSA System Components and Their Mapping to the Implementation	
Learner Entity (LE)	
Behavior data flow from Learner Entity to Assessment (B)	
Evaluation process (E)	
Assessment data flow from Evaluation to Coach (A)	
Learner information data flow between Evaluation and Learner Records (L1)	
Learner Records data store (R)	
Learner information data flow from Learner Records to Coach (L2)	
Learner information data flow from Coach to Learner Records (L3)	
Learning Preferences between Learner Entity and Coach (L4)	
Coach process (C)	
Query control flow from Coach to Learning Resources (Q)	
Learning Resources data store (LR)	
Catalog Info data flow from Learning Resources to Coach (QI)	
Locator data flow from Coach to Delivery (L5)	
Delivery process (D)	
Locator control flow from Delivery to Learning Resources (L6)	
Learning Content data flow from Learning Resources to Delivery (IC)	
Interaction Context data flow from Delivery to Evaluation (M)	
System and Subsystem Description of Implementation (Please attach document)	

The following are definitions for the non-LTSA component labels above:

- **Claimant's Name, Organization, Address:** The contact name, the organization name, and the postal address associated with the Implementation Conformance Statement (ICS).
- **Date of Claim:** The date the ICS was claimed.
- **Name of Implementation:** A human readable identifier associated with the implementation.
- **Version of Implementation:** The version number associated with the ICS claim.
- **Digital Signature/Checksum:** If available, the hash algorithm type and the checksum associated with the implementation. Note 1: This element of the ICS is optional. Note 2: The method of calculating the digital signature/ checksum is implementation-defined.
- **URL for Related Resources:** If available, a Uniform Resource Locator of informative resources that are associated with the implementation. Note: This element of the ICS is optional.
- **System and Subsystem Description of Implementation:** If available, additional descriptive information associated with the implementation. Note: This element of the ICS is optional.

.

13 Annex E: Illustrations of conforming implementations (informative)

This Annex is informative and not normative.

This Annex contains examples and illustrations of implementations that conform to and map to this Standard.

13.1 Conceptual vs. actual implementations

This subclause is informative and not normative.

An important feature of the LTSA is the mapping of the "conceptual" implementation to the "actual" implementation. Actual implementations, 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, preamplifier, and amplifier are separate "audio architecture" components but, typically, they are manufactured together as a "stereo receiver".

The following diagrams show sample mappings of subsets of the LTSA system components (conceptual) to actual systems.

Note 1: These diagrams summarize the mappings. More detailed system decomposition and mappings would add precision to the diagrams.

Note 2: These diagrams are illustrations of potential mappings. Other LTSA abstraction-to-implementation mappings are possible for each illustration, i.e., the diagrams illustrate *a* mapping, not *the* mapping.

13.2 Tight component integration: web browser

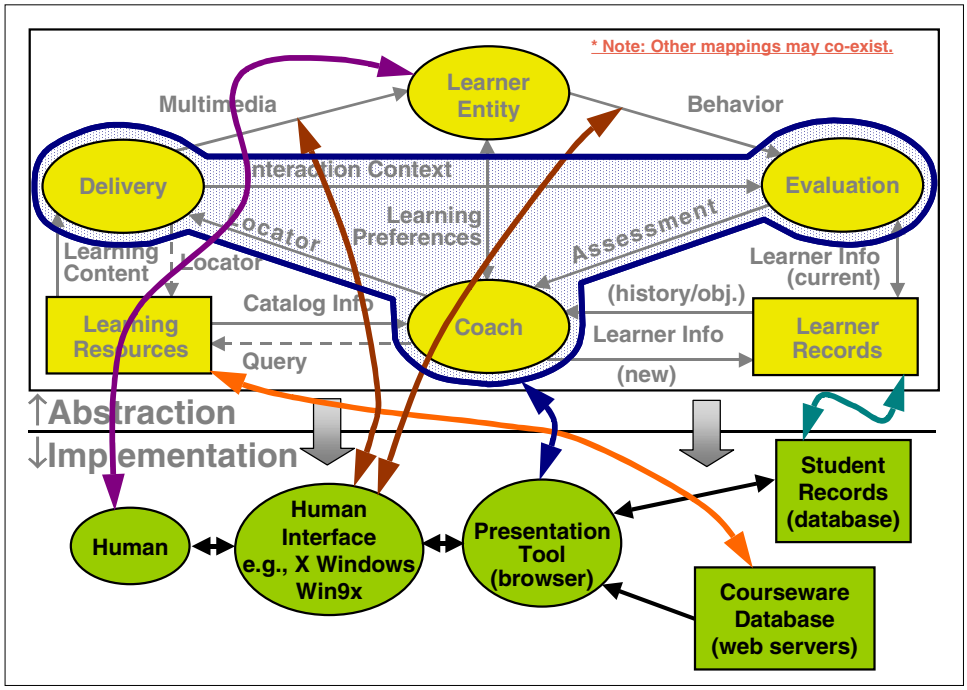


Figure 31. Example of tight integration of LTSA system components. The presentation tool (browser) represents several LTSA system components.

A hypothetical web-based learning technology system can be mapped to the LTSA system components. The diagram above shows the individual mappings of LTSA components to/from the web-based system . The Human is mapped to the LTSA learner entity, the Courseware Database is mapped to the LTSA learning resources, and Student Records are mapped to the LTSA learner records. The Presentation Tool (Web Browser) is mapped to (an implementation of) the LTSA evaluation, assessment, coach, locator, and delivery components. The Human Interface (e.g., X Windows or Microsoft Windows) is mapped to the LTSA behavior and multimedia components.

This diagram illustrates tight integration of several LTSA components, i.e., delivery, coach, evaluation, etc., and their mapping to the single implementation component, the Presentation Tool (web browser).

13.3 IDEALS Modular Training System

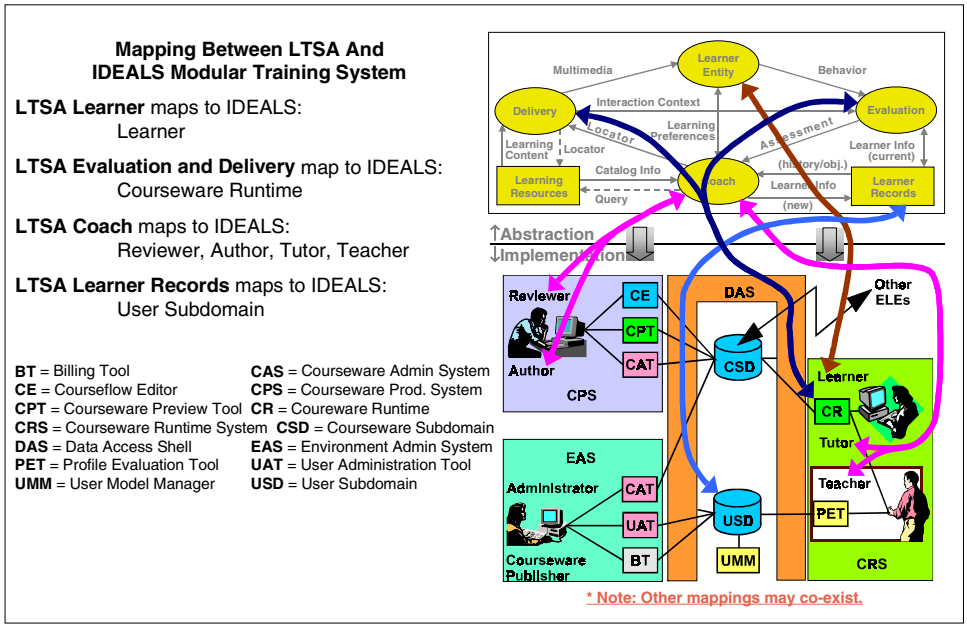


Figure 32. Components and LTSA mapping of the IDEALS Modular Training System.

The above mapping shows an actual collaborative university learning environment when mapped to the LTSA system components. The actual implementation has (1) many more implementation components than the LTSA system components, (2) components that don't map directly (but indirectly!) into LTSA system components (e.g. quality assurance by the Reviewer role), (3) functional components that address features below the abstraction level of the LTSA (e.g., the Author role).

Throughout this document, the abstract System Component view does not distinguish between human or machine representation of Processes and Stores. In the above block diagram of the implementation, however, this distinction is made by notifying human roles.

Some comments on the mapping of the roles Teacher, Tutor, Author, and Reviewer into the Coach process may ease its understanding. Predominantly, the human Teacher is substituted by a teacher agent (that is represented by the Courseware Runtime and script modules from the Learning Resources Store). Author and Reviewer are in charge of producing these scripts and the other learning resources and by this take part in the Coach Process. Even the roles Administrator and Courseware Publisher, and the User Model Manager process might be mapped to the Coach, due to their influence on courseware access rights, on the extent of courseware provision, and on learner model management. In group sessions, the Learner role may also take part in the Coach Process.

13.4 Shared component responsibility: flight simulator and instructor

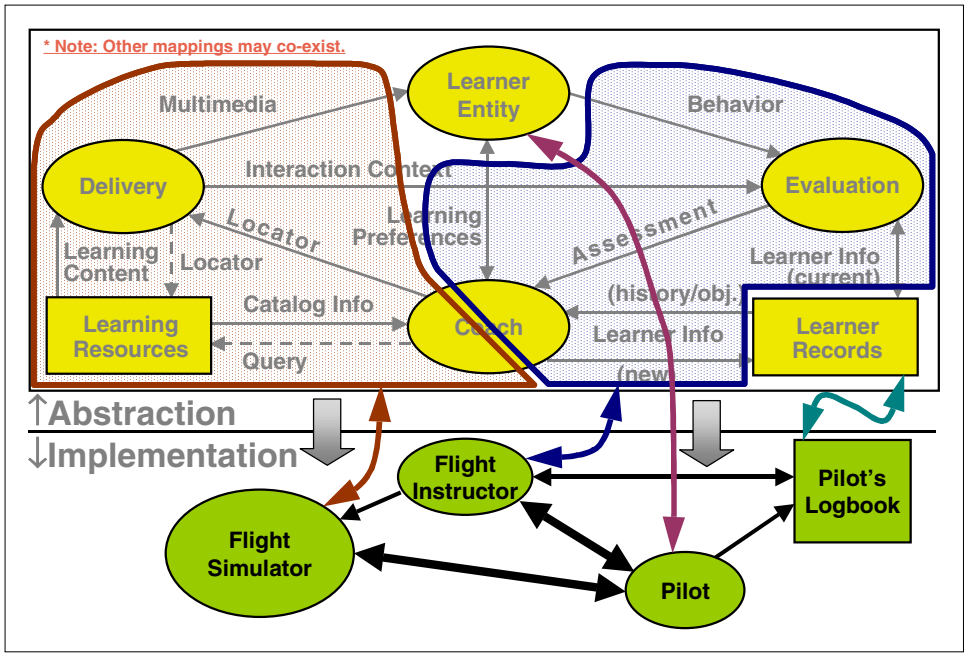


Figure 33. Example of a flight simulator, flight instructor, and pilot. Shows automated delivery (flight simulator) combined with human evaluation (flight instructor) and shared coach. Pilot maintains learner records in his/her logbook.

The collaboration of a Pilot, Flight Instructor, and Flight Simulator can be mapped to LTSA components. The Flight Simulator is implemented as a tight integration of the LTSA coach (portion), query, catalog info, locator, learning resources, learning content, delivery, and multimedia components. The Flight Instructor is represented as the LTSA interaction context, behavior, evaluation, learner information, assessment information, and coach (portion) components. The Pilot's Logbook is represented as the LTSA learner records component.

Note: The behavior is represented as part of the Flight Instructor and not shared with the Flight Simulator. This may seem counterintuitive at first because the Pilot "flies" the Flight Simulator, but the Flight Simulator does not *evaluate* the Pilot — the Pilot only *interacts* with the Flight Simulator. Thus, the only behavior of interest is the Pilot's behavior observed by the Flight Instructor because only the Flight Instructor does evaluation.

The diagram above illustrates a single LTSA system component (coach) mapped into more than one implementation component (Flight Simulator, Flight Instructor).

13.5 Student has multiple roles/responsibilities: self-paced courses

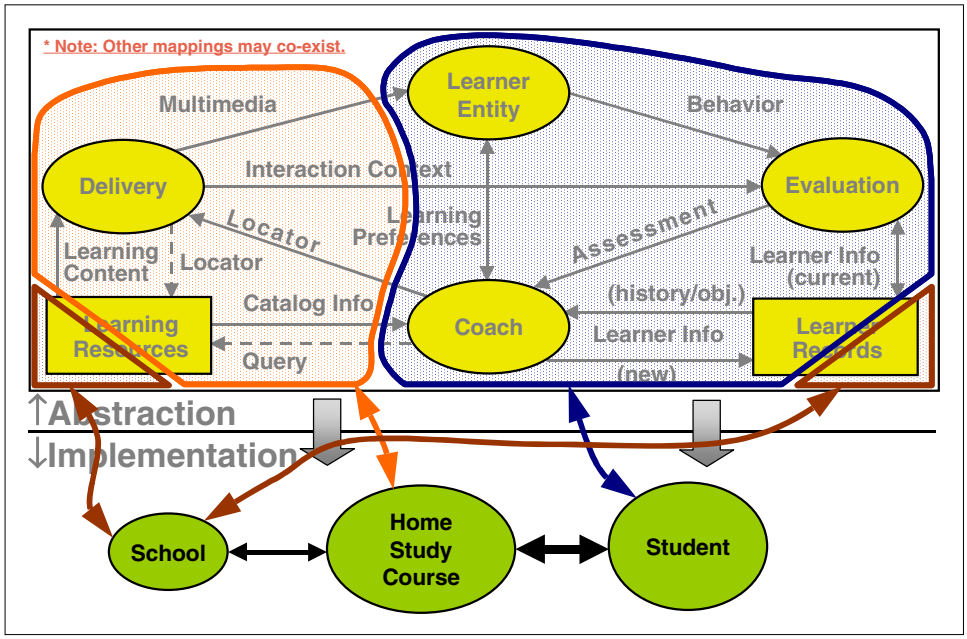


Figure 34. A home study or self-paced course. The school provides the study materials and, possibly, maintains some of the student's grades and certifications. The student progresses at his/her own pace. The student "directs" his/her own learning.

The home study or self-paced course can map to the LTSA system components. This example is important because the Student is represented as many LTSA components: learner entity, interaction context, behavior, evaluation, assessment information, learner information, learner records (portion), and coach.

The diagram above illustrates a Student (an individual human) that has several roles and responsibilities (learner entity, coach, evaluation, etc.).

13.6 Limiting case: non-electronic, traditional classroom

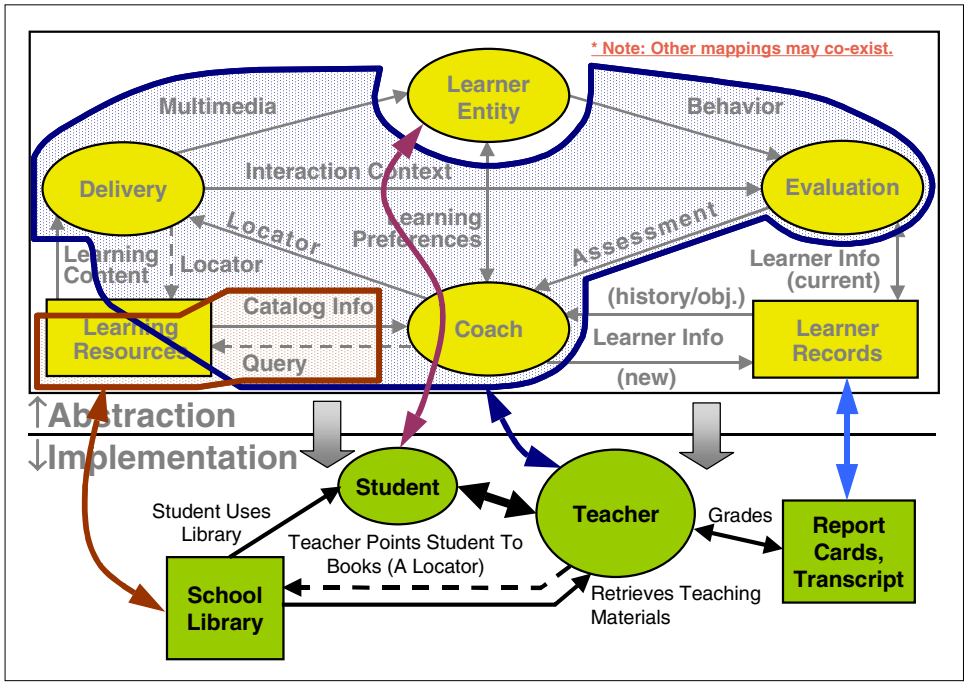


Figure 35. A traditional classroom (teacher, student, library, and report cards) mapped to LTSA system components. The LTSA can be used in non-electronic scenarios.

A non-electronic, traditional classroom can map to the LTSA system components. While mapping LTSA system components to a system void of technology might seem purposeless, this mapping is important because it addresses the "limiting case" of technology, i.e., no learning technology. Since LTSA is applicable in the "limiting case", the LTSA is applicable in a wide spectrum of actual implementations from low technology (e.g., classroom) to high technology (e.g., flight simulator and intelligent tutor).

14 Annex F: Conformance to IEEE 1471 (informative)

IEEE 1471, Recommended Practice for Architecture Description, is a document that provides guidance for describing architectures. In particular, it recommends:

- Identifying stakeholders.
- Identifying stakeholders' concerns.
- Identifying stakeholders' views.

The LTSA satisfies the recommendations of IEEE 1471 architecture descriptions by providing the Stakeholder Perspectives, as described in Annex C, Illustrations of Stakeholder Mappings, in this Standard.

15 Annex G: Methodology (informative)

This Annex is informative and not normative.

Note: This Annex will be removed prior to publishing of this Standard.

This Annex explains the methods and techniques used for analysis, synthesis, and refinement (i.e., development) of the Learning Technology Systems Architecture (LTSA). *This Annex is not a tutorial* — there are many books and materials on systems engineering and architecture. This Annex is intended to be a brief overview of the process.

Developer overview (informative)

The methodology used to develop this architecture and its layering is based on the Yourdon systems analysis methodology. Text description, system notation, bus notation, and combinations are used to describe each of the layers.

The Yourdon techniques for software engineering have been used for over 20 years and for the development of hundreds of thousands of systems.

Administrator, teacher, and learner overview (informative)

The purpose of outlining the methodology is to reveal the rigorous development process in use — this Standard was not "created in a vacuum". The methodology used to develop this architecture is based on common techniques used in software engineering for twenty years and for tens of thousands of systems. The methodology is "tried and true". At times, this type of methodology requires the application of "judgment calls", based on the experience of the architects and engineers who perform this kind of technical analysis.

15.1 Information inclusion

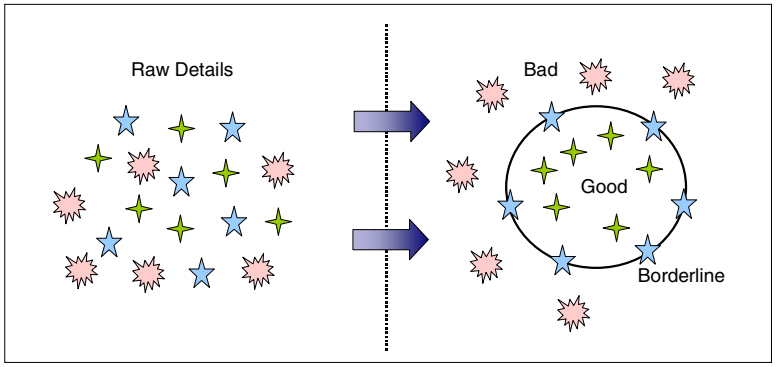


Figure 36. Organize raw details into good examples, borderline examples, and bad examples.

Information is gathered on the topic (in this case, learning technology systems). There are three types of raw information:

- Information that supports the main theme of the architecture and provides good examples — information that helps answer the question "What is in the scope of the hypothesis?". Information and examples of this kind will be used in the main description of the architecture.
- Information that is somewhat supportive of the main architectural theme, but does not provide the best examples — information that may answer the question "What is a borderline case?". This information will be used to determine the boundaries of the concepts.
- Information that is unrelated or contradicts the main theme — "What is outside the scope of the hypothesis?". Hopefully, there are few contradictory examples, or there is some rationale for the contradictions. This information will be used to identify topics that are inapplicable, unrelated, or outside the scope of the specification. In some cases, it may be easier to describe what is "outside" than what is "inside".

15.2 Information exclusion

When organizing information at a particular level of detail, it may be important to exclude certain information at that level. For example, when discussing the architecture of a house, it might be distracting and counterproductive to attend to plumbing issues; but plumbing issues may be relevant when drawing up blueprints.

Information exclusion might be better addressed by asking the question: what details obscure the main focus or main theme? For example, a cartographer does not include all details when creating a map — his/her judgment is based on the purpose and use of the map being created. This point may be illustrated by considering the diagramming of a web browser and web server as they are integrated into a larger system.

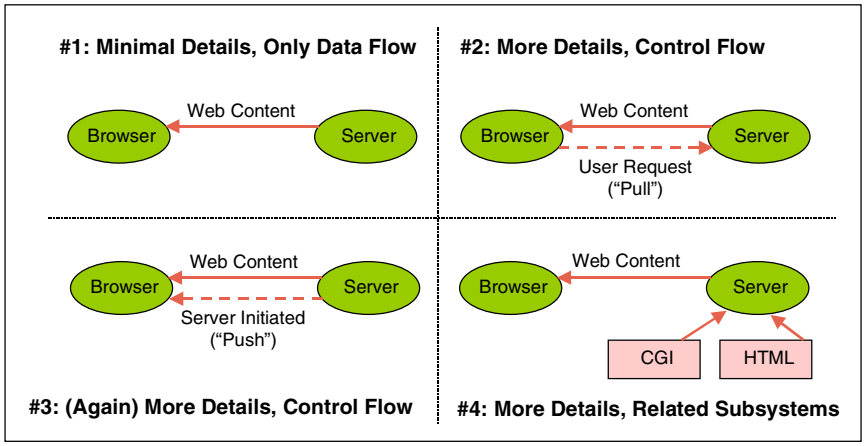


Figure 37. Essential vs. non-essential details for, say, web browsers and servers. The choice of which details to include/exclude is dependent on context and use. Compare to cartography: the choice of which details to include/exclude when map-making is dependent on the use and context of the map.

In #1, the web browser and web server are diagrammed simply as: the web server transferring web content to the web browser. Of course, there are more details to this transaction, such as who initiated the transfer (#2, #3) and the source of files and programs (#4) that are served up to the client. But are these details necessary? If the analysis requires addressing those details now (e.g., incorporation of "push" and "pull" technology) then it may be appropriate to include those details. However, if the diagram can still be understood *without* the details, then the details should be excluded and, possibly, incorporated into lower refinement layers.

Excluding non-essential details greatly improves the readability, interpretation, analysis, and acceptance of diagrams and notations.

Note: Throughout the LTSA specification, certain "judgment calls" have been made on what to include and exclude for the sake of clarity, utility, and consensus.

15.3 Organizing details

Once the raw information is collected, the information is organized by levels of detail or "granularity". Information of coarser granularity (lesser detail) is organized as "abstractions" while information of finer granularity (greater detail) is organized as "implementations". There may be a need to identify several "refinement layers", i.e., not just one abstraction level and one implementation level but levels in between.

Information at the same level of detail is organized into function groupings called subsystems.

Miscellaneous information that is important, but does not describe a system boundary, a system interface, or system functionality ("what it does"), is kept separately and called "implementation constraints". For example, in a computer networking system the main purpose of the system is to transfer information, but cost and security might be implementation constraints.

15.4 Boundaries

Implementation-implementation boundary

For information at the same level of detail, each subsystem that is identified must have a clearly defined boundary. Subsystem boundaries are called "implementation-implementation" boundaries, e.g., an interface between two subsystems.

Abstraction-implementation boundary

For information at different levels, each refinement layer must have a clearly defined boundary between its abstraction above and its implementation below. Refinement boundaries are called "abstraction-implementation" boundaries.

(NOTE: "abstraction-abstraction" boundaries refer to two abstractions that point to the same implementation. Currently, this technique is not necessary for this Standard.)

15.5 Notation conventions

Three primary notation conventions are used throughout the LTSA: text description, system notation, and bus notation. The choice of notation is dependent on the nature of the subject. System notation is useful when there are a handful of subsystems, the functions or roles of each subsystem is well-defined, and the connections between subsystems are established and unchanging (see LTSA layers 1, 3, 4). Bus notation is useful when there are a large number of subsystems, the functions or roles are undefined or changing over time, and the connections between subsystems are dynamic or on-demand (see LTSA layer 5). Text description is useful when the subsystem boundaries are unclear, the subsystem boundaries are overlapping, or the system and bus notations are impractical (see LTSA layers 2 and 4).

15.6 Text description

The subsystems at the same level of detail are described via text description rather than a diagrammatic notation. Diagrammatic notations, typically, require firm subsystem boundaries and non-overlapping subsystems. If the boundaries of a subsystem are not well-defined, a textual description might be a useful placeholder until the boundaries are defined. If well defined boundaries are not possible, a text description might be the only technique for describing the layer.

The LTSA layer 2, Learner-Related Design Features, uses text description.

The LTSA layer 4, Stakeholder Perspectives and Priorities, uses a combination of systems notation and text description.

15.7 System notation

For subsystems at the same level of detail, each subsystem is connected, as appropriate, to other related subsystems. These connections define the flows between subsystems. Subsystem and flow analysis has been described in many good books, including several by Ed Yourdon (see "http://www.yourdon.com"). The following is a very brief summary of the notation.

The LTSA layer 1, learner-environment interactions, uses system notation. Layer 1 consists of 2 processes and 1 data flow. One of the processes (learner entity) actually represents several subsystems (multiple learners represent the collective learner entity).

The LTSA layer 3, system components, uses system notation. Layer 3 consists of 4 processes, 2 data stores, and 9 one-way data flows, and 2 two-way data flow.

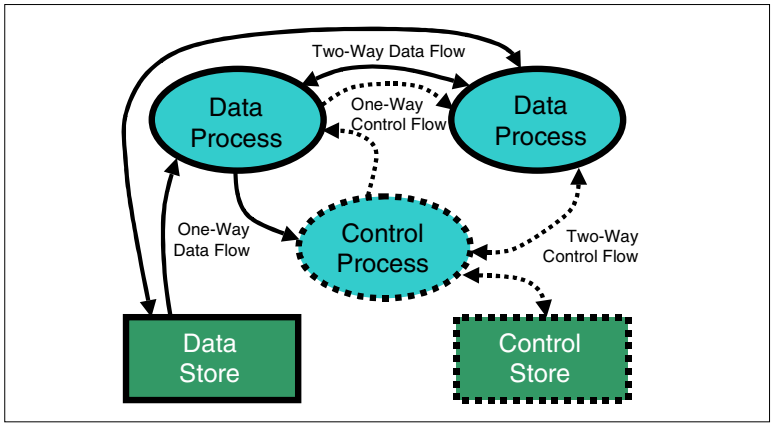


Figure 38. System notation uses components: process, stores, and flows. Processes can manipulate data information, control information, or both. Stores can hold data or control information. Flows can transfer data or control information. Flows can be one-way or two-way.

15.7.1 Processes

Processes are represented by ovals. A process subsystem is a (generic) process, i.e., something that is "alive" and able to transform its inputs into outputs. For example, a *data transformation process* transforms data inputs into data outputs. A data transformation process is represented by an oval drawn with a *solid* line. Information technology systems are mostly data transformation processes. Some information technology systems are called real-time systems. Real-time systems include both data transformations and control transformations. A *control transformation process* transforms control inputs into control outputs. A control transformation process is represented by an oval drawn with a *dashed* line. In practice, many processors combine a mixture of control inputs, control outputs, data inputs, and data outputs. A transformation that operates on a mix of both data and control is represented as a data transformation process, i.e., an oval drawn with a *solid* line.

15.7.2 Flows

Arrows are used to represent connections between the subsystems of the whole system. The arrows represent the flow of information, data or control, between subsystems.

15.7.2.1 One-way flows

Information that flows in a single direction from one subsystem to another is called a one-way flow. In the case of connecting two subsystems, a one-way flow has a single *source* (origin) *AND* a single *sink* (destination). In the case of connecting multiple subsystems, a one-way flow has a single *source* (origin) *OR* a single *sink* (destination); a single source with multiple sinks or a single sink with multiple sources is considered a one-way flow. In considering data flows, it is important to distinguish between the direction of data transfer (data flow) and the person, agent, or system that initiated the data transfer (control flow). For example, when a person uses a web browser to call up a page from a web server, this transaction is notated as a data flow going from the server to the browser, i.e., a data output from the server connected to a data input of the browser, even though the person (browser) initiated the request. In other words, for web "push" and "pull" technology, the data flow is the same (i.e., a one-way flow from server to browser), but "pull" technology emphasizes the control flow (e.g.,

initiating the request) from browser to server, while "push" technology emphasizes the control flow from server to browser.

15.7.2.2 Two-way flows

Information that flows in both directions between subsystems is called a two-way flow, e.g., telephone calls (two-way data flow), video conferencing (two-way data flow), network routing information (two-way control flow). In many cases, a protocol is used to organize the two-way flow of information between two subsystems.

A special case concerns the "updating" of information in a data store (e.g., a database) or a control store (e.g., event history). For example, when a record is updated in a database, typically, the record is retrieved, modified, and then stored. For notational purposes, there is no distinction between the creation of the original database record and its subsequent update (the modification of only certain fields): both creation and updating are notated as one-way flows into the store.

15.7.2.3 Data flows

The most common flow is the data flow, which represents data moving among two or more subsystems. A data flow connected to a subsystem represents data input (a one-way flow), data output (a one-way flow) or both (a two-way flow). Data is the unit of information, relative to the level of descriptive detail, that represents the main processing of the whole system (see 2.7.2.5, Data vs. Control, below). For example, in a subsystem that adds numbers (a calculator), the flow that supplies the numbers to be added (data entry) might be an input data flow; while the flow that emits the result of the calculation (output display) might be an output data flow. Data transferred in both directions is an input/output data flow (a two-way flow), e.g., data transferred over a modem.

15.7.2.4 Control flows

The control flow represents inputs and/or outputs that control the processing of data among two or more subsystems. A control flow can be a control input (a one-way flow), a control output (a one-way flow), or both (a two-way flow). In the example above of the calculator subsystem that adds numbers, the signal to start calculating the sum (the ADD or "=" key) is an input control flow, and the signal to indicate that a calculation has overflowed (the ERROR light) is an output control flow. Control transferred in both directions is an input/output control flow (a two-way flow), e.g., the signaling that negotiates modem speed and quality when modems connect.

15.7.2.5 Data vs. control

The choice of calling information data or control is somewhat arbitrary. For example, establishing a connection between a web browser and a web server might be described as *control information from the perspective of web applications*, but would be described as *data information from the perspective of network routers*. A guideline might be: (1) if the information in question is involved in the main purpose of the system or the main inputs and outputs, it would be labeled as data information; (2) if the information changes the processing of information, starts or stops processing, or starts or stops the flow of information, it would be labeled as control information.

15.7.3 Stores

A store holds information. A *data store* holds data information, e.g., a database. A *control store* holds control information, e.g., an event history.

15.8 Example of systems notation tiers/hierarchies

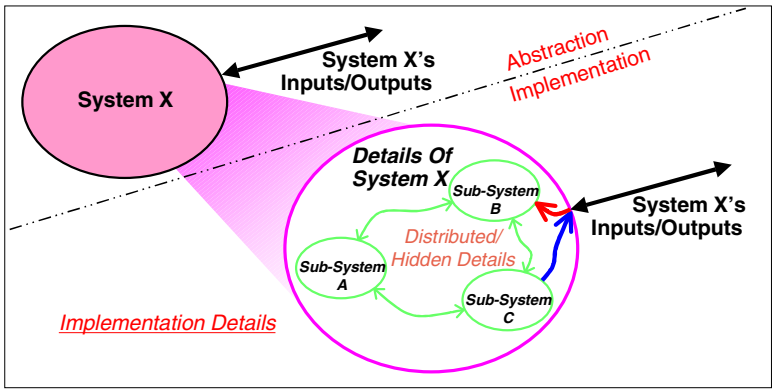


Figure 39. Example of a hierarchy of systems notations.

The diagram above shows a system notation used to define a subsystem of a larger system (that uses system notation, too).

15.9 Iterative abstraction

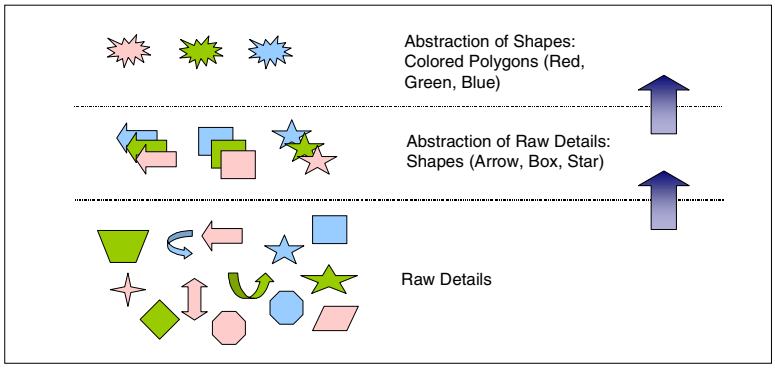


Figure 40. Abstracting raw details to higher level abstractions. This diagram shows three layers but, typically, there are more (the LTSA has five). The abstraction process continues until the system is reduced to roughly 3-7 components.

Once information has been gathered, the conceptual understanding is described using one or more of the techniques above (other notations are possible, too). For example, a useful architecture for a given system may require three levels of abstraction: (1) take raw information and create a low-level abstraction; (2) take the low-level and create a mid-level abstraction; and (3) take the mid-level and create a high-level abstraction, i.e., the architecture.

After each cycle of abstraction, a new, higher-level abstraction is produced. Another output from the abstraction technique is a set of implementation constraints, i.e., features of the implementation (e.g., cost, performance) that are not represented in the functionality ("what it does") of the abstraction.

The abstraction technique (creating higher level abstractions, based on implementations and/or lower level abstractions) is repeated until the system is reduced to a handful of components, e.g., 3-7 components. The number of abstraction levels equals the number of steps required to reduce the system to a handful of components.

15.10 Iterative implementation

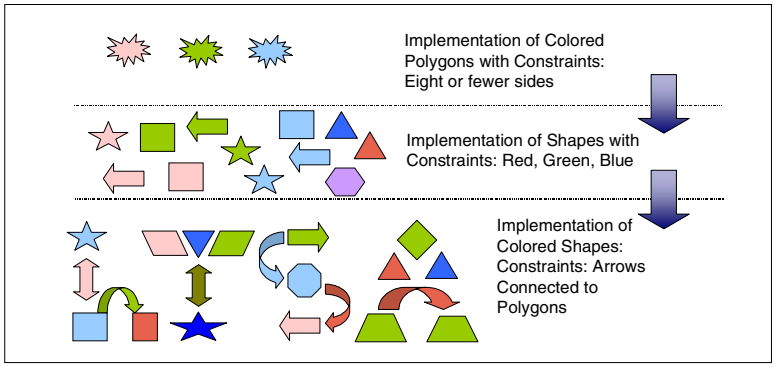


Figure 41. Re-implementing the system. Each abstraction creates an implementation via constraints. The system is re-implemented but architectures (high level abstractions) may reveal new structures and/or create new systems not previously conceived.

Once the highest-level abstraction is reached (the architecture specification), the system is then re-implemented on the basis of that abstraction (most general concepts). Re-implementing the system consists of implementing each abstraction layer in the context of implementation constraints. In other words, the system is built solely based on the specification. An "implementation" needn't be an actual functioning system: an implementation might be a more detailed description of a higher level abstraction (concept).

If the newly implemented system (e.g., a high-level abstraction now built as a mid-level implementation) produces the same functionality, interfaces, services, and qualities (implementation constraints) as the original mid-level specification, then the iterative implementation process continues (success). If the newly implemented system does not match the requirements of the original mid-level specification, then the higher-level abstraction is incorrect or the implementation constraints are poorly defined (failure: go back and correct a previous step).

The iterative implementation process is continued until the last (lowest-level) abstraction is implemented successfully, i.e., matches the original requirements and desirables of the existing and emerging systems in the data gathering step.

The purpose of re-implementing the system is to verify the correctness of the abstractions and the implementation constraints. When completed, this process validates the highest-level abstraction — the architecture specification. In practice, an architecture specification is not completely re-implemented, but only the high-risk portions are prototyped. What is essential is the satisfaction of critical functional requirements (e.g., Is the high-level system still a learning technology system?) and/or important implementation constraints (e.g., Can the multimedia be delivered over the internet? Can the learner collaborate cost-effectively with other learners?).

15.11 Judgment calls

The methodology is merely a guideline — sometimes there is no clear "right answer". Sometimes, "judgment calls" are required, based on the experience of the architects and engineers. The following

are examples of "judgment calls" in the development of the LTSA. (See also 2.1 Information inclusion, and 2.2 Information exclusion.)

Example #1

In LTSA layer 1, environment's effect on the learning is shown by a *one-way* arrow, but why isn't the learner's effect on the environment shown, e.g., two-way arrow? Learners *do* have an effect on their environment (e.g., a learner's effect on a student teacher; the effect of the learner's research that addresses the state of the art). While the learner's effect on his/her environment might be a part of the learning experience (e.g., teacher training, graduate school), these details are less important to the main theme of applying learning technology to learning experiences.

Example #2

In the LTSA system components, the flow between delivery and learning resources is a locator (control flow) requesting some learning content (data flow). There may be other information, too (e.g., electronic commerce, intellectual property rights management), but these two arrows (one data flow, one control flow) still represent the main theme. Important: A one-way arrow does not prohibit flow in the opposite direction or two-way flows at lower abstraction-implementation layers.

Example #3

In the LTSA system components, both the "queries" (queries control flow) to the learning resources and their "responses" (catalog info data flow) are diagrammed, but similar "queries" to the learner records are not diagrammed — only the "responses" (coach extracting learner entity's history) are diagrammed. The generation of "queries" is a significant step of the coach processing. The extraction request upon the learner entity's history (i.e., a "query" to the learner records) is not a significant design issue for the coach so only the "responses" are diagrammed. See subclause 15.2, Information Exclusions, in this Annex for further information on a rationale for excluding features.

15.12 Summary

Abstracting, architecting, and implementing a system involves many iterations of "trying concepts" (abstractions) and "building them to verify that the abstractions work" (implementations). Many abstractions and implementations are possible with no single "right" answer. Architectures are high-level abstractions that represent a range of systems, not a single system at a single point in time: architectures are general solutions that are adaptable over time.

16 Annex H: Document development (informative)

This Annex is informative and not normative.

Note: This Annex will be removed prior to publishing of this Standard.

This Annex concerns the development of this Standard. The past (revision history and resolved issues), present (release notes and comment returns), and future (open issues) releases of this Standard are identified here.

16.1 Revision history

The following are the revisions of this Standard and summary information for each revision:

- Draft 1, 1996-12-05, initial draft. Presentation at the 1996-12 meeting of IEEE P1484.
- Draft 2, 1997-01-28. The initial draft of the top three refinement layers. This release was largely incomplete, but was intended to give the reader a first glance at the work in progress.
- Draft 3, 1997-09-23. Combined presentation and paper. Completed top four refinement layers: knowledge exchange, human-centered features, system components, and stakeholder perspectives. Initial draft of operational components.
- Draft 4, 1998-05-21. Completed operational components and conformance Clauses. Updated diagrams. Rewrote text for lay audience (college level, non-technical).
- Draft 5, 1999-12-06. Revised operational components and interoperability Clause. Revised layer 2 to include motivations for learner-directed learning.
- Draft 6, 2000-11-14. Converted to IEEE format. Reorganized the document so that only layer 3 (system components) is normative.
- Draft 7, 2001-03-11. Incorporated Working Draft ballot comments.
- Draft 8, 2001-04-06. Incorporated changes, as per 2001-03 1484.1 WG meeting.
- Draft 9, 2001-11-xx, current draft. Incorporated changes, as per ballot resolution of drafts 6, 7, and 8.

16.2 Release notes for this document

The following notes apply to this draft of this Standard.

- This draft Standard is ready for Sponsor Ballot in IEEE LTSC.
- This draft Standard is may be reviewed among standards committees and fora.

16.3 Resolved issues

The following issues have been resolved:

1. The sections comparing several learning issues (intentional vs. non-intentional, organic vs. inorganic) have been removed since they are not important for presenting the main technical discussion and rationale.
2. Diagrams have been added to clarify the system and component organization.
3. The "knowledge exchange" layer has been substantially rewritten and relabeled as "learner-environment interactions".

4. Missing sections have been completed.
5. Graphics have been completely redesigned.
6. A flow has been added between the delivery and evaluation components to provide context to the evaluation process for the learning experiences of the delivery process.
7. Conversion to IEEE format.
8. Layers 1, 2, 4, and 5 have been moved to informative wording. Layer 3 remains as normative wording.
9. A conformance label has been specified.
10. Conformance wording has been improved.
11. Removed extraneous informative wording on human-centered features.
12. Removed Annex on background information.
13. Added more descriptive wording to Pro-Forma ICS.
14. Shorten wording in Clause 1, Introduction.
15. Move Methodology Annex to end.
16. Resolve comments received after Draft 6 ballot.

16.4 Open issues

The following are open, unresolved, and/or outstanding issues for this Standard:

1. Sponsor Ballot.

16.5 Comments on this document

All comments are appreciated. Please return all comments on this release of this document by **Friday, 2001-11-15 23:00 UTC**. Please deliver all comments to the IEEE 1484.1 Architecture and Reference Model Working Group by sending E-mail to:

ltsc-arch@majordomo.ieee.org

To subscribe to the working group mailing list, send the one-line message

subscribe ltsc-arch

to the E-mail address "majordomo@majordomo.ieee.org".

The Technical Editors may be contacted at any one of the following:

Telephone: +1 212 486 4700
Fax: +1 212 759 1605
E-mail: frank@farance.com or tonkel@farance.com

Frank Farance or Joshua Tonkel
Farance Inc. (Edutool division)
Island Box 256
New York, NY
10044-0205
USA