



Standard Guide for Laboratory Information Management Systems (LIMS)¹

This standard is issued under the fixed designation E 1578; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last reapproval. A superscript epsilon (ε) indicates an editorial change since the last revision or reapproval.

1. Scope

1.1 This guide covers issues commonly encountered at all stages in the life cycle of Laboratory Information Management Systems from inception to retirement. The sub-sections that follow describe details of scope of this document in specific areas.

1.2 *High Level Purpose*—The purpose of this guide includes: (1) help educate new users of Laboratory Information Management Systems (LIMS), (2) provide standard terminology that can be used by LIMS vendors and end users, (3) establish minimum requirements for primary LIMS functions, (4) provide guidance for the specification, evaluation, cost justification, implementation, project management, training, and documentation, and (5) provide an example of a LIMS function checklist.

1.3 *LIMS Definition*—The term Laboratory Information Management Systems (LIMS) describes the class of computer systems designed to manage laboratory information.

1.4 *Laboratory Categories*—The spectrum of laboratories that employ LIMS is wide spread. The following break down provides an overview of the laboratory categories that use LIMS as well as examples of laboratories in each category.

1.4.1 *General Laboratories:*

1.4.1.1 Standards (ASTM, IEEE, ISO), and

1.4.1.2 Government (EPA, FDA, JPL, NASA, NRC, USDA, FERC).

1.4.2 *Environmental:*

1.4.2.1 Environmental Monitoring.

1.4.3 *Life Science Laboratories:*

1.4.3.1 Biotechnology,

1.4.3.2 Diagnostic,

1.4.3.3 Healthcare Medical,

1.4.3.4 Devices, and

1.4.3.5 Pharmaceuticals Vet/Animal.

1.4.4 *Heavy Industry Laboratories:*

1.4.4.1 Energy & Resources,

1.4.4.2 Manufacturing & Construction,

1.4.4.3 Materials & Chemicals, and

1.4.4.4 Transportation & Shipping.

1.4.5 *Food & Beverage Laboratories:*

1.4.5.1 Agriculture,

1.4.5.2 Beverages,

1.4.5.3 Food, and

1.4.5.4 Food Service & Hospitality.

1.4.6 *Public Sector Laboratories:*

1.4.6.1 Law Enforcement,

1.4.6.2 State & Local Government,

1.4.6.3 Education, and

1.4.6.4 Public Utilities (Water, Electric, Waste Treatment).

1.4.7 *Laboratory Size:*

1.4.7.1 This guide covers topics regarding LIMS for a range of laboratory sizes ranging from small with simple requirements to large multi-site/global laboratories with complex requirements. Although the guide addresses complex issues that impact primarily large LIMS implementations, laboratories of all sizes will find this guide useful. The implementation times and recommendations listed in this guide are directed at medium and large laboratories.

1.5 *Integration*—Integration between LIMS and other external systems (document management, chromatography data systems, laboratory instruments, spectroscopic data systems, Enterprise Resource Planning (ERP), Manufacturing Execution Systems (MES), Corrective Action and Preventative Action (CAPA), Electronic Laboratory Notebooks (ELNs) and data archive) provides significant business benefits to any laboratory. Integration between LIMS and other external systems is discussed at a high level in this guide including data interchange and XML standards.

1.6 *Lifecycle Phases*—The LIMS lifecycle described in this guide includes the following phases: (1) project initiation, (2) requirements analysis, (3) design, (4) build/configure, (5) test and commission, (6) operation and maintenance, and (7) retirement. This guide is intended to provide an understanding of the LIMS system life cycle and good practices for each of the activities. It will help first time LIMS implementers plan and manage their LIMS projects while seasoned LIMS users

¹ This guide is under the jurisdiction of ASTM Committee E13 on Molecular Spectroscopy and Chromatography and is the direct responsibility of Subcommittee E13.15 on Analytical Data.

Current edition approved Sept. 1, 2006. Published September 2006. Originally approved in 1993. Last previous edition approved in 1999 as E 1578 – 99 (1993).

may use the LIMS system life cycle to maintain existing LIMS and prepare for the implementation of the next generation LIMS.

1.7 Audience—This guide has been created with the needs of the following stakeholders in mind: (1) end users of LIMS, (2) implementers of LIMS, (3) quality personnel, (4) information technology personnel, (5) LIMS vendors, (6) instrument vendors, (7) individuals who must approve LIMS funding, (8) LIMS application support specialists, and (9) software test/validation specialist. Information contained in this guide will benefit a broad audience of people who work or interact with a laboratory. New LIMS users can use this guide to understand the purpose and functions of LIMS. The guide can also help prospective LIMS users in understanding terminology, configurations, features, design, benefits and costs. Individuals who are purchasing a LIMS may use this guide to identify functions that are recommended for specific laboratory environments. Research and Development staff of commercial LIMS vendors may use the guide as a tool to evaluate, identify, and potentially improve the capabilities of their products. LIMS vendor sales staff may use the guide to represent functions of their LIMS product to prospective customers in more generic and product neutral terms.

1.8 Out of Scope—The full description and use of systems mentioned in this guide within the context of interfaces to LIMS are beyond the scope of this standard. Examples of these systems include Chromatography Data Systems (CDS), Electronic Laboratory Notebooks (ELN), Data Archive, Scientific Data Management Systems (SDMS), Enterprise Resource Planning (ERP), Manufacturing Execution Systems (MES) and Electronic Document Management Systems (EDMS).

2. Referenced Documents

2.1 ASTM Standards:²

- E 1340** Guide for Rapid Prototyping of Information Systems
- E 1947** Specification for Analytical Data Interchange Protocol for Chromatographic Data
- E 1948** Guide for Analytical Data Interchange Protocol for Chromatographic Data
- E 2066** Guide for Validation of Laboratory Information Management Systems
- E 2077** Specification for Analytical Data Interchange Protocol for Mass Spectrometric Data
- E 2078** Guide for Analytical Data Interchange Protocol for Mass Spectrometric Data

2.2 ANSI Standards:³

- ANSI HL7** Arden Syntax for Medical Logic Systems

2.3 EPA Data Standards:⁴

- 40 CFR 160**, Code of Regulations, 54 FR 34067, August 17, 1989

- 40 CFR 792, Code of Regulations, 54 FR 34043, August 17, 1989

- EPA 2185** Good Automated Laboratory Practices Principles and Guidance to Regulations For Ensuring Data Integrity In Automated Laboratory Operations with Implementation Guidance, 1995 Edition

2.4 FDA Regulations:⁵

- FDA 21CFR Part 11**, Code of Regulations, 57 FR 32185, July 21, 1992

2.5 GAMP:⁶

- GAMP 4** Good Automated Manufacturing Practice (GAMP) Guide for Validation of Automated Systems in Pharmaceutical Manufacture, ISPE, December 2001

- GAMP Good Practice Guide**, Validation of Laboratory Computerized Systems, April 2005 (this is a supplement to the GAMP 4 document)

- GAMP GPG**, IT Infrastructure Control and Compliance, ISPE, September 2005

2.6 IEEE Standards:⁷

- IEEE 829** 1998 IEEE Standard for Software Test Documentation

- IEEE 830** 1998 Recommended Practice for Software Requirements Specifications

- IEEE 1008** 1987 IEEE Standard for Software Unit Testing

- IEEE 1012** 2004 IEEE Standard for Software Verification and Validation

- IEEE 1028** 1997 IEEE Standard for Software Reviews

- IEEE 1063** 2001 IEEE Standard for Software User Documentation

2.7 Instrument Interface Standards:

- AnIML (Analytical Information Markup Language)**, an emerging standard for laboratory instruments covering multiple analytical techniques. The E13.15 subcommittee is responsible for the development of this standard.⁸

- NetCDF (Network Common Data Form)**, an interface for array-oriented data access and a library that provides an implementation of the interface. The netCDF library also defines a machine-independent format for representing scientific data. Together, the interface, library, and format support the creation, access, and sharing of scientific data. Unidata Program Center at the University Corporation for Atmospheric Research (UCAR).⁹

- Staged Electronic Data Deliverable (SEDD)**, EPA—eXtensible Markup Language (XML)—joint standard developed by US EPA Office of Superfund Remediation and Technology Innovation (OSRTI) Analytical Services

⁵ Available from Food and Drug Administration (FDA), 5600 Fishers Ln., Rockville, MD 20857.

⁶ Available from the International Society for Pharmaceutical Engineering (ISPE), 3109 W. Dr. Martin Luther King, Jr. Blvd., Suite 250, Tampa, FL 33607 USA.

⁷ Available from Institute of Electrical and Electronics Engineers, Inc., (IEEE), 445 Hoes Lan., P.O. Box 1331, Piscataway, NJ 08854-1331.

⁸ Reference Schaefer, B. A., Poetz, D., and Kramer, G. W., "Documenting Laboratory Workflows Using the Analytical Information Markup Language," *Journal of the Association for Laboratory Automation*, 9 (6), 2004, pp. 375-381, and <http://animl.sourceforge.net/>

⁹ Available from UCAR: <http://www.unidata.ucar.edu/software/netcdf/guidecd/guidecd-16.htm>

² For referenced ASTM standards, visit the ASTM website, www.astm.org, or contact ASTM Customer Service at service@astm.org. For *Annual Book of ASTM Standards* volume information, refer to the standard's Document Summary page on the ASTM website.

³ Available from American National Standards Institute (ANSI), 25 W. 43rd St., 4th Floor, New York, NY 10036, <http://www.ansi.org>.

⁴ Available from United States Environmental Protection Association (EPA), Ariel Rios Bldg., 1200 Pennsylvania Ave., NW, Washington, DC 20460.

Branch (ASB) and US Army Corps of Engineers (US ACE)¹⁰

ISO/IEC 12207 Subcommittee for Electronic Data Standards (SEDS), reference spectroscopic databases sponsored by the International Union of Pure and Applied Chemistry (IUPAC) and standards related to the Joint Committee on Atomic and Molecular Physical Data (JCAMP) and JCAMP-DX (XML in the chemistry area) ISO Standards ISO/IEC 12207 Information technology—Software life cycle processes, as amended by ISO/IEC 12207 AMD2¹¹

2.8 *NRC Standards*.¹²

FDA CFR Part 21 10 Code of Federal Regulations (CFR) Part 21.42 FR 28893, June 6, 1977

FDA CFR Part 50, Appendix B 10 Code of Federal Regulations (CFR) Part 50 Appendix B. 35 FR 10499, June 27, 1970, as amended at 36 FR 18301, Sept. 11, 1971; 40 FR 3210D, Jan. 20, 1975

FDA CFR Part 50, Appendix E 10 Code of Federal Regulations (CFR) Part 50 Appendix E. 45 FR 55410, Aug. 19, 1980, et sequentia as amended

FDA CFR Part 50, Appendix K 10 Code of Federal Regulations (CFR) Part 50 Appendix K. 21 FR 355, Jan. 19, 1956, unless otherwise noted

3. Terminology

3.1 This guide defines terminology used in the LIMS field. Section 3.2 defines LIMS terms specific to this guide. Users of this document should request a terminology list from each vendor with a cross reference to the terms used in this guide.

3.2 *Definitions of Terms Specific to This Standard*:

3.2.1 **CAPA**, *n*—acronym for corrective action, preventative action.

3.2.2 **Chromatography Data System (CDS)**, *n*—computer system used to acquire, analyze, store and report information from chromatography instruments.

3.2.3 **EDMS**, *n*—acronym for electronic document management system.

3.2.4 **ELN**, *n*—acronym for electronic laboratory notebook. computer system designed to replace paper laboratory notebooks.

3.2.4.1 *Discussion*—Lab notebooks in general are used by scientists and technicians to document research, experiments and procedures performed in a laboratory. A lab notebook is often maintained to be a legal document and may be used in a court of law as evidence. Similar to an inventor's notebook, the lab notebook is also often referred to in patent prosecution and intellectual property litigation. Electronic laboratory notebooks enable electronic access to information including searching, data capture from instruments and collaboration between laboratory personnel and personnel outside the laboratory.

3.2.4.2 *Discussion*—ELNs can be divided into two categories: specific ELNs contain features designed to work with

specific applications, scientific instrumentation or data types. Refer to laboratory execution system (LES) as an example of a specific ELN. Cross-disciplinary ELNs or generic ELNs are designed to support access to all data and information that needs to be recorded in a lab notebook.

3.2.5 **ERP**, *n*—acronym for enterprise resource planning.

3.2.6 **GALP**, *n*—the GALPs are a union of federal regulations, policies, and guidance documents. Several of the GALP provisions are embodied in EPA's Good Laboratory Practice Standards (GLPs). The GLPs are regulations that govern the management and conduct of most nonclinical laboratory studies submitted to EPA's office of Toxic Substances and its Office of Pesticide Programs. Reference **EPA 2185**.

3.2.7 **GAMP**, *n*—acronym for good automated manufacturing practice.

3.2.8 **LES**, *n*—acronym for laboratory execution system. Computer system employed in the laboratory at the analyst work level to aid in step enforcement for laboratory test method execution.

3.2.8.1 *Discussion*—Laboratory execution systems (LES) are a sub class of electronic laboratory notebooks (ELNs) that focus on step execution of defined laboratory test methods. The LES are typically employed in Quality Control laboratories that have defined test methods. The functionality of LES and LIMS overlap in the areas of result entry, instrument integration and specification flagging. Deployment options include LES and LIMS systems deployed as an integrated solution, LIMS only or LES only (for limited functions).

3.2.9 **Laboratory Information Management System (LIMS)**, *n*—(1) acronym for laboratory information management System. Computer application(s) software and hardware that can acquire, analyze, report, and manage data and information in the laboratory; (2) computer software that is used in the laboratory for the management of samples, test results, laboratory users, instruments, standards and other laboratory functions such as invoicing, plate management, stability LIMS, work flow automation; and (3) a class of application software which handles storing and managing of information generated by laboratory processes. These systems are used to manage laboratory processes including defining master data, sample management and chain of custody, work assignment, instrument and equipment management, standard and reagent management, scheduled sample collection and testing, result entry, result review, reporting, trending and business rule enforcement. These systems interface with laboratory instruments (for example, chromatography data systems (CDS), spectrophotometers and balances) and other information systems such as enterprise resource planning (ERP), manufacturing execution systems (MES), or health care based laboratory information systems). A LIMS is a highly flexible application, which can be configured or customized to facilitate a wide variety of laboratory workflow models.

3.2.10 **LIMS configuration**, *n*—refers to the process of preparing the LIMS for use in a particular laboratory. It typically involves using an interface provided by the vendor to enter information that describes the types of samples, analytical methods, specifications, etc. used in the laboratory.

¹⁰ Available from US EPA <http://www.epa.gov/superfund/programs/clp/sedd.htm>

¹¹ Available from International Organization for Standardization (ISO).

¹² Available from U.S. Nuclear Regulatory Commission (NRC), PDR, 01F13, Washington, DC 20555.

3.2.11 *LIMS customization*, *n*—refers to the process of modifying the LIMS to meet the requirements of a particular laboratory. It typically involves adding tables, modifying table structures and writing code or programs to alter the behavior of the LIMS.

3.2.12 *metadata*, *n*—(1) data about data; and (2) information that describes another set of data.

3.2.12.1 *Discussion*—Metadata in the LIMS context typically includes all data that supports a test result that is recorded in a LIMS. Examples include for a pH test, a pH result can be supported by metadata including what instrument was used, what is the calibration date of the instrument, what standard buffer solutions (reagents) were used to calibrate the pH probe sensor and the expiration dates for the standard solutions. sample

3.2.13 *sample registration*, *n*—the process of registering samples in a LIMS.

3.2.14 *SDMS*, *n*—acronym for scientific data management system.

3.2.15 *spectroscopic data systems*, *n*—computer system used to process, visualize, interpret, store and report information from spectroscopic and non-chromatographic instruments.

3.2.16 *static tables*, *n*—descriptive LIMS database tables where templates, tests, calculations, specifications, and related information are defined and stored (commonly found in “look up/reference/dictionary” tables).

3.2.16.1 *Discussion*—LIMS stores look up information to speed sample registration and test assignments. Generally prior to sample registration the static tables need to be configured. Some LIMS implementations can enter static table information directly from the sample registration step. Synonyms of static tables include master data and configuration data.

4. Significance and Use

4.1 *Relevance*—This guide is intended to educate those in the intended audience on many aspects of LIMS. Specifically, the guide may:

- 4.1.1 Help educate new users of LIMS;
- 4.1.2 Help educate general audiences in laboratories and other organizations that use LIMS;
- 4.1.3 Help educate instrument manufacturers and producers of other commonly interfaced systems;
- 4.1.4 Provide standard terminology that can be used by LIMS vendors and end users;
- 4.1.5 Establish a minimum set of requirements for primary LIMS functions;
- 4.1.6 Provide guidance on the tasks performed and documentation created in the specification, evaluation, cost justification, implementation, project management, training, and documentation of LIMS; and
- 4.1.7 Provide high-level guidance for the integration of LIMS with the most commonly integrated systems such as laboratory instruments, CDS, ERP, ELN, SDMS and so forth.

4.2 *How Used*—This guide is intended to be used by all stakeholders involved in any aspect of LIMS implementation or maintenance.

4.2.1 It is intended to be used throughout the LIMS life cycle by individuals or groups responsible for LIMS including specification, build/configuration, validation, use, upgrades, retirement/decommissioning.

4.2.2 It is also intended to provide an example of a LIMS function checklist.

5. LIMS Concept Model—Graphic Picture of Functionality

NOTE 1—The LIMS primary functions, integration points, life cycle and industry segments are described in Fig. 1.

5.1 *LIMS Concept Model—High Level*—The core LIMS functions cover a wide range of laboratory workflows, information management and integration with other enterprise computer systems. These core LIMS functions are illustrated in the LIMS box. Integration with LIMS and a number of external systems, instruments, reporting tools and automated test tools are illustrated by the boxes above and below the LIMS box in Fig. 1.

5.2 *LIMS Concept Model—Mid Level*—The LIMS functions and their relationship with external systems, extended LIMS functions and future LIMS functions is illustrated in Fig. 2. The diagram defines:

5.2.1 Core LIMS functions (LC) with sub-divisions of LC-1.X Operations and LC-2.X Support;

5.2.2 Extended LIMS functions are described by items listed in boxes LE-X;

5.2.3 Systems external to LIMS are illustrated with boxes coded with E; and

5.2.4 Future LIMS functions are described in boxes coded with LF.

5.3 *LIMS Life Cycle Phases*—Fig. 3 defines the high level LIMS life cycle phases of (1) initial LIMS implementation, (2) LIMS operations, and (3) system maintenance. Each of these primary phases is further decomposed into primary functions. The numbering scheme use matches the mid level definitions in Fig. 2 and also tie to the requirements section located in Appendix X1.

5.4 *LIMS Concept Model—Industry Segments*—All laboratories require basic LIMS work flow including sample registration, assignment of tests, entry of results, review and approval and reporting. Laboratories in various industries may require additional functionality to meet special workflow requirements. An environmental laboratory for example may require tracking of sample containers, processing of samples in batches with control samples, instrument integration to address the number of components reported, and multiple levels of review. There also may be specific reporting requirements. Fig. 4 illustrates some of the additional functions that may be required to address the needs of laboratories in particular industry segments. The functions illustrated would be over and above the basic laboratory workflow.

6. LIMS Workflow and Sample Lifecycle

6.1 *LIMS Workflow Introduction*—The LIMS workflow model (see Fig. 5) provides a generic representation of the process flow in a typical laboratory. The purpose of the work flow diagram is to elucidate the LIMS functions and interaction

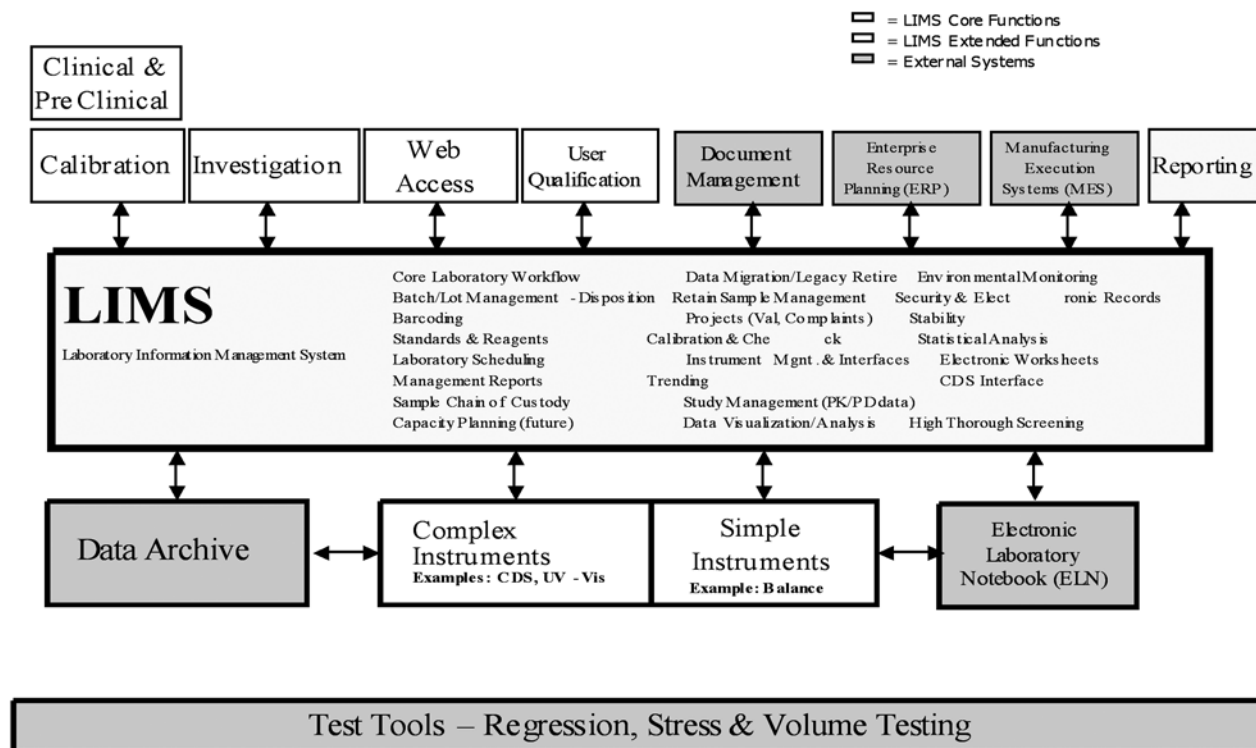


FIG. 1 LIMS Concept Model—High Level

points with typical laboratory work processes (processing of samples, analysis and reporting). Specific laboratory requirements will vary widely from one laboratory to another. The definition and composition of a sample or collection of samples will also be specific to an individual's laboratory, and will change significantly from industry to industry. The individual's own laboratory workflow and sample definition should be defined as part of the LIMS life cycle. Fig. 5 explains the basic LIMS functions and workflow interactions. The numbers in the parentheses in 6.2 refer to specific workflow processes (bubbles) in Fig. 5, which in turn relate to the areas shown in the LIMS concepts models (Figs. 1-4) in Section 5. To provide clear examples of what may be performed in each of the workflow model functions, items from all levels of the LIMS concept model are used. The description in 6.2 does not include every concept model function and is not limited to a particular level.

6.2 Generic LIMS Workflow—Fig. 5 shows both the logical dataflow through the LIMS process in combination with the corresponding physical sample processes. Each box in Fig. 5 is uniquely numbered (cross reference to Fig. 2 and described in detail in subsequent sections).

6.3 LIMS Functional Areas—The following section describes the generic LIMS workflow steps as outlined in Fig. 5. Each step is described with a cross reference to Fig. 2. The index number shown in parentheses refers to both Fig. 2 and the Appendix X1.

6.3.1 Sample Registration (LC-1.1):

6.3.1.1 The initiation of a request for testing/sampling starts the process. Examples of sample requests include manual forms, electronic forms, phone requests, web requests, process-

driven requests, time or calendar-based requests, ad-hoc requests, and LIMS-generated requests. Information obtained from the sample request includes biographical, client, requested test(s), and safety information. Some LIMS implementations allow the laboratory to pre-log or post-log samples or the client to pre-log samples through a web portal.

6.3.1.2 The LIMS must first be properly configured and the relatively fixed information about personnel, customers, tests, reports, and the like must be entered into the static tables. After the LIMS is configured, the process begins with a sample registration. The LIMS assigns a unique number to each sample that is registered. The unique number can be a sequential integer or a user-defined sequence. Multiple samples that are submitted together for registration can be logically linked in the LIMS. For example, all samples for a particular customer. The system will normally provide functionality to capture descriptive information about the sample(s) such as who submitted the sample(s), costs, sample description, and what tests are to be performed on the sample. Other information may also be important, such as the priority of the tests, what level of accuracy and precision of testing is needed, what hazards the sample might present to the laboratory personnel, what approximate levels of components are expected, and what should be done with the sample when analysis is complete. Sample registration can precede or follow physical sample collection. The LIMS sample registration function should be a simple, straightforward process with an intuitive and efficient user interface.

6.3.1.3 A confirmation report is often issued to ensure users the system accepted the sample order. LIMS statuses are updated for the sample/order. The system needs to record the

Numbering System

	LC = LIMS Core Functions 1.0 = Operations 2.0 = Support		LE = LIMS Extended Functions
	E = External Systems		LF = LIMS Future Functions

E-1 Document Maintenance - Develop and Maintain ▲ SOPs Service Level Agreement (SLA) and Operational Level Agreement (OLA) ▲ Specifications User Requirements ▲ Test Methods Functional Requirements ▲ Training Materials Technical Specifications		
Configuration Management (LC-2.1) ▲ Gather Source documents ▲ Process change control ▲ Build master data ▲ Test/Verify master data ▲ Move master data to production ▲ Retire master data	Sample Registration (LC-1.1) ▲ Laboratory service request ▲ Register Lot / Batch ▲ Register samples ▲ Print sample labels	System Administration (LC-2.2) ▲ Hardware maintenance ▲ Software maintenance ▲ Data maintenance ▲ Disaster recovery
Resource Management (LE-1) ▲ Analyst training records ▲ Qualification status	Sample Management (LC-1.2) ▲ Receive samples ▲ Distribute samples ▲ Store samples ▲ Sample chain of custody	Compliance Management (LE-2) ▲ Electronic records ▲ Electronic signatures ▲ Privacy compliance ▲ Financial compliance
Equipment Management (LE-3) ▲ Calibration Management ▲ Preventive maintenance ▲ Service / Repair Management	Core Laboratory Work Flow (LC-1.3) ▲ Assign analyst ▲ Assign instrument ▲ Prepare test ▲ Perform Test ▲ Data entry	Batch / Lot Management (LE-2) ▲ Creation ▲ Review ▲ Disposition
Instrument Data Capture (LE-6) ▲ Chromatography Data Systems (CDS) ▲ Direct data capture ▲ File / Parse capture ▲ Raw data file storage ▲ Meta data capture	Result Review (LC-1.4) ▲ Data verification ▲ Peer, supervisor, director, QA ▲ Data validation ▲ ensure that the data	Scheduled Event Management (LE-5) ▲ Environmental ▲ Stability
Systems Integration (E-2) ▲ ELNs / LES ▲ ERP / MRP ▲ MES ▲ SCADA ▲ Document Management ▲ Statistical Analysis ▲ Cross application analysis	Sample Approval (release) (LC-1.5)	Standards & Reagents Management (LE-7)
Validate LIMS (E-4) ▲ Qualification ▲ Verification ▲ Validation	Reporting (LC-1.6) ▲ Core Laboratory Workflow ▲ Laboratory Metrics ▲ Ad hoc reports ▲ Certificates of Analysis ▲ Management reports ▲ Regulatory reports	Inventory Management (LE-8) ▲ Controlled substances ▲ Reagents ▲ Stability
		Investigation Management (LE-9)
		Control Chart Trending (SQC) (LE-10)
		LIMS Data Archive (LC-2.3)
		Capacity Planning & Laboratory Scheduling (LF-1)

FIG. 2 LIMS Concept Model—Mid Level—Requirements Structure

fact that an order was made (for keeping operational metrics) and when it was made so the system can track the time intervals for the remaining steps of the process. This will also allow laboratory management to determine turnaround time, sample status and various overdue conditions.

6.3.2 Sample Management (LC-1.2):

6.3.2.1 Sample Collection (LC-1.2.1)—Sample collection may be a manual, automated, or robotic process. The sample collection can be assisted by the LIMS (post sample registra-

tion) in some environments by printing collection lists and generating labels (bar code) for the sample containers. Sample collection can precede sample registration or follow sample registration; the actual order will vary from laboratory to laboratory. LIMS statuses can be updated (post sample registration) during the sample collection step. The LIMS can provide information on how to collect samples, specific sample plans, container and preservation requirements, safety (Material Safety Data Sheets (MSDS)) information, sample storage

LIMS LIFE CYCLE PHASES

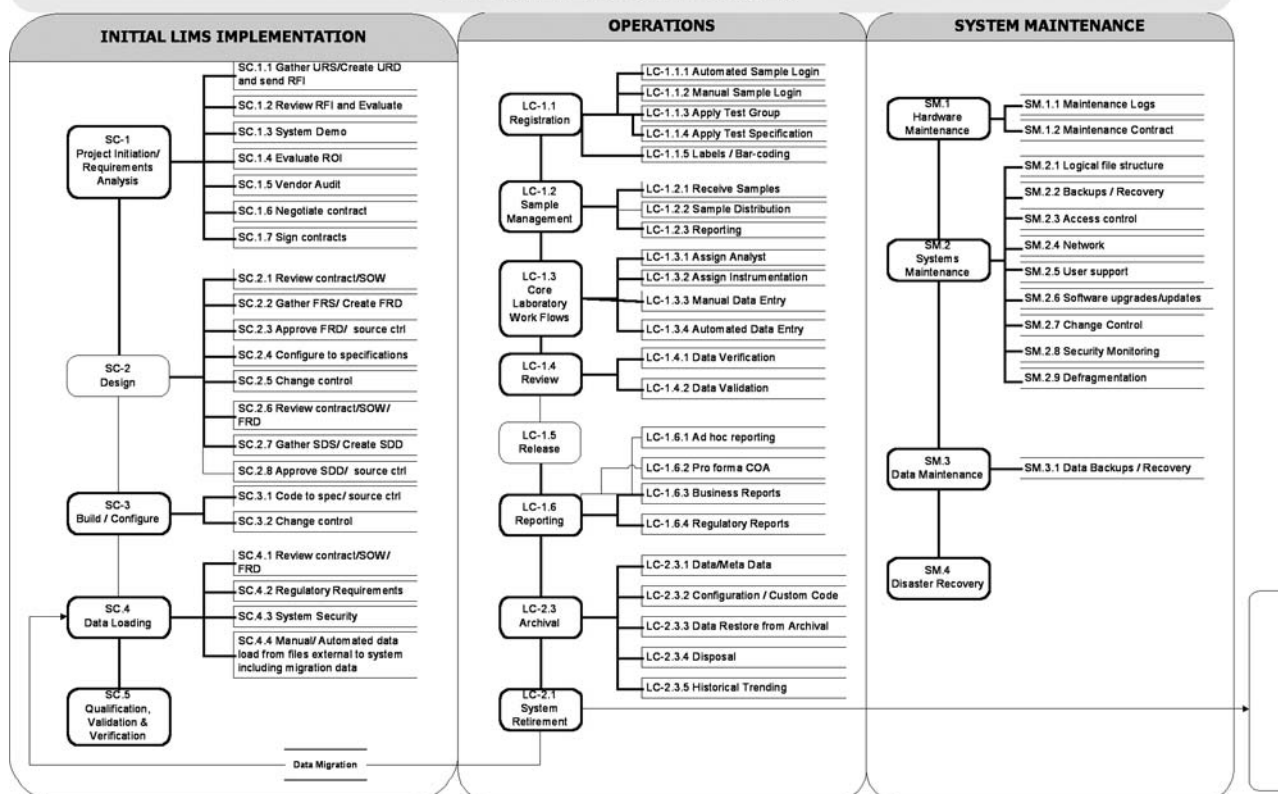


FIG. 3 LIMS Life Cycle Phases

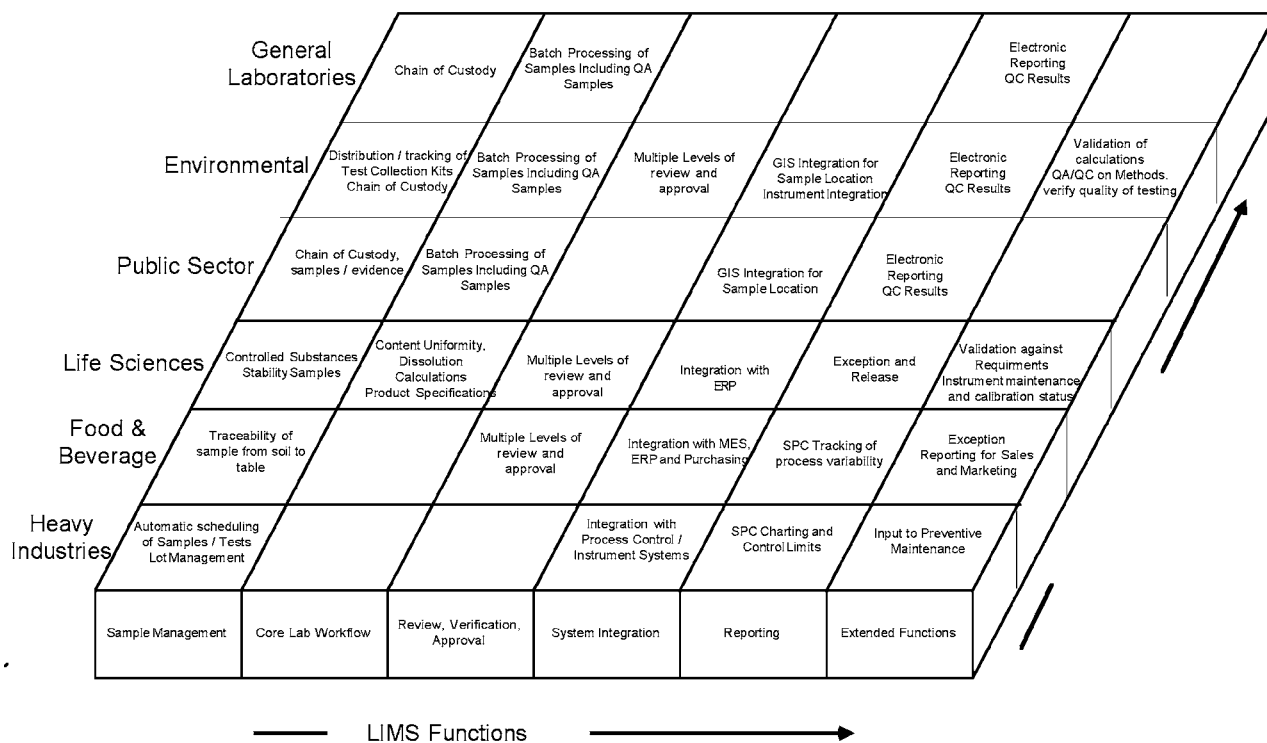


FIG. 4 LIMS Needs by Industry Segment

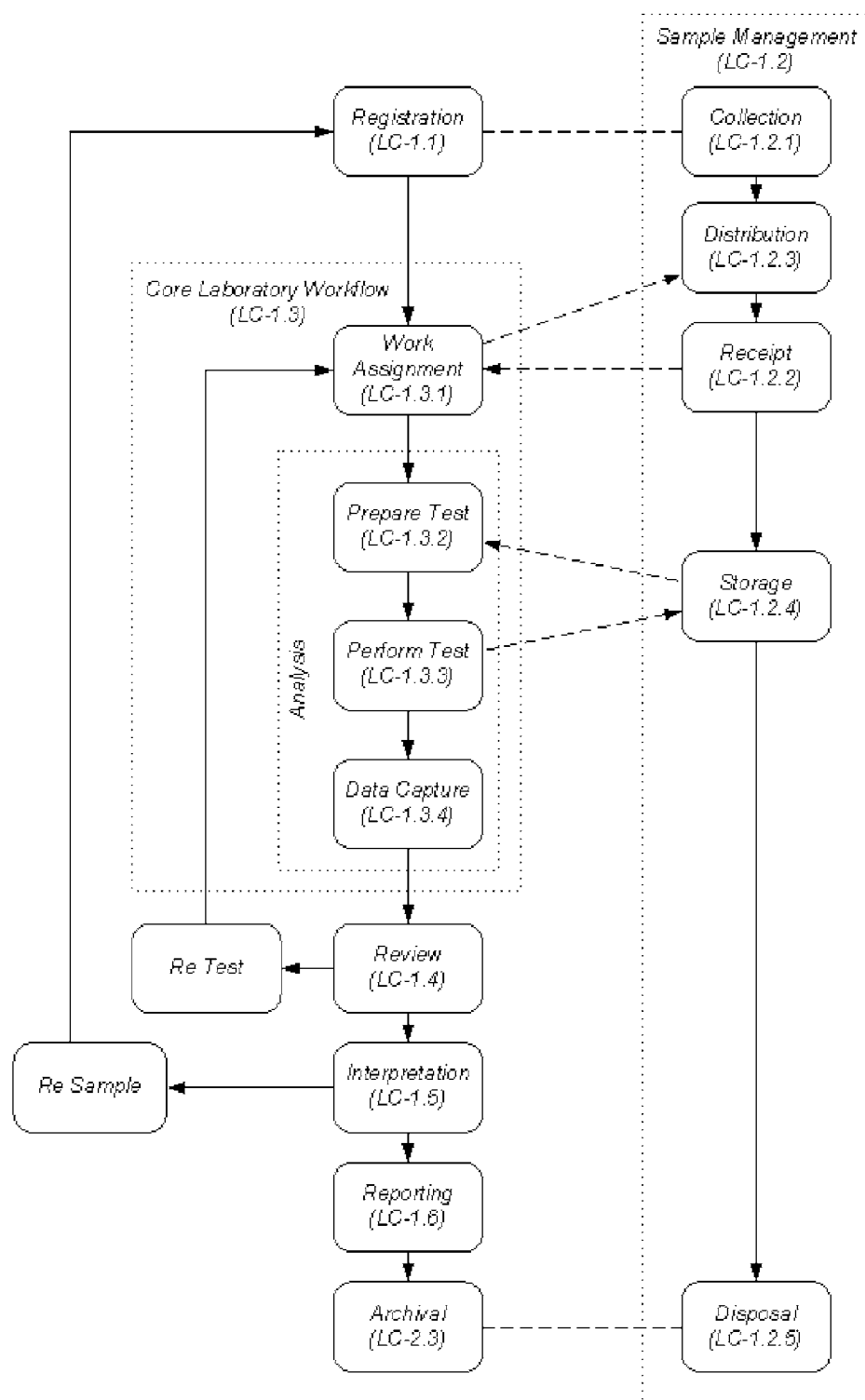


FIG. 5 Generic LIMS Workflow

requirements, and sample routing information. Chain of custody for the sample can be tracked by the LIMS, although this may not supplant legal chain of custody requirements.

6.3.2.2 Sample Receipt (LC-1.2.2)—The physical receipt of samples in the lab is recorded in the system. The receipt process includes the important LIMS functions of sample

registration (if samples were not registered in LIMS prior to sample collection), initial sample checking and labeling. Sample orders or groups of samples may be checked against customer or project sampling requirements. Additional information such as the number of samples received and the arrival time may be recorded and the status of samples may be updated

for the sample/order from logged to received. Where collection lists are used, a missed sample report may be used to indicate those samples that could not be obtained for whatever reason. The customer may be issued a receipt to confirm delivery of the sample.

6.3.2.3 It may be necessary to divide the sample into sub-samples or aliquots for simultaneous analysis at different workstations or laboratories. The LIMS specifies the aliquot requirements for a sample based on the tests to be performed on it. Sample problems may be noted and recorded at this point. There may be insufficient sample for the test to be performed or to prepare the individual aliquots needed for multiple tests. Other problems with the samples may be observed and recorded such as a unexpected color or physical state. Some samples may require a preliminary treatment such as addition of a preservative. If so, instructions can be presented to a technician to ensure correct performance of this step. The LIMS should be flexible enough to allow these checks to be performed and documented during the sample registration process or by technician prior to sample preparation or testing. LIMS should also assign a unique laboratory identifier for each sample received or each aliquot generated and print an appropriate label for each.

6.3.2.4 *Distribute Samples (LC-1.2.3)*—The distribution process includes important LIMS functions of work list, sample routing and custody. Chain-of-custody may be required to provide documented evidence of control and traceability of custody of sample containers and their contents. Examples of situations where chain-of-custody requirements may be encountered include handling of controlled substances, pieces of evidence (forensic) supporting legal court cases, or radioactive materials. The LIMS should provide a listing of all the tests that must be performed and may even provide details to the technician regarding aliquots required for each test, the amount of material required, and where samples are to be sent. The sample or any aliquot may be sent to a workstation in the laboratory or off site to a remote facility for analysis. The date and time of sample distribution is important since it designates when the sample becomes available to the various laboratory workstations for analysis. Sample status may be updated to indicate samples are available for analysis at this time as well.

6.3.2.5 *Store/Retrieve Samples (LC-1.2.4)*—Samples or any aliquots that might have been prepared can be retained in fixed storage rooms/locations while awaiting analysis. LIMS statuses are updated for the sample/order. Inventories can be maintained for reference samples, laboratories reagents, standards, QC samples, time-based samples (shelf life stability), in addition to normal samples.

6.3.2.6 *Dispose of Samples (LC-1.2.5)*—The proper documentation of sample disposal following analysis is an increasing concern. The LIMS can be used to track final sample disposition and waste removal.

6.3.3 *Work Assignment (LC-1.3.1)*:

6.3.3.1 *Schedule Work (LC-1.3.1)*—The LIMS may be configured to automatically group samples into runs or sequences and to schedule work (tests) for each sample/order or it may be configured to allow authorized users to perform these functions manually. The LIMS may be configured to add laboratory

standards, control samples, and QC samples to the scheduled work flow or it may be configured to allow authorized users to manually insert standards and control samples.

6.3.4 *Analysis*—Analysis contains multiple subjects. Subjects addressed in Analysis include sample preparation, measurement, QC samples, and data capture. The analysis activity will vary from laboratory to laboratory. Fig. 5 also shows a re-test and re-sample loop. A more detailed discussion of these topics follows:

6.3.4.1 *Prepare Test (LC-1.3.2)*—Most samples need some preparation before analysis. In addition, some samples may require that preliminary processing steps be performed prior to creating aliquots of the sample for preparation. The LIMS may be configured to provide sample preparation directions for these preliminary processing and sample preparation steps. The LIMS may also be configured to automatically insert into a sequence of unknown samples, the standards and blanks needed to calibrate or verify operation of the preparation method. The LIMS may be configured to compute experimental factors (dilution factor or actual weight) from this data. Alternatively, these types of parameters may be calculated outside of LIMS and manually entered by the technician. All samples, standards and QC samples that undergo these preliminary processing or preparation steps, or both, need to be uniquely identified. QC samples can include spikes, spike duplicates, blanks, and sample duplicates. How LIMS will assist in the proper tracking of these QC samples needs to be carefully examined. The LIMS should maintain links between all such QC samples and their associated unknown samples. The QC sample associations can be determined by the technician, who manually groups the samples in LIMS, or the LIMS may be configured to automatically batch unknown and control samples standards based on some criteria such as the method or instrumentation to be used. Any irregularities or exceptions encountered during preparation may be entered as preparation notes that can be displayed on printed reports or on screen and may help explain any unusual results obtained later in the process. LIMS statuses are updated for the sample/order to indicate that sample preparation has been completed. These associations can be determined by the technician who informs the LIMS, or by the LIMS which tells the technician. They can be tagged on to the reports and may help explain any unusual results. LIMS statuses are updated for the sample/order. After sample preparation, the analyst may use functionality available in the LIMS to put the prepared samples and associated standards and QC samples into a specific sequence in which they need to be measured. Alternatively the LIMS may be configured to automatically order samples and control standards into a sequence based upon the method and instrumentation to be used. Additional standards and QC samples may need to be added and tracked by the LIMS at this point. These additional standards and QC samples need to be uniquely identified in LIMS and linked properly to their associated regular samples.

6.3.4.2 *Perform Test (LC 1.3.3)*—Certain supporting meta-data should be collected as part of the measurement process. This may include instrument settings, any additional standards,

QC samples and blanks used, and any irregularities, difficulties, and unusual behavior observed. This information helps document the procedures used, and may help explain unusual results. Test results/determinations are the main output of the measurement process. Intermediate and final test results for the samples, standards and their associated QC samples may be reported out in hard copy or electronic formats, or both. In addition, the measurement process may produce values for additional internal blanks, standards, and instrument self-checks. The concepts of what is raw data and what needs to be retained for legal evidence may be defined differently for each client or agency involved.

6.3.4.3 In cases where instruments are bi-directionally interfaced to LIMS, the sequence of unknown samples and control standards may be transferred to the instrument to streamline instrument setup prior to analysis. The sequence may include information such as sample ID, sample weight, dilution factor or other pertinent information. The sequence information may be saved in a formatted electronic file created by LIMS and loaded into the instrument software by way of a file import function. Alternatively, sequences may be printed onto worksheets with bar codes and the sample sequence quickly loaded into the instrument software by scanning bar codes from the printed worksheet or by scanning bar codes on prepared sample containers. Where no interface is present, sequences may need to be manually entered into an instrument system by an analyst or a printed worksheet may be used to manually record test results.

6.3.4.4 *Data Capture (LC-1.3.4)*—The results of the measurement must be entered into the LIMS. The amount of the supporting measurement data to be transferred to the LIMS should be carefully evaluated. As clients request that more of this supporting measurement data be reported, the LIMS should have the capability to acquire, store and report more of this supporting data. It may be entered by way of electronic interfaces or, in low volume applications, typed in by technicians. When a test result/determination is entered, the statuses of the sample/order and result determination are updated. The management functions record the fact and times that results were captured so that they can keep statistics of work accomplished and track the progress of each test order. Audit trails record biographical information about each LIMS transaction. Direct instrument integration with the LIMS is critical to fully realizing the business benefits of the LIMS implementation. Instrument integration takes place within box LC-1.3.4.

6.3.5 Test Review (LC-1.4):

6.3.5.1 *Test Result Review and Interpretation*—A laboratory may require that each test result undergo one or more levels of documented review and interpretation. The LIMS should be designed to document at least two levels of review. The original sample result would normally be reviewed and interpreted by the primary technician. This review would be documented in the LIMS. A laboratory may require that results be reviewed by a second qualified person (this is industry specific and dependent on regulatory requirements) to ensure that the results were properly entered and interpreted. To help in this process, the LIMS may show the results for standards and blanks in addition to sample results. The technician can

judge whether the method was in control. The LIMS can show summaries of work done for review. Unusual or out-of-range results can be flagged for more careful scrutiny. If normal values are known for the substance being tested, they can be displayed. Results outside of normal can be highlighted or displayed separately for closer review. The LIMS can enforce laboratory SOPs that require the reviewer to be a different person from the tester. Corrections or changes to LIMS data can be made during the verification step. Changes to LIMS results should be audit trailed and a reason given for why a correction has been made. Audit trails should contain original data and all changes to the result record including date/time of change, who made the change, and the reason for the change. In some cases, the LIMS can provide a review by exception paradigm where results meeting specifications or within acceptable limits are passed automatically through test review and only unexpected test results are queued for manual review. Results can be approved, changing the result status. A test (one or more determinations) can be scheduled for re-test, or if that is not possible, the result can simply be marked as invalid. Electronic signatures may be used to confirm changes in status to the LIMS records if the regulations or guidelines require this. Management functions need to know when results are verified—another milestone in the progress of a test/sample/order. LIMS statuses are updated for the sample/order. Not all LIMS implementations require audit trails. The LIMS implementer needs to determine whether audit trails are important, what information should be audited, and whether reasons for changes should be recorded.

6.3.5.2 *Re-Test Loop*—Retests can be initiated at multiple points in the LIMS work flow. Fig. 5 shows possible re-test paths. A re-test is defined as one or more additional determinations on the original sample/order container. These retests would normally be ordered if a given test was suspected to fail for reasons that may include failed quality control parameters, instrument malfunction or technical judgment. The LIMS should document each retest along with an appropriate justification.

6.3.5.3 *Re-Sample Loop*—Re-samples can be initiated at multiple points in the LIMS work flow. Fig. 5 shows possible re-sample paths. A re-sample is defined as one or more additional samples. The LIMS needs to establish forward and backward links to samples that are added by way of the re-sample loop. These resamples would normally be ordered if a given sample was suspected to fail for reasons that may include where insufficient sample was available for a retest, technician judgment that the original sample was not appropriate for the test performed, or to confirm a test failure.

6.3.6 Sample Approval (LC-1.5):

6.3.6.1 A laboratory may require that each sample undergo a documented approval process. The LIMS should be designed to document this process. The laboratory exists to generate information for the parent/client organization. Some of the LIMS today are configured to better assist that ultimate purpose. They may organize and configure results to make interpretation and decision-making easier. This can be done by combining results from many samples, adding additional

non-laboratory-generated information to the reports or Certificates of Analysis, and including generic information related to the test or activity that caused the samples to be analyzed in the first place. Analysis is frequently done to confirm quality or properties of a material. In this case, material specifications can be entered into the LIMS so that results can be checked against acceptable values. Electronic Signatures may be used to confirm results or authorize changes in status to the LIMS records. Sometimes statistical routines can be used with collections of results to determine trends and make other conclusions. Spectral libraries can be used to identify materials. Artificial intelligence is used in some cases to help understand the results. LIMS statuses are updated for the sample/order.

6.3.6.2 The output of the approval segment can be reports, Certificates of Analysis, decisions based on predefined criteria, or direct process control actions.

NOTE 2—In many cases the interpretation function will coincide with the reporting process. In many LIMS data are interpreted in a reported format either in electronic or paper forms.

6.3.7 *Reporting (LC-1.6)*—Once test results are verified, both external and internal reports can be generated.

6.3.7.1 Test results, along with QC data, can be reported to the customer. This can take a variety of forms, including hardcopy reports, electronic data deliverables, and web-based systems. The report generators within a given LIMS need to be flexible to accommodate the different reporting needs of individual clients. The LIMS should come with preprogrammed formats for the most common hardcopy and electronic reporting formats. Conventional Certificates of Analysis are commonly required to be reported as an example of a hardcopy report. Many clients are now relying on the use of various electronic formats that support the transfer of data from the LIMS database to the client's database using electronic transfer formats such as XML.

6.3.7.2 Reports can include summaries for internal laboratory use by management. Management functions are told when the reports are issued, because this marks the end of the turn-around time. LIMS statuses are updated for the sample/order. By collecting statistics and time-stamps at various points in the process, reports can be prepared for laboratory managers. Number of samples processed at each workstation by shift, day of week, and hour of day can be prepared. This can help identify peak demands, roadblocks, and other problems. It provides good documentation to justify new instruments or personnel. Turnaround times document the laboratory's responsiveness to customer needs. Overdue results and work remaining in the system help managers to determine how well the laboratory is responding to current demands. Personnel time accounting can be tracked by the time each sample is at each workstation. This can be used to bill by project, and to monitor personnel performance. The number of tests done can be used to estimate the consumption of reagents and supplies. Instrument calibration and maintenance records can be maintained and reported by the LIMS.

6.3.7.3 Quality control reports can also be prepared for internal laboratory use. Statistical reports can be generated to evaluate the performance of a given method within the

laboratory. Control charts can be generated based on analysis of specific QC samples. The QC ranges as derived from these control charts can be used to set the limits as used by the laboratory to evaluate their internal processes or used to compare how the laboratory's performance compares to the published performance of a given method. Some clients require that these statistical ranges also be reported to them.

6.4 *LIMS Data Model:*

6.4.1 *Types of Data*—The database technology employed by LIMS varies with each vendor and implementer. The LIMS database tables are divided into two broad areas: (1) LIMS static database tables where descriptive information is defined (for example, profiles, tests, calculations, specifications, and related information (commonly found in "look up/reference/dictionary" tables)) and (2) dynamic tables where sample and result/determination information is stored as samples are logged and results are entered. The terms static and dynamic represent general characterization of LIMS database tables reflecting the frequency of changes to the data. The LIMS user needs to closely study how the current laboratory information organization and work flow match the two database areas (static and dynamic). The time required to implement a LIMS is dependent on tools and structure of the static database tables.

6.4.2 *Statuses*—LIMS are capable of maintaining information on the status of samples, individual test/determinations, comparison of results to specifications, verification of results, approval of samples/orders, and much more. Status information is updated as each LIMS transaction takes place. The functions/work flows all have an impact on LIMS status information. Examples of sample/order statuses include: unavailable, available, and received in the lab, testing in progress, suspended, complete, approved, and rejected. Examples of test/determination statuses include: available, complete, approved, out of specification.

6.4.3 *Data Load and Migration*—A LIMS is capable of maintaining information for a broad range of business and laboratory data required for the effective operation of the laboratory. LIMS contains data that not only reflects the current operation state of the laboratory but also historical information on past performance and events. Where LIMS is introduced into an environment where a LIMS system has not previously been deployed, much of the information typically managed in the LIMS will exist in the form of paper documentation or disparate electronic sources such as documents, spreadsheets or specialized databases. When implementing the LIMS this data need to be loaded into the new LIMS system to provide the base configuration. Static data are always loaded into the LIMS as part of the LIMS deployment lifecycle. The decision on how to deal with historical dynamic data should be evaluated on the basis of risk. Appropriate strategies for dealing with this data include migration, preservation and archival. Where an existing LIMS is being replaced with a new solution, it may be possible to migrate data from the source LIMS to the new target LIMS deployment. Migration of data needs to be carefully analyzed and planned. The plan should include processes to verify that the data are successfully migrated to the new database.

7. LIMS Infrastructure, Integration and Interfaces

NOTE 3—This section describes in more technical detail the industry standards for the infrastructure necessary for LIMS in general and interfacing and integrating LIMS to other systems such as ERP and laboratory instruments. Reference Figs. 1 and 2, block E-2.

NOTE 4—Fig. 6 describes the Conceptual Laboratory Informatics and Corporate Computing Architecture that is typically found in larger LIMS implementations.

NOTE 5—The use of wireless infrastructure technology is critical to achieving business benefits in laboratories where mobility is important.

NOTE 6—Fig. 7 describes the ISA-95 Conceptual Information Technology Topology model. The standard provides a formal model for exchanged data between business systems and manufacturing systems. The models provide a definition of: (1) manufacturing operations management, (2) the activities on the shop floor that take production schedules and perform the actual work required to manufacture products and provide visibility of production. The LIMS is shown as a Level 3—Operations Management level in Fig. 7. The LIMS typically integrates with lower levels (ties to laboratory instruments) in Levels 1 and 2, as well as ties to higher Level 4 systems like ERP.

7.1 Hardware/Platforms—The criteria for LIMS selection should be driven by the software function. Hardware should be a second priority behind overall software functionality. Computer hardware technology and price-performance ratios used to support LIMS are changing rapidly. The LIMS implementer should start with vendor guidelines for sizing to ensure computer hardware will match projected needs. The implementer should follow up vendor hardware sizing recommendations with site visits and performance testing on pilot systems in-house (Vendors sometimes under-specify the hard-

ware to keep initial costs low in order to capture your business). Hardware sizing is dependent on many factors. Important factors include: (1) number of concurrent users, (2) number of records (sample and determinations) per year, (3) number of records to be maintained on-line, (4) archive requirements, (5) type of reporting required and, (6) external loads on the system from non LIMS applications. Hardware sizing includes central processing unit (CPU), clock speeds, bus data width, memory, disk capacity, disk I/O, archive media capacity, and network communication rates. The first-time LIMS users should be aware that LIMS (database) transactions often place demanding loads on computer hardware. Reports that are required to summarize data for large data sets can take minutes to hours to run. The user needs to plan the implementation goals, schedules, and resources. The LIMS may take 6 to 24 plus months to implement in a large laboratory. The laboratory may be better off buying a small processor for implementation and upgrading to a faster platform near the end of the implementation (when hardware prices should be cheaper). Plan for growth 1 to 3 years ahead. Business cycles do not always result in laboratory expansion. Consider whether the LIMS you evaluate can be scaled back to a smaller, simpler system as well as to a larger, more complex one. Database software vendors often have significant surcharges for scaling licenses back to smaller systems, and hardware and software discounts may be heavily affected by downsizing. Portability of software between hardware systems is important if you expect to change hardware platforms over the life of the LIMS.

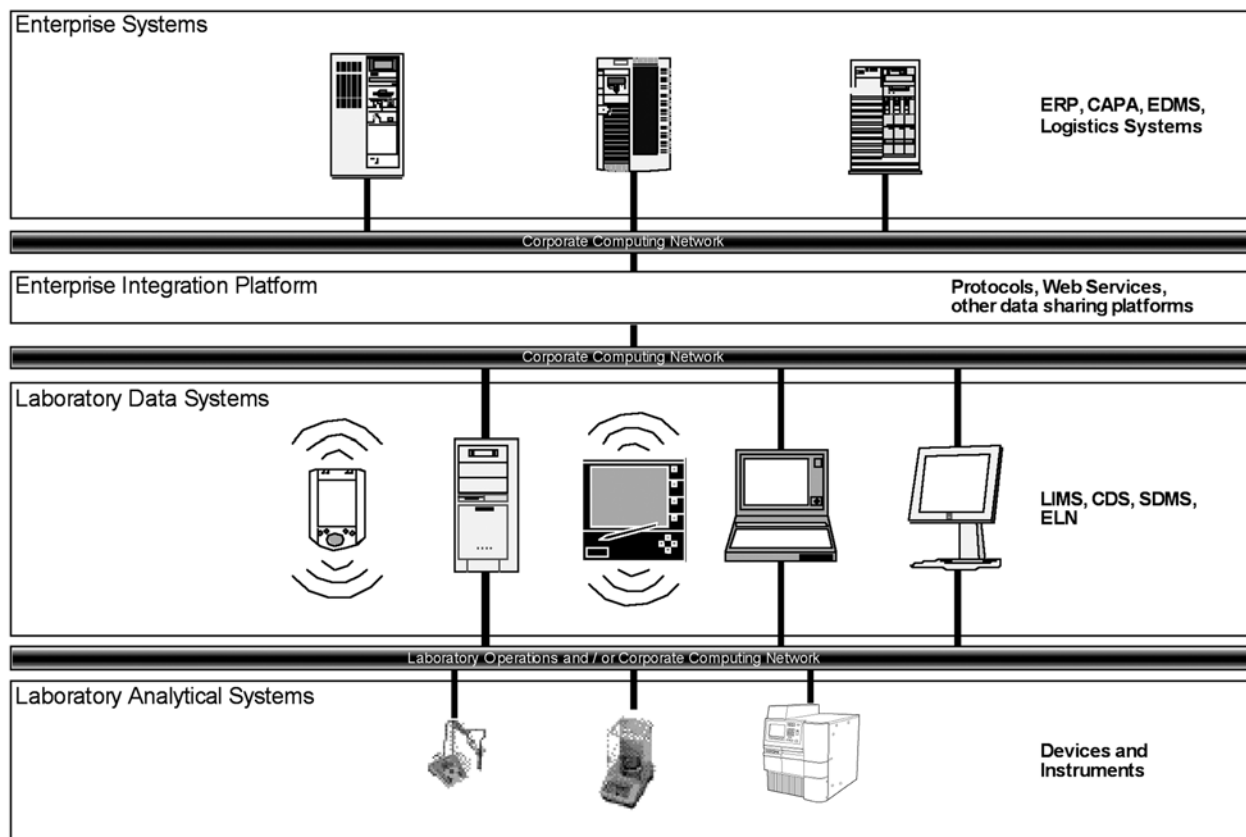


FIG. 6 Conceptual Laboratory Informatics and Corporate Computing Architecture

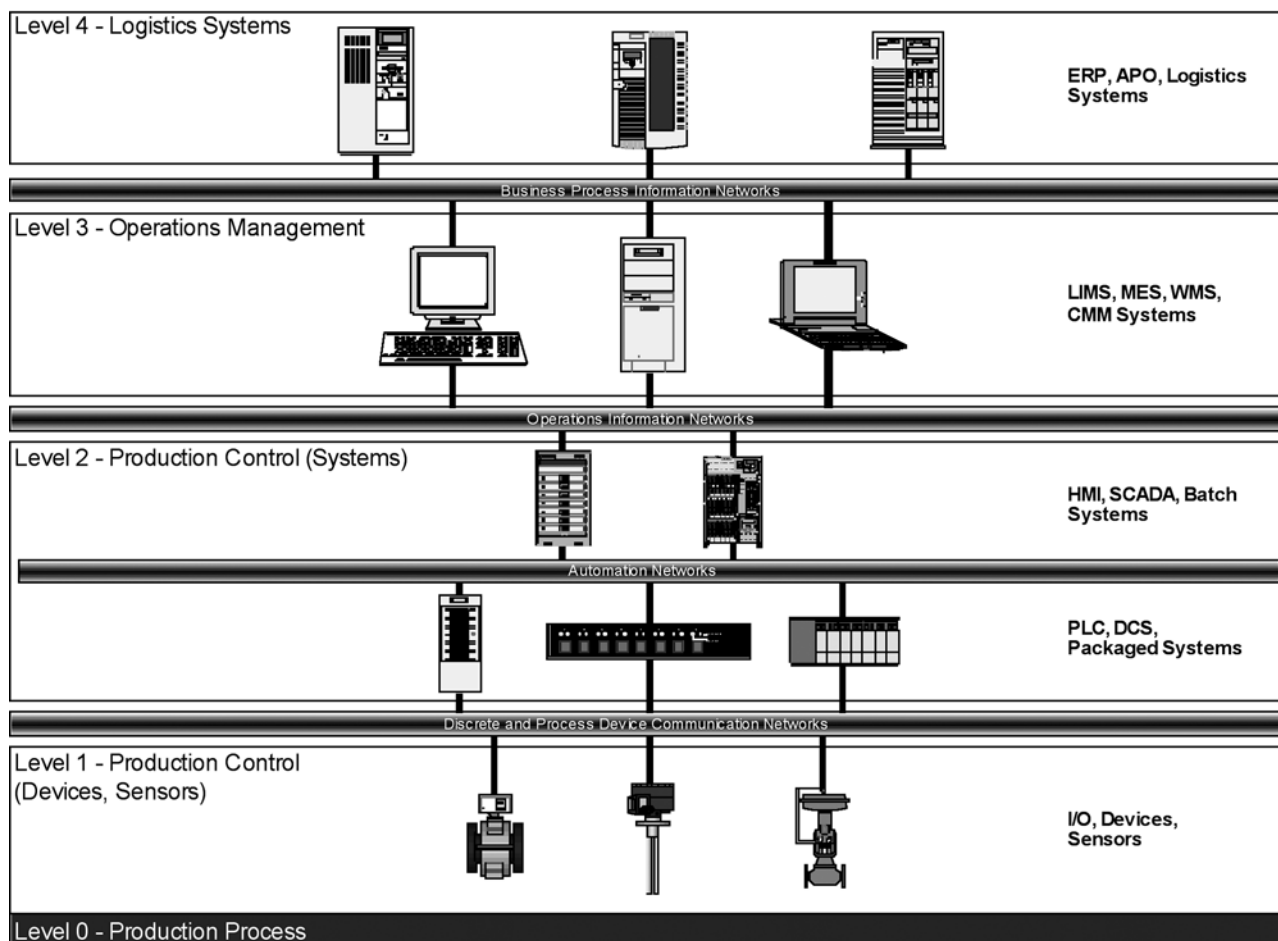


FIG. 7 ISA-95 Conceptual Information Technology Topology

The ability to transfer data between different computer systems is vital in a heterogeneous computing environment. Select a hardware system that can be scaled up (CPU speed and storage capacity) to meet changing requirements.

7.2 Database Recommendations:

7.2.1 General Requirements—The database component of the LIMS system is often the architecture that requires the greatest level of flexibility due to the ever-increasing demands of information exchange between the laboratory and the enterprise. The LIMS system should be based on a commercial database management system or database toolbox that is reliable, effective and supported external to your LIMS vendor (this is especially true if there is a chance that you may change your LIMS in the future). Commercial relational database management systems can be organized, configured and tuned to meet a wide variety of usage and performance scenarios. Proprietary LIMS database management systems are not widely utilized but may be required to meet specific performance requirements. Proprietary LIMS database management systems carry risks of tying laboratory data to a specific vendor, which may add cost to any migration to a new system in the future.

7.2.2 Key Considerations—When evaluating the database platforms, the following features should be considered.

7.2.2.1 Standardization—The database should allow an application or the database administration personnel to interact

with the database based on industry best practices and common standards for relational database systems, such as the use of structured query language (SQL) for database queries. Ideally the database platform should also conform to the database platforms and standards currently in place in your enterprise.

7.2.2.2 Core Design Flexibility—The database design provided by the LIMS vendor should be well-documented and sufficiently flexible to accommodate common LIMS administration and configuration tasks, such as the maintenance of users, modification of workflows, reference information in lookup tables and the addition of user-defined fields. The design should preserve the referential integrity of information in the database, that is, addition, deletion or updating of data in one area should be dependent upon the impact of data that it refers to. All data types used in the workflow of the laboratory should be accommodated by the database, including a full range of numeric, date/time and text/memo data types. Other data types required may include more advanced functionality such as the ability for data types to support the storage of images, multimedia and files.

7.2.2.3 Extended Design Flexibility—The database should support the ability to modify the database structure as needed, including adding/modifying fields, indexes, relationships, and tables. However, careful consideration should be given to the impact on LIMS functionality prior to making these changes.

7.2.2.4 Data Replication—The function of data replication is critical to protecting and maintaining information in the LIMS. Laboratories that utilize electronic records (with no paper backup) must provide added protection of information captured within the LIMS. Typical snap shot or incremental backup processes run nightly but provide limited protection between backup intervals. Data replication tools within the database layer (and some times between the storage layers (storage area networks (SANs)) provide added protection against data lost by replicating all transactions between data centers or remote servers.

7.2.2.5 Multiple Environments: (Development, Quality (Test) and Production Platforms)—The database platform should support the ability for multiple environments (copies) of the LIMS database and migration tools to move objects between environments. Typical implementations include a development environment for code/configuration development, a quality environment for formal testing and master data building and a production environment for production information. The database environments include application development components for data administration, application customization and integration. These tools often include the capability to develop stored procedures, views (stored queries) and functions for access by scheduled processes and other applications or application modules.

7.2.2.6 Maintenance—The database platform should allow the database administrator to fine-tune the performance and security of the database with functions such as indexing, table space management and process schedulers.

7.2.2.7 Personnel—If some or all of the LIMS system will be managed and maintained in-house or by parties other than the LIMS vendor, consideration should be given to the availability of technical personnel with skill sets in the technologies utilized by the database platform.

7.2.2.8 Integration—The database platform should allow the importation and exporting of data in industry standard formats (for example, XML and ASCII text) and provide toolkits or third party middleware and/or messaging support for integration to other applications and information systems.

7.2.2.9 Backup and Recovery—The database platform should support industry best practices for backup and recovery in an efficient and expedient manner. This function should either be included in the database toolkit or supported comprehensively by third party tools.

7.3 LIMS Application Platform:

7.3.1 General Requirements—LIMS systems are developed on a wide variety of application platforms, and many standards and programming languages are used which provide adequate features and functions. It is therefore important that the LIMS vendor provide detailed documentation on the technical capabilities and design of the application architecture. The documentation should allow you to clearly evaluate the flexibility of the application relative to the capabilities needed by your laboratory. An overriding requirement for many organizations is to evaluate the application architecture in the context of its ability to be configured, customized and integrated with other systems (see 6.3.3).

7.3.2 Key Considerations—When evaluating the application platform, the following capabilities and features should be considered:

7.3.2.1 Modularity—The application's functionality at the architecture level should be clearly separated into logical modules with clearly defined standard interfaces between modules. These modules can be defined based upon feature groupings, layering of the application architecture (for example, presentation, business logical, data interfaces) or both. Often this is accomplished through the use of object-oriented design techniques in conjunction with other industry-standard best practice approaches to application design. Design modularity can minimize unexpected consequences of application changes through configuration, customization or integration by isolating the areas of the application affected and facilitating application testing.

7.3.2.2 Configuration Tools—The LIMS application should provide an extensive administrative interface so that end users can configure the application as much as possible without programming or direct database intervention. The user interface should also be evaluated for the ability to add, remove and change design and form elements on the screen to create productive forms and workflows with minimal programming. Ideally the LIMS application should allow for customization that incorporates embedding of content from external systems such as embedded multimedia(chromatograms, gel plate, short training videos, and so forth).

7.3.2.3 LIMS Programming Languages—The LIMS should provide programming to customize LIMS applications or build calculations within the LIMS application. The LIMS programming tools can be very powerful and allow your laboratory or IS staff to customize the LIMS application to meet changing requirements. Caution is recommended when customizing a vendor-supplied LIMS to ensure that your system is compatible with future vendor software upgrades. See section below on customizing LIMS.

7.3.2.4 LIMS Application Data Structure Match with Business Workflow—The LIMS application and its underlying technology should closely match your organization's laboratory workflow requirements and information structure. The application and database architecture of a LIMS system should be assessed on its flexibility to configure, customize and integrate the system to fit the organization's needs. Note: The overall fit of a LIMS to laboratory operations is generally more important than specific advanced technology.

7.3.2.5 Performance Design and Data Integrity—The application architecture should be designed to use the operating system and hardware platform specified as efficiently as possible. This includes evaluation of concurrent usage, peak usage and the ability for individual end users to multi-task (that is, open multiple screens or application functions in the same user session) without losing work. The use of test automation tools and building a performance-testing model of the system early in the process provides significant benefits. The automation tools can be employed to monitor changes in performance, perform regression testing when changes in software are applied and tune the system as it is developed and even during the operate and maintain phases of the LIMS life cycle.

7.3.2.6 Database Interoperability—If the technology standards of the organization change frequently, the LIMS application should be evaluated for the ability of the application architecture to support widely used database platforms.

7.3.2.7 Personnel—If some or all of the LIMS system will be managed and maintained in-house or by parties other than the LIMS vendor, consideration should be given to the availability of technical personnel with skill sets in the technologies utilized by the application. Knowledge of supported operating system, programming languages and design techniques used to customize and integrate with the application are important.

7.3.2.8 Application Programming Interfaces (API)—If customization or integration of the LIMS system is anticipated, the LIMS vendor should be able to provide a well-documented API for interfacing with the LIMS application, with a clear model for interfacing with the application's functions at a granular level.

7.3.2.9 Integration Standards—The LIMS system should provide a means to integrate and exchange data based on common methods, such as the XML-based SOAP protocol and/or based on the integration methods supported by applications that commonly integrate with the LIMS system, such as document management systems and ERP systems.

7.4 Customization and Integration of LIMS:

7.4.1 Comparisons of Configuration, Extensibility and Customization—Customization is often used as a blanket term referring to three distinct activities which differ technically. They are:

7.4.1.1 Configuration—Use of the system management features provided with the LIMS package to target LIMS features to the specific usage activities and workflow of the organization. Configuration is often limited to changing only the reference data or predefined settings in the LIMS application. Core laboratory requirements should be configurable in the LIMS system.

7.4.1.2 Customization and Extensibility—Changes to the LIMS product, which result in the enhancement or addition of LIMS features, required by the organization's specific needs. Customization is chosen when the current and anticipated future LIMS feature set, or integration with another software application, will not accommodate the organization's specific requirements (all other business and budget considerations notwithstanding). Customization often requires changes to the database or application design of the LIMS product or the addition of custom programs.

7.4.1.3 Integration—The ability of the LIMS product to exchange information with other software applications. This may involve either the exchange of data with other applications and/or the exchange of application logic and functionality with other applications. Integration allows for LIMS to leverage the features of other applications without adding custom features to LIMS itself. Integrating LIMS may require design changes to the database or application architecture of the LIMS product, but should ideally be minimal if the LIMS product is flexible for integration.

7.4.2 Common LIMS Customization and Integration Activities—Some examples of where LIMS is customized or integrated with other systems include:

7.4.2.1 Workflow—Customizing LIMS to accommodate more advanced workflows for productivity, compliance or data enrichment purposes. This may also include integration to:

- (1) Document management systems (standard operating procedures, chain of custody management, and so forth),
- (2) Training and e-Learning Systems, and
- (3) Enterprise Resource Planning Systems.

7.4.2.2 Chemical Information Systems:

(1) Reporting and Business Intelligence—Customizing LIMS reports and/or adding ad hoc reporting capabilities. This may also include customization of data exports or database views to external reporting systems or data warehouses/marts, or integration to information dashboards and portals.

(2) Data Capture—Customizing the LIMS user interface for more efficient data capture, integration to electronic notebooks and dedicated data capture applications, or integration to instrumentation.

(3) Security—Customizing LIMS to provide more granular security permissions for system features, incorporating compliant E-signatures, integrating with enterprise directory or authentication systems (LDAP directories, security badge systems, and so forth) for single sign-on.

7.4.3 Business Considerations for Customizing or Integrating LIMS—One of the most critical evaluation criteria in the selection and implementation of any LIMS system is the organization's need for the customization and integration of LIMS and the capabilities of the candidate LIMS system to perform these functions flexibly. Customizing LIMS or integrating LIMS with other systems can profoundly impact production selection, implementation, ongoing management, and total cost of ownership of the LIMS investment. The decision to customize LIMS for a particular business purpose or integrate LIMS with another information system must follow a deliberate process that forms a part of the LIMS System Life Cycle. For a more detailed explanation of the business and management considerations of customization, see sections on:

7.4.3.1 LIMS system life cycle,

7.4.3.2 Developing a business case, and

7.4.3.3 Change control.

7.4.4 Technical Planning for Customizing or Integrating LIMS—Organizations planning to customize or integrate LIMS need to understand how the technical aspects of the effort will impact the implementation and ongoing management of the LIMS system. The actual content and process for analyzing the impact of a customization or integration project will vary by the organization's system life cycle methodology. However, at minimum, any impact analysis and technical plan should include the following activities before technical implementation:

7.4.4.1 Requirements Decomposition—A detailed review of functional requirements with an assessment of the technical activities and resources needed to meet the requirement. This should also include an assessment of the technical risks that may be encountered in implementing the requirement.

7.4.4.2 Architecture Review of Current LIMS—A complete review of the application and database architecture of the LIMS system. The review should include, at minimum, an assessment of:

(1) Programming and database design changes or additions needed to meet the requirements.

(2) Programming and database interfaces provided by the LIMS vendor.

(3) The impact of the change to other LIMS features and functions. Great care is needed to insure that the customization or integration with another system does not alter the expected and validated LIMS functions already in use.

(4) The impact of the change when LIMS or the system that LIMS is integrating with is patched or upgraded. This is a critical factor—poorly designed customizations and interfaces or inflexible LIMS systems can render feature enhancements and even core LIMS functions useless when an upgrade occurs. Even in less extreme circumstances, poor design can lead to unexpected costs and effort to coordinate the maintenance of LIMS and the enhanced components.

(5) The impact of the change and use of the change to the overall performance of the LIMS system. For example, will a scheduled overnight batch exchange of data with another system consume enough system resources to negatively impact other system maintenance and batch activities?

7.4.4.3 Architecture of Integrating System—A complete review of the application and database architecture of the system that will be integrating with the LIMS system. This assessment should follow the same criteria as the architecture review of current LIMS.

7.4.4.4 Personnel—Types and sources of technical personnel needed. Technical personnel used for customization can include the following skill sets:

(1) Business analysis,

(2) Application and database design,

(3) Application and database programming (on the language, RDBMS and software tools supported by the LIMS vendor),

(4) Middleware programming (supporting XML programming, data extraction, transformation and load (ETL), and so forth), and

(5) Database and system administration.

7.4.4.5 Careful consideration should be given to the sources for obtaining the technical personnel needed for the project (LIMS vendor, other product vendor, outsourcing firms, in-house IT staff) and what is the best combination of skill, cost and availability for each phase of the project. (For example, one may use the LIMS vendor or an outsourcing firm for the development of a LIMS customization, but in-house staff for maintenance of that customization).

7.4.4.6 Integration versus Customization—A decision on whether the requirements will be best served by a LIMS customization or integration with another application that already meets the requirement.

7.4.4.7 Prototyping—As with implementation of the core LIMS system, prototyping is a critical step with any customization so that scope of effort, technical approach, and technical risks can be validated and quantified prior to implementation.

7.5 Enterprise Computing Architecture—Figs. 1 and 2 provide an overview of the LIMS computing architecture, the integration points with LIMS and the overall LIMS functions. Figs. 6 and 7 illustrate an example of an integrated enterprise computing architecture that span from enterprise systems down to the bench level laboratory analytical systems (instruments).

7.5.1 Multi-Site Deployments (Globalization (Corporate Multi Site Deployment))—Globalization is the process and environment by which companies conduct business (internal activities and commerce with others) in many countries across the globe. The advent of advanced communications technology such as, the internet and the rapid expansion of trade (among other factors) have greatly distributed scientific activities in general and lab environments in particular across the world. The result has led LIMS users with global operations to demand LIMS vendors provide features that:

7.5.1.1 Configure most or all user interface features with multiple languages.

7.5.1.2 Automatically adjust for differences in time zones and character sets.

7.5.1.3 Streamline distribution of software updates across wide area networks and/or the Internet.

7.5.1.4 Accommodate lab oratory-related regulatory issues across multiple countries and/or jurisdictions.

NOTE 7—Globalization is usually addressed via a combination of configurable and customizable features within the LIMS application.

7.5.2 Configurations for Multi-Site LIMS Deployments—Multi-Site LIMS deployments typically take three forms (see Table 1)—central, regional and distributed.

7.5.3 Content Localization—Content localization provides a user interface that reflects the specific geographic needs of each user generally involves three elements- (language, character sets, and time zones), with each setting usually set in the user's LIMS profile or in the local installation.

7.5.3.1 Language localization is a translation of the language in the user interface for the local user. It is generally a configurable item in global LIMS deployments, usually driven by database language content, configuration files, and/or an XML-based user interface framework.

7.5.3.2 Character sets refer to the set of characters employed to express the selected language in the user interface. The choice of character sets, are particularly important when considering user interface customizations. Unicode character sets are often used to specifically address local language issues without impact to the rendering of the user interface.

7.5.3.3 Time zone refers to the specific local time zone employed by the LIMS system for time stamping and audit trail purposes. It is especially important to be mindful of time zone configurability in a LIMS system when storing, comparing or aggregating data across multiple time zones, such that audit trail can be preserved. Expressing the time as an offset of Greenwich Mean Time (GMT) for each time zone unifies this information (for example, U.S. Eastern Standard Time is GMT–5 hours) is commonly used with in LIMS implementations.

7.5.4 Regulatory and Functional Issues—Globalization issues include specific functional needs across regulatory jurisdictions and other functional issues derived from the local

TABLE 1 Multi-Site LIMS Deployment Strategies

Description	Benefits	Challenges
Central deployments house the entire LIMS application, database and infrastructure in a single data center or group of data centers, with all application functions centrally managed. Often the user interface is delivered via a web browser or thin client terminal (including remote desktop) application to users. Configuration for the needs of each laboratory is managed centrally as well.	Central management consolidates support resources and usually costs. Changes and upgrades to the LIMS are available to users in all locations immediately. Local IT support is greatly minimized since there is no need for local support of the application (beyond training, and so forth). Deployment to new locations is much quicker and easier than with distributed systems. Data are centralized, making it easier to gather and analyze business intelligence on organization-wide activities and trends. Delivers a happy medium between cost savings from economies of scale and delivering the right return on investment to meet user needs. Local IT support is still minimized, since this method generally uses a "thin client" approach. Deployment to new locations is still relatively quick and easy compared to a fully distributed model. Changes and upgrades can be delivered quickly while maintaining flexibility. Data are still more easily aggregated across common business needs.	Application architecture is critical to the flexibility of LIMS for local laboratory needs. The system must be highly configurable to smoothly handle local language and regulatory issues. Customized features must be analyzed to insure that the entire user community benefits (and is not otherwise harmed) by the changes. Infrastructure must be carefully planned for network availability and redundancy, especially since a central deployment must be available 24/7 with global operations.
Regional deployments combine the economies of scale of the central model with the flexibility of the distributed model. Clusters of LIMS users are defined with common geographic, regulatory or usage communities so that LIMS implementations can be centralized into a few installations while still meeting the core application needs of each group.	Content localization is out-of-the-box, since there is an installation for each location which is configured for local needs. Configuration and customization can be performed largely without regard to needs beyond the local deployment. Ideal where each location's needs are clearly separate from one another and/or functional differences by location are clearly delineated. Network availability is generally confined to more manageable local network (LAN) issues versus more complex wide area network issues.	User clusters must be defined carefully, or the result is simply a higher-cost centralized model. User requirements must be clearly aligned across language, regulatory and functional needs so that the LIMS system is properly configured. Customizations and upgrades must be carefully synchronized, both among the region's users and across the organization so regional deployment, redundant systems can actually be deployed in other regions, making disaster recovery economical. that technology and personnel resources can be used efficiently. Careful infrastructure planning must be done to make sure system availability is not affected by regional events. In a regional deployment, redundant systems can actually be deployed in other regions, making disaster recovery economical. Deployment, maintenance, upgrades, infrastructure and support can be very costly since each specific installation must have these resources available. Determining a global strategic direction for aligning LIMS with the organization is difficult, due to a high degree of local variance. Lack of standardization. Common regulatory and management standards are difficult to automate across the organization, and benefits from one deployment are difficult to translate to another location. Data aggregation to analyze and report on global operations is nearly impossible with highly distributed data sources. Disaster recovery plans are difficult and/or costly to implement since each plan and implementation will vary by location.
Distributed deployments typically emphasize flexibility for end users, but usually incur the highest support and deployment costs. Distributed models consist of either stand-alone fat client desktop applications, or a client- server model, with a server installation at each location. Distributed models are generally not used when globalization is a significant issue for LIMS deployment.		

laboratory environment. These issues can either be handled through configuration (for example, FDA 21 CFR Part 11 support) or customization. It is important in these scenarios that customizations are performed with upgrade path, support and impact to users in other locations in mind.

7.5.5 User Security—User security varies greatly by technical implementation across LIMS systems, but in general, LIMS systems should be evaluated to according to the following basic user security issues: (1) Does the security framework provided by the LIMS system provide audit trail and permission control in a comprehensive manner as compared to the needs of the organization? (2) Is the security framework flexible and granular enough to allow control of security at a functional or task level? (3) Can the security framework be conveniently administered, with a single security framework for all modules of the LIMS, and can it be integrated with other enterprise frameworks? LIMS security frameworks generally support the following capabilities at the user, group and enterprise level:

7.5.5.1 User Security:

(1) **Audit Trail**—The LIMS system should minimally be able to provide a user id and timestamp on all data activities a user performs (insert, update and delete) and allow that data to be accessible for audit trail reporting.

(2) **Single Sign-on**—Most LIMS systems have a unified authentication scheme whereby a user can log in once and access all of the functions for which he/she has permission without requiring an additional log in.

(3) **Session Timeout**—The authentication scheme should also allow a configurable setting for requiring the user to enter user name and password again when the user has been idle on the system for more than a configured period of time (this is sometimes managed outside of LIMS by the enterprise network settings).

(4) **Password Policy**—The LIMS system should meet your company standards for requirements on password renewal, password combinations (that is, minimum character lengths

and combinations of characters and numbers), and encryption strength of database password storage.

(5) *Regulatory Compliance and Electronic Signatures*—The LIMS system should comply with applicable regulatory standards and company standards where electronic signatures or user acknowledgement of electronic records, or both, are employed.

7.5.5.2 Group Security:

(1) *User Assignment*—The LIMS system should support the ability to assign individual users to system groups or roles.

(2) *Query Assignment*—The LIMS system may optionally allow assignment of users in batch by querying other user information, such as department.

(3) *Functional Assignment*—The LIMS system should allow for the assignment of permissions to groups for specific operations in the LIMS system, such that a single group can have a standard set of permissions for a configurable set of granular activities (for example, performing specific data entry tasks, running specific reports or categories of reports, and so forth).

(4) *Audit Trail*—The LIMS system should provide an audit trail (userid and timestamp) of all changes to group membership or group permissions along with the administrative user who made the changes.

7.5.5.3 Enterprise Security:

(1) *Enterprise Directory or Network Security*—A LIMS system may optionally integrate with an organization's enterprise security directory (for example, LDAP, active directory, windows domain, and so forth). This feature provides the benefit of a single set of credentials for multiple applications, seamless integration for LIMS into corporate security policies, and more convenient and robust security administration of users (for example, removing a user in the directory removes the user's access to all applications, rather than requiring the user to be removed from user databases in each application).

(2) *Integration with Physical Security Systems and Public Key Infrastructures*—A LIMS system's security framework may also optionally integrate with an organization's physical security systems, such as security badges or biometric devices. See 7.4.4 for systems that supplement userid/password credentials with encrypted certificates, smart card systems, and so

forth. These features provide an additional validation and audit trail for authenticating users of LIMS, and (with the appropriate device integrated with LIMS access) may provide the additional convenience of passive authentication to the LIMS system rather than manually typing in a user id and password.

7.5.6 *Infrastructure*—The network and hardware infrastructure required by a LIMS system varies widely by the particular LIMS software and feature package chosen, but it is especially important to understand what network and equipment resources will be needed depending upon your chosen deployment method (central, regional, distributed). Table 2 lists a summary of some of the common infrastructure requirements and choices based on deployment strategy.

7.6 *Exchange of Data between LIMS and Enterprise Resource Planning (ERP Systems)*—Interfacing LIMS and ERP to enable an integrated quality management function as detailed in the chart below is a common requirement in quality control laboratory environment where both are deployed. The inspection requirements arise from the passage of materials along the supply chain managed by the ERP logistics modules. Data sharing between LIMS and ERP can facilitate a more rapid and efficient quality control/production process through the life cycle of each production batch (Fig. 8):

7.6.1 *Integration Options* (Fig. 2, block LE-6)—Approaches to facilitate LIMS integration with ERP systems range from the creation of customized interfaces to vendor provided configurable solutions. An array of technologies and techniques exist to help facilitate integration. LIMS vendors may provide vendor specific/certified interfaces for ERP integration, which removes much of the burden of custom development and costly maintenance from the implementer. The ISA 95 Enterprise/Control System Integration standards are also highly relevant and widely supported by many ERP and MES vendors. The standards describe integration between the business logistic management layer, which includes ERP systems, and the manufacturing operations management layer, which includes MES and LIMS systems. The World Batch Forum has developed XML Schemas that map to the ANSI/ISA-95 models. These define how to represent the ISA-95 information in XML, known as Business To Manufacturing Markup Language or B2MML V2.0.

TABLE 2 Infrastructure Requirements by Deployment Strategy

	Central	Regional	Distributed
Network Configuration	Wide-area network or virtual private network	Wide-area network or virtual private network	Local-area network
Security Infrastructure	Enterprise directory or internal LIMS database	Enterprise directory or internal LIMS database	Internal LIMS database or local Domain
Backups and Redundancy	Daily or continuous incremental data backups automatically to offsite location. Redundant data centers. Redundant network access points.	Daily or continuous incremental data backups automatically to offsite location. Redundant data centers, or shared failover between regional centers. Redundant network access points.	Daily backups with manual transport to offsite storage. Usually no data center or network redundancy.
Hardware Configuration	Mainframe or multiple- CPU servers with fully redundant components (server, motherboard, memory, disk, power supply, and so forth). Thin client configuration (terminal, remote desktop, web browser).	Multiple-CPU servers with fully redundant components (server, motherboard, memory, disk, power supply, and so forth). Client-server (desktop application with database and application server on back-end) or thin client configuration (terminal, remote desktop, web browser).	Departmental server with minimal redundancy or no server (for deployments with a few number of clients). Client-server (desktop application with database and application server on back-end) or standalone desktop configuration with local data store.

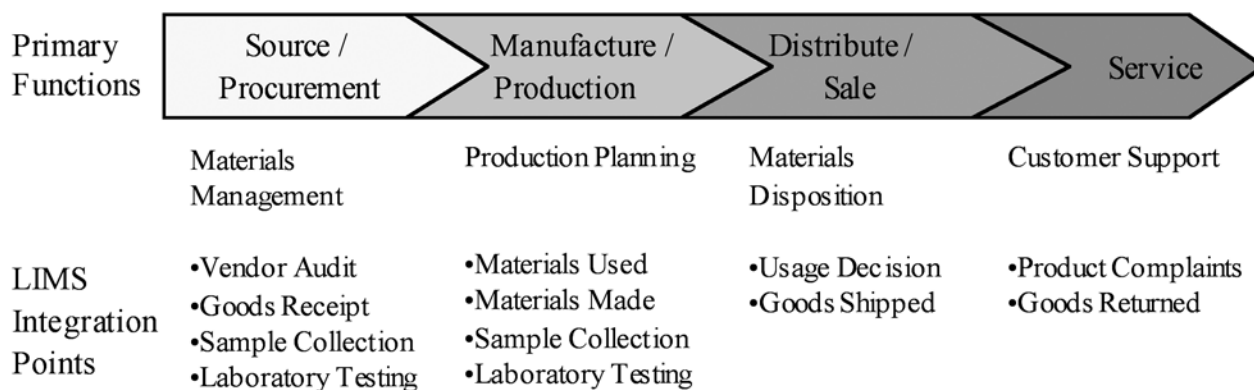


FIG. 8 LIMS—ERP Integration Points

7.6.2 *Integration Points Between LIMS and ERP*—Fig. 9 outlines the basic integration points between LIMS and ERP system.

7.7 Instrument Integration:

7.7.1 Interfaces between the LIMS and laboratory instruments (such as balances, pH meters, spectrophotometers, chromatographic data systems (CDS)) typically involve interfacing the LIMS system to another data system controlling the instrument hardware. The CDSs are typically addressed as a specific category due to their inherent complexity. Instrument integration with LIMS is a key factor in delivering effective cost benefits for the implementation.

7.7.2 Due to the inherent complexity of managing instrument sample test sequences (in some cases an auto sampler) in both the LIMS and the laboratory instrument data system, it is highly desirable that the LIMS download a test sequence to the laboratory instrument data systems via an interface to assure accurate exchange of information.

7.7.3 The laboratory instrument data system performs the measurement and transfers back to the LIMS the results for further processing in the LIMS. The laboratory instrument test result (determination, measurement) is transferred to LIMS via the LIMS-Instrument interface. Alternatively, the laboratory instrument data system may pull from the LIMS pertinent Samples, perform the necessary measurements, and then push back the results to the LIMS.

7.7.4 In many cases, it is highly desirable that the result transfer from the data system to the LIMS be done in real time such that further LIMS result processing (cross-technique calculations, approvals, reporting, and so forth) be done as soon as the measurements have been performed.

7.7.5 Where calculations are performed needs to be fully analyzed to gain the most from the LIMS and automation associated with specific instruments. The location of calculations should be guided by what functionality each system (LIMS versus laboratory instrument data system) provides. A good example is CDS. The CDS is good at performing system suitability, peak integration, calibration curves. The LIMS is good at cross-technique calculations, comparison to specifications, correcting for moisture and other factors, trending, approving and reporting laboratory results. Typical approaches for LIMS-CDS calculations include the use of ratios where the

CDS reports a sample response over a standard response and the LIMS is used to calculate the final result and compare it to specifications and further reporting.

7.8 Workflow Integration:

7.8.1 Laboratories typically manage and process samples in several ways. The two primary approaches include; (1) testing individual samples one at a time, completing all tests on one sample, and (2) batching multiple samples that have the same test.

7.8.2 A single sample or a group of related samples (typically referred to as a worklist) is created inside the LIMS based on the laboratory's need for batching of samples with identical or similar test techniques.

7.8.3 The LIMS is typically used to query the available samples with identical or similar test requirements. This speeds the assignment and completion of testing.

7.9 Industry Data Exchange Technology:

7.9.1 Interfacing LIMS with instruments typically means interfacing two moderately complex or highly complex systems together. Best IT practices must be used to do so. Ideally the interface between the LIMS and the instrument should be file-less as files represent security risks or increase overall system complexity to mitigate this risk. The XML data standard represents the de-facto industry standard as the information format to exchange structured data between two systems. Furthermore, XML can be permanently stored as files or temporary exchanged as a stream between two cooperating applications. Due to its broad acceptance, XML format is very highly recommended as the base format for interface to instruments. Industry standard tools (such as XSL) can be used to transform the data representation from one vendor to another, again increasing the value of an XML-based interface.

7.9.2 Several standards exist to facilitate the exchange of instrument data to LIMS: ANSI HL7 defines clinical information data sets; Specification E 1947; Guide E 1948; Specification E 2077; and Guide E 2078.

7.9.3 The US Environmental Protection Agency (EPA) employs the staged electronic data deliverable (SEDD)¹⁰ standard. The EPA SEDD standard employs the extensible markup language (XML) technology and is a joint standard developed by U.S. EPA Office of Superfund Remediation and Technology

Possible Points of Data Exchange Between ERP and LIMS

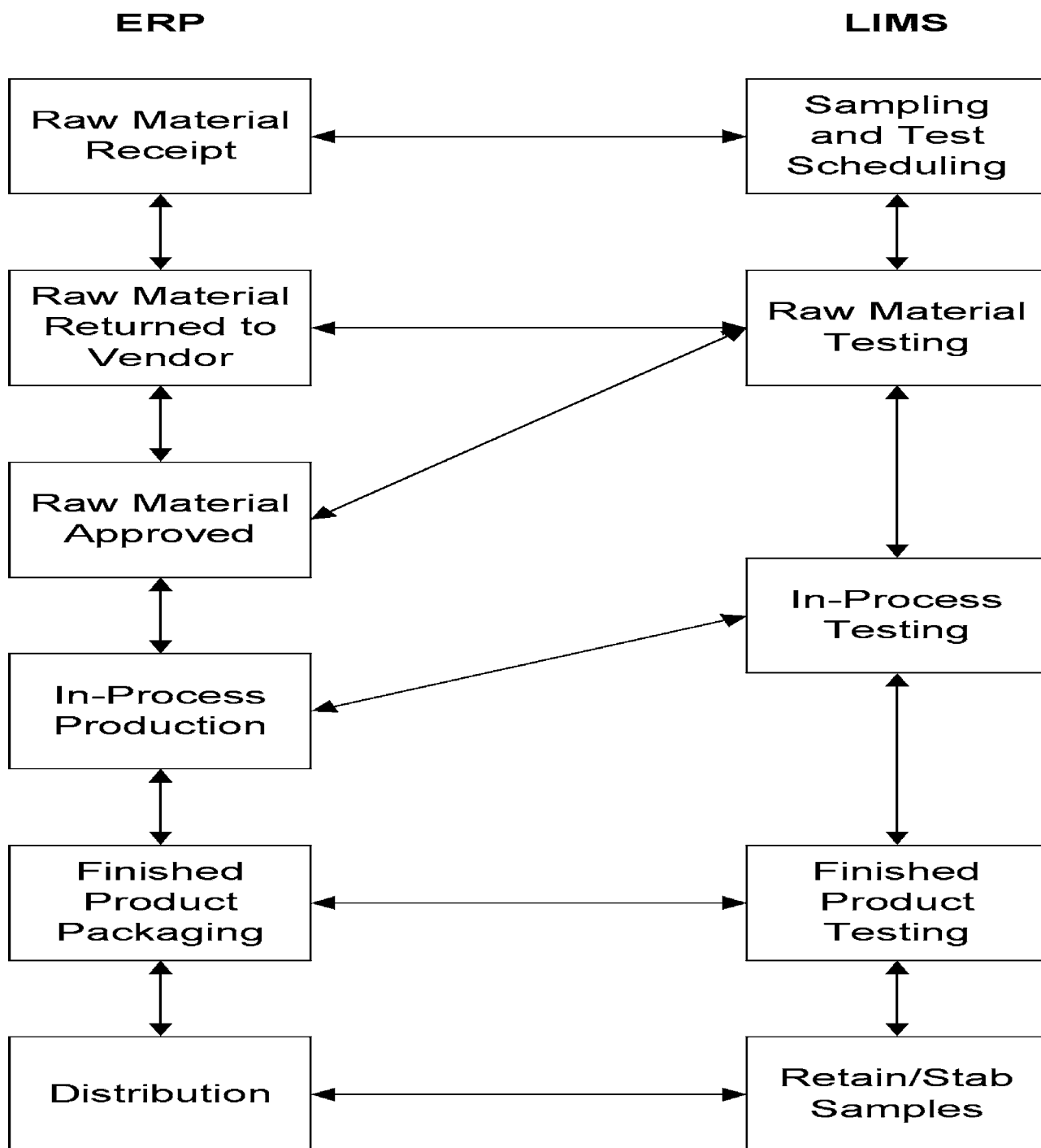


FIG. 9 Possible Points of Data Exchange Between ERP and LIMS

Innovation (OSRTI), Analytical Services Branch (ASB), and U.S. Army Corps of Engineers (U.S. ACE).

7.9.4 The XML technology provides a standard data format to exchange information between the LIMS and the instru-

ments. Emerging standards such as analytical information markup language ([AnIML](#)) build on the XML data format to provide a base standardization of common information elements representation. The AnIML Core Schema defines xml

tagging for any kind of analytical information. A set of analytical technique instance documents (atid) xml files, one per analytical technique, applying tight constraints to the flexible core and which in turn are defined by the technique schema; extensions of technique schema are possible for vendor-specific and institutional-specific parameters. The utilization of XML data format standards (such as AnIML) provide standard structures for integrating instruments to LIMS.¹³

7.10 Enterprise Application Integration and Middleware—If the organization has a defined enterprise application integration strategy, or a standard enterprise middleware platform, the LIMS application should be evaluated for its ability to easily integrate with the middleware's supported standards. Integration with a central middleware platform can substantially reduce integration costs and complexity. A single integration implementation of the LIMS system to the middleware platform can then potentially support information exchange with multiple enterprise applications connected to the same middleware hub.

7.11 Digital Content—A wide array of digital media (images, site and corporate SOPs, reports from miscellaneous instruments, user training records, supplier reports, etc) are stored in a secure way in the laboratory. Storage and management of these supporting documents and electronic records are typically in external EDMS or data archive systems with links LIMS. Access to these digital content sources can be accomplished through a wide range of technical solutions (enterprise data share, content management system, and so forth). Access to this data storage from the LIMS is highly desirable, especially if integrated into the LIMS workflow through configuration.

7.12 Electronic Laboratory Notebook (ELN) Integration—The integration of ELNs with LIMS can take many forms dependant on the business functions performed by each system. Electronic Laboratory Notebook functions vary widely but are generally in two categories, specific ELNs and cross-disciplinary ELNs. specific ELNs contain features designed to work with specific applications, scientific instrumentation or data types. Specific ELNs can be closely integrated with a LIMS to pass information from LIMS to the ELN and from the ELN to LIMS. Refer to laboratory execution system (LES) as an example of a specific ELN. cross-disciplinary ELNs or generic ELNs are designed to support access to all data and information that needs to be recorded in a laboratory notebook and are typically integrated with a LIMS as a pointer or reference within the LIMS. Specific LIMS implementations can vary between a full implementation that covers the laboratory bench and instrument integration up to final results and material (lot) disposition (with LIMS performing the function of an ELN) to more limited implementations where the LIMS manages the final result but does not fully cover the laboratory bench or instrument integration functions and the

ELN manages information on the laboratory bench and integration with instruments and passes this information to LIMS.

8. LIMS System Life Cycle

8.1 Introduction—The LIMS system life cycle refers to the activities that are taken to acquire, implement, operate, and eventually retire a LIMS. While many of these activities must occur simultaneously, the LIMS system life cycle is best understood by organizing the activities into phases that typically occur almost sequentially. This organization of the activities into the phases facilitates planning and provides checkpoints for managing a LIMS project. The LIMS system life cycle is illustrated in Figs. 10 and 11. The top diagram reflects the overall LIMS system life cycle, the bottom diagram shows the hardware life cycle. The numbering scheme (SX-n) refers back to the life cycle phase diagram in 5.3. The LIMS system life cycle starts with an initial LIMS implementation. This may be the introduction of a LIMS to replace manual procedures in the laboratory or the replacement of a legacy LIMS that has become obsolete. The initial LIMS implementation is analogous to a software development project and typically takes 1 to 2 years to complete. This is followed by an operational phase where the LIMS is used for operations to manage data and work flow. The LIMS will evolve during the operate and maintain phase. Support resources will add new reports and enhancements through the use of configuration and customization tools. The LIMS vendor will provide updates and upgrades as improvements are made in the product. The operate and maintain phase can easily last 10 or more years. Retirement will occur when the system or underlying infrastructure becomes obsolete or when organizational changes dictate a new direction for the LIMS.

8.1.1 LIMS Life Cycle Phases:

8.1.1.1 Project initiation is the first phase in the initial LIMS implementation and also for major LIMS upgrades that occur during the operate and maintain phase. The LIMS project or upgrade is conceived and the business case is developed including estimating costs and benefits. The initial scope and boundaries of the project are determined.

8.1.1.2 Requirements analysis is the phase for determining what functions and features are required of the LIMS and determining how the LIMS will support the laboratory work flow. Requirements analysis occurs during the Initial LIMS Implementation and also for major LIMS upgrades. Requirements provide the basis for the LIMS product selection. A LIMS product is typically selected during the requirements analysis phase of the initial LIMS implementation.

8.1.1.3 Design is the phase where system functional requirements are translated into detailed logical and physical design specifications. Design occurs during initial LIMS implementation and during major LIMS upgrades.

8.1.1.4 Build/Configure is the construction phase of the LIMS system life cycle. Activities include development, customization, configuration and developmental testing of the LIMS. Build/Configure activities occur during both initial LIMS implementation and during major LIMS upgrades.

8.1.1.5 Testing and commission is the phase during which the LIMS is put into operation. Activities include validation

¹³ Schaefer, B. A., Poetz, D., and Kramer, G. W., "Documenting Laboratory Workflows Using the Analytical Information Markup Language," *Journal of the Association for Laboratory Automation*, 9 (6), 2004, pp. 375-381.

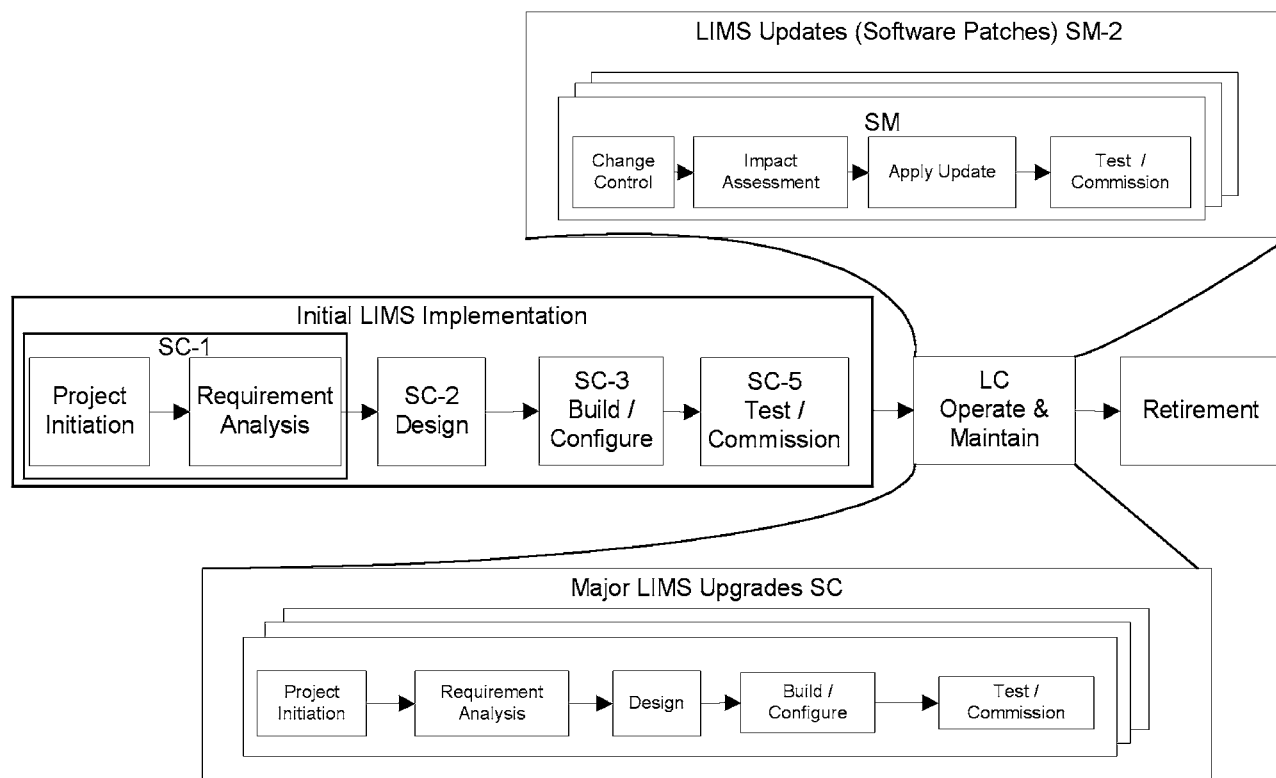


FIG. 10 LIMS System Life Cycle

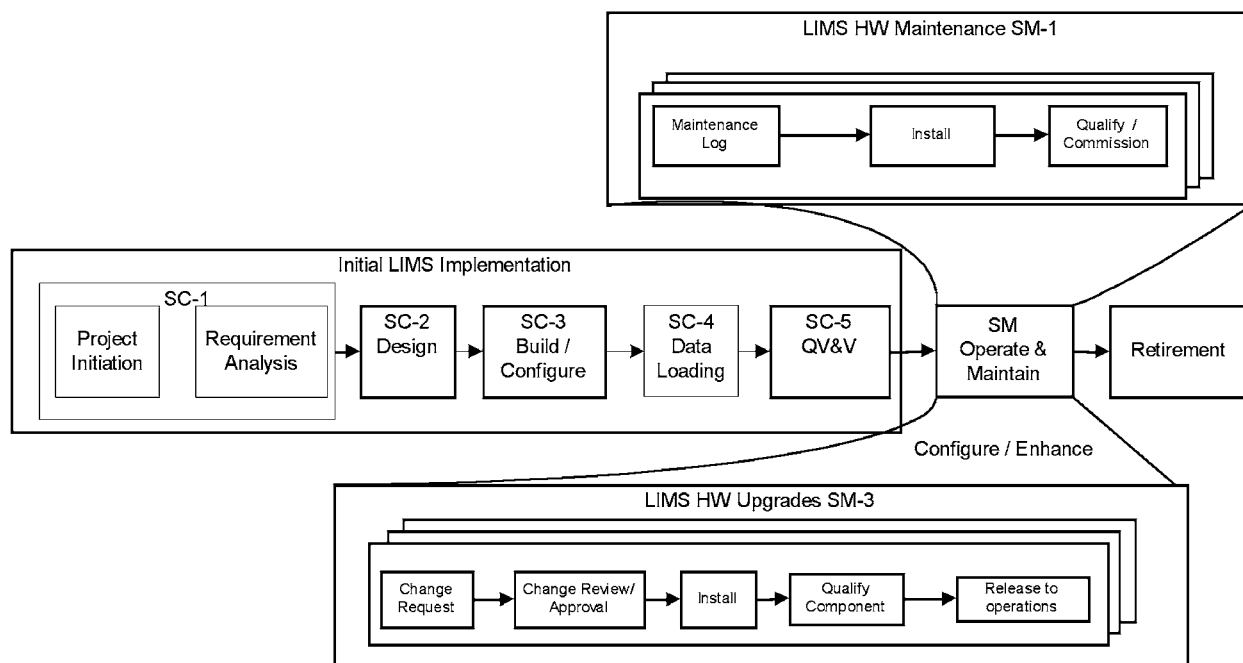


FIG. 11 LIMS System Life Cycle—Hardware

testing, data loading, final acceptance and deployment. Reference the GAMP documents in 2.5 for detailed guidance on validating LIMS in regulated environments.

8.1.1.6 operation and maintenance is the production phase. The LIMS is used to support the laboratory operations.

Maintenance and support is performed on the LIMS to keep it aligned with changing business and technology requirements. The LIMS evolves through updates and upgrades. Updates usually involve small changes to address software bugs with the system. These updates can be applied to the system after

their impact is analyzed. Upgrades are provided when the vendor makes a improvements in the LIMS product. Implementing an upgrade typically involves applying new functionality in the laboratory. Applying upgrades can be a major project and follows a process similar to the initial LIMS implementation.

8.1.1.7 Retirement is the final phase in the LIMS system life cycle. Usually driven by technology obsolescence, Retirement involves planning for the next generation LIMS including providing access to relevant historical laboratory data.

8.1.1.8 An understanding of the LIMS system life cycle and good practices for each of the activities will help first time LIMS implementers plan and manage their LIMS projects. Seasoned LIMS users can use the LIMS system life cycle to maintain existing LIMS and prepare for the implementation of the next generation LIMS.

8.1.2 *LIMS Life Cycle Implementation Project Flow*—The flow for a typical LIMS system life cycle is illustrated in Fig. 12.

8.1.2.1 The time frames illustrated in Fig. 12 are given as examples for implementing a LIMS in a typical laboratory with approximately 100 users including 30 to 50 analysts. The implementation would provide complete functionality to support the operation. Functionality could include manual and automated sample receiving and assignment of tests, sample tracking, chain of custody, assignment of work, bi-directional instrument integration with 10 instrument types, multiple levels of review and approval, reporting, instrument calibration and maintenance management and standard and reagent management. Customization and configuration would be limited by the amount of resources applied in design and build/configure phases. Effort and duration of validation activities assumes the use of a commercial LIMS product and that the vendor has

properly tested the product during their development. Validation would include installation and operational qualification of the system, verification of calculations and user acceptance testing to verify work flow and user requirements but would not include detailed functional testing of LIMS functionality.

8.1.2.2 It should be noted that while the phases occur sequentially, Design could start prior to completion of requirements and build/configure could start prior to completion of design. This would occur particularly with a prototyping implementation where feedback from user review is used to adjust the design. Build/Configure is normally the longest and most difficult phase during Initial LIMS Implementation and also during major LIMS upgrades. The operate/maintain phase is the longest phase in the LIMS life cycle and should last many years as long the LIMS provides benefits to the business.

8.1.2.3 The bottom of the diagram provides an example of typical resource utilization during the various phases of the LIMS life cycle. Three types of resources are illustrated including lab or business users, LIMS experts or developers and validation experts.

8.1.2.4 Lab Users are heavily involved in the business planning that occurs during project initiation and requirements analysis. They have less involvement during the design phase but get re engaged during build/configure and are normally heavily involved in the test/commission phase. Lab/business users provide administration and support and participate in updates and upgrades during the operate and maintain phase.

8.1.2.5 Some support from LIMS experts is useful during project initiation and also during requirements analysis where they contribute an understanding of LIMS functionality and what is reasonable and feasible. LIMS experts do the bulk of work during the design and build/configure phases. They also provide support for the test/commission phase.

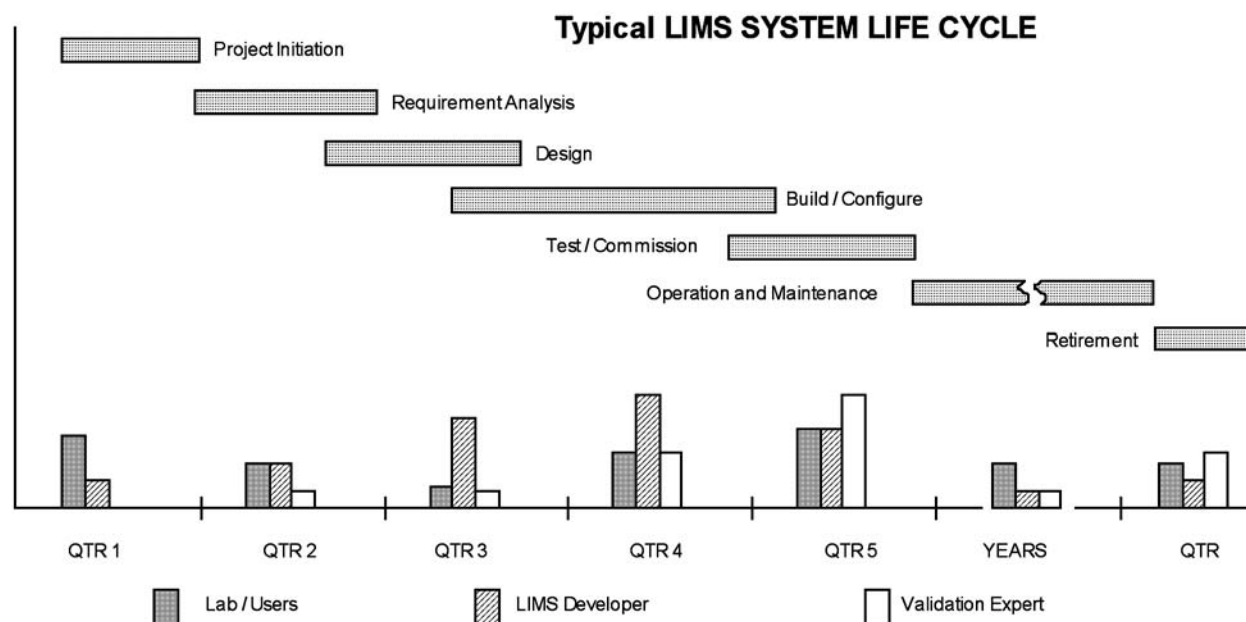


FIG. 12 Typical LIMS System Life Cycle

8.1.2.6 In regulated environments, validation experts get involved during the requirements and design phase to begin validation planning. They write test scripts during the build/configure phase. Their heaviest involvement occurs during test/commission phase where they facilitate execution of testing.

8.1.3 *Factors Affecting LIMS Projects*—Alternate scenarios could result in considerably different time frames for implementation of a LIMS project. Some of the factors that would impact on LIMS project time frames are listed in [Table 3](#).

8.1.3.1 The time required by laboratory personnel to fully implement a LIMS and gain the financial benefit is frequently underestimated. The underestimation of LIMS implementation time is much more severe in large installations.

8.1.3.2 LIMS implementation, as with most significant IT projects, requires a long term commitment from various groups in order to be successful. This includes ownership and support from the user community management coupled with effective delivery of the IT components of the project from the IT group. This IT group may be internal to the specific company or may include outside consultants/third parties.

8.1.4 *Some of the Common Pitfalls that Occur During a LIMS Project*:

8.1.4.1 Lack of commitment from user community management,

8.1.4.2 Lack of experience with LIMS from IT community,

8.1.4.3 Under-estimating the effort and complexity of the task,

8.1.4.4 Inadequate resource planning and commitment,

8.1.4.5 Excessive customization,

8.1.4.6 Inadequate change management,

8.1.4.7 Failure to address the role of the lab in the organization,

8.1.4.8 Failure to manage expectations, and

8.1.4.9 Too slow getting LIMS into production.

8.1.5 *Keys to Success Mirror Frequently Encountered Pitfalls*:

8.1.5.1 Assign a project manager and develop and manage against a project plan with activities, checkpoints and deliverables. Provide frequent project status reports.

8.1.5.2 Involve key users throughout the project life and require commitment from them.

8.1.5.3 Make sure that adequate work process evaluation is done before starting the project to ensure that the end-point is clear and well understood by all.

8.1.5.4 Manage scope. Utilize a change control board to approve any additions or changes to scope.

8.1.5.5 Avoid excessive customization by using as much out-of-the-box functionality at possible. Customization adds time to the development and testing activities and provides additional opportunities for introducing error.

8.1.5.6 Utilize experienced LIMS resources to implement the project. Resources assigned to development and configuration activities should have experience with the specific LIMS product being installed.

8.1.5.7 Provide adequate training to all resources engaged in the project.

8.1.6 *Other LIMS Project Considerations*:

8.1.6.1 Organizations should consider fresh new ideas on how to effectively use the LIMS tool. Replicating outdated paper systems should be avoided.

8.1.6.2 New policies will be needed to protect the valuable information assets. Examples of policies include security, data backup, data archive, and disaster recovery.

8.1.6.3 The new LIMS should be designed to make the data request and report formats as transparent to the end users as possible, so that disruption of services is kept to a minimum.

8.1.6.4 The importance of a clearly defined alternate method of reporting is critical, and this alternative (manual backup) should be tested on a periodic basis.

8.1.6.5 Laboratory information maintained in the LIMS needs to be freely available to client users who are authorized to use the information.

8.1.7 *Alternatives to LIMS Life Cycle Implementation*:

8.1.7.1 Initial LIMS implementation can be slow and expensive to complete. Following the LIMS system life cycle produces the best results for companies that understand LIMS and can support the implementation of those functional requirements that are not met by their selected product. Smaller companies may benefit from alternatives to the formal techniques listed in [8.1.8.4](#) through [8.5](#). Alternatives include (1) installing and trying each of several LIMS that meet the basic requirements, (2) simply evaluating LIMS against the LIMS guide checklist in [Appendix X1](#) and (3) adjusting laboratory work processes and procedures to make best use of selected commercial LIMS functionality.

8.1.8 *Validation*:

8.1.8.1 The validation of LIMS is a mandatory step for regulated industries. Specific validation requirements exist for

TABLE 3 Factors Affecting LIMS Project Time Frames

Factors Affecting LIMS Project Time Frames	Impact
Significant customization of standard "out of the box" functionality required to meet requirements.	Longer Design and Implementation and Transition Phases
Large or multiple laboratories with multiple products and/analyses and/or complex work flows	Longer times required for all phases. Most impact in Design and Implementation and Transition Phases
Harmonization of requirements for multiple labs	Considerably longer Requirements Analysis Phase, other phases may also extend
Large data migration effort required from legacy LIMS	Potentially longer Transition Phase
Complete Validation with Operational Qualification requiring complete functional testing	Longer Design and Implementation and Transition Phases
Small Lab, few products and analyses, general acceptance of "out of the box" functionality with minimal customization.	Shorter Requirements and Design and Implementation Phases

industries regulated by the [FDA CFR Part 21](#), EPA ([40 CFR 160](#)), and [FDA CFR Part 50, Appendices B, E, and K](#).

8.1.8.2 Validation of LIMS can add three to twelve months to the implementation time. Documentation plays an important role in the validation process for LIMS. See [IEEE 829](#), [IEEE 1008](#), [IEEE 1012](#), and [IEEE 1028](#).

8.1.8.3 Reference the GAMP documents in [2.5](#) for additional detailed guidance on validating LIMS.

8.1.8.4 The individual phases in the LIMS system life cycle are addressed in more detail in the sections that follow.

8.2 *Phase 1—Project Initiation*—The purpose of this section is to outline the activities that occur during the project initiation phase of a LIMS system life cycle. The section contains three parts including project initiation and planning, developing a business case and developing a quality plan. Using this section will provide the reader with a guideline to follow as they initiate a new LIMS project.

8.2.1 *Project Initiation and Planning*—A project definition document should be developed that defines the scope of LIMS early in the LIMS project. Fundamental questions that should be addressed include: (1) will all labs within a department or organization be included or just a few; (2) is there more than one physical site included in the LIMS; (3) are there any time boundaries on LIMS implementation/operation; (4) are there any staffing limitations; (5) are there any training/skills limitations; (6) are communication links to external computer systems required; (7) are laboratory instruments going to be directly linked to the LIMS?

8.2.2 *Developing a Business Case*—The decision to implement, upgrade or replace a LIMS can driven by many factors. For example, it may be necessary to upgrade a LIMS to meet regulatory reporting or compliance requirements. Regardless, the business aspects of a LIMS need to be considered; for example, total resources (funding available, number, and skills of laboratory staff), time requirement (for implementation, processing laboratory samples), short and long-range business plans, and objectives. Implementation of a LIMS can usually be justified on the basis of a cost-benefit analysis. Good LIMS cost-benefit analysis requires time, solid understanding of the laboratory environment, and comprehension of the benefits realized. Care must be taken not to over analyze the cost-benefit factors beyond the precision required for the project. Cost-benefit factors need to be addressed with other non-cost factors in making the decision to install a LIMS. The cost-benefit components of not implementing LIMS should also be addressed. LIMS Costs An analysis of LIMS costs should consider the total cost of ownership including the initial purchase costs, the cost of implementation and the ongoing cost of ownership. LIMS benefits are normally considered as tangible or hard benefits that can easily be quantified, and Intangible or soft benefits that are recognized but are more difficult to calculate. In fact, with some creativity and hard work, most benefits can be calculated and it is valuable to do this. In calculating benefits, it is important to document the assumptions that go the calculations. An understanding of the benefits of implementing the LIMS not only help justify the project but should set the priority for implementation of certain features or functions that can provide significant benefit. Some

areas to investigate for LIMS benefits are listed in [Table 5](#). It is easy to get overly enthusiastic about the potential benefits from introducing information technology like LIMS. It is important to understand that the benefits result from hard work integrating the best of technology, people and work processes. Focus on what is important and avoid creating expectations that are too high and cause management to become disillusioned. This usually leads to project failure. Common error in LIMS planning include:

8.2.2.1 Expecting to achieve a paperless office in the initial phase of the implementation,

8.2.2.2 Expecting an out of the box product to meet all requirements,

8.2.2.3 Expecting an immediate increase in productivity,

8.2.2.4 Underestimating laboratory staff time required to build LIMS test tables and format the system to user specifications, and

8.2.2.5 Failure to have strategic planning (funds, personnel, and space) for expansion/replacement of the LIMS.

8.2.3 *Quality Plan*:

8.2.3.1 *Quality Planning*—A sound, well thought out quality plan insures a smooth, successful implementation and validation of a LIMS system. A quality management system should include a quality policy and objectives, a quality manual, required documented procedures as well as documents needed to ensure effective planning, operation and control of processes, and required records. The documented procedures within the quality management system must define how such a system will be implemented, validated, and maintained in a validated state during routine use and retirement of the system (system life cycle In addition, the quality plan defines the required documents for such a project such as user requirements Ddocument, functional specifications document, system design document, validation plan, and so forth.

8.3 *Phase 2—Requirements Analysis*—The purpose of this section is to outline the activities that occur during the requirements analysis phase of a LIMS system life cycle. The section contains five parts including workflow analysis, business requirements, validation/quality planning, risk analysis, and system evaluation and selection. Using this section will provide the reader with a guideline to outline their LIMS requirements.

8.3.1 *Workflow Analysis—Current and Future State*:

8.3.1.1 *Model Current State Laboratory Practices*—Meet with LIMS users, end users, laboratory managers and external users of laboratory information. Diagram sample work flow and information captured in the laboratory (see LIMS work flow diagrams, [Figs. 4 and 5](#)). The time required to model current laboratory practices can range from a few days to several months. Extended modeling may be counterproductive, if the time exceeds several weeks. Rapid prototyping may be more productive (see [Guide E 1340](#)). There are many good reference materials and external consulting parties available to assist in the modeling process.

8.3.1.2 *Model Future State Laboratory Practices*—The future state for laboratory practices needs to be defined prior to LIMS implementation/selection. Failure to perform this step may lead the user to automate a broken wheel. First fix the

TABLE 4 Total Cost of LIMS Ownership

Type of Cost	Category	Example Components	Comments
Purchase Costs	Computer Hardware	Servers PC Terminals Hand Held Devices	
	Software	LIMS Data Base Instrument Integration Graphics	
	Infrastructure	Cabling/Wiring Routers Power Conditioning Climate Control Space/furniture Installation	In many cases infrastructure will already be in place.
Implementation Costs	Project Related	Project Management Requirements Development Design	It is important to understand the impact on laboratory staff who will may participate in the project while continuing their normal work activities.
	LIMS Configuration/Customization	Sample Types Specifications Tests/Methods Calculations Reports Customization of code	
	Test/Commission	Development of SOPs Training Testing Data Loading Data Migration Validation	The level of effort to load the static data are generally the single largest cost factor for LIMS implementations (this is especially true for large/complex implementations), reference 8.6.2. The effort to author SOPs, Training the project staff and end users, validate (test) the application (especially for regulated industries) and migrate historical data are typically the second largest cost area for LIMS implementations.
	Extended Functions	Instrument Integration System Integration	
	Annual Contracts	Software licenses Software support Rent/lease	
Ongoing Costs	Internal Support	System Administration Training Enhancement Validation	Generally a % of the software license purchase.
	Enhancement Support	Configuration Training Validation	
	Consumables	Media	It is frequently necessary to make changes to keep the LIMS current with the lab environment and needs.

wheel (laboratory work flow) and then automate the optimum work flow. LIMS should not be used to set laboratory policy or procedures, but LIMS may be used to enforce them. Invest the time to evaluate LIMS solutions as part of the “Model Future State Laboratory Practices” exercise. Understanding capabilities of LIMS will enable users to define the best approach to the future state.

8.3.1.3 Harmonize Process—During the modeling process, find areas to harmonize within the laboratory, between laboratories, and across sites when possible. This will allow the LIMS effort to be leveraged throughout the enterprise.

8.3.2 Business Requirements Analysis:

8.3.2.1 The high level business objectives and strategies need to be understood and considered as a prerequisite for developing detailed business requirements for a LIMS. The business requirements need to be written, clearly define what is needed and be approved by key stakeholders of the future business process. The business requirements need to be realistic in terms of what can be expected of commercial systems or what can be expected for custom developed systems based on know resources and their capabilities. This information is typically defined in a high-level project charter.

8.3.2.2 Document Specific Business Objectives—Clearly defined and documented business expectations will provide the backdrop for the LIMS project. Gain agreement from all parties who will be stakeholders in the LIMS project including laboratory personnel, management, quality oriented staff, information technology/services people, legal, documentation specialists, and engineering. Be certain that all parties understand the business objectives, and know how to test decisions against these business objectives.

8.3.2.3 Legal Issues—Legal constraints on how your laboratory uses information need to be addressed. Regulatory requirements may necessitate specific LIMS features like audit trails of LIMS transactions.

8.3.2.4 Business requirements may require signed hard copies for all laboratory documents. Legal departments (if they exist) should be consulted on how you are planning to use the LIMS. Concepts of best available evidence for laboratory records need to be reviewed and understood by LIMS users. Careful examination of regulations should be done to determine if there is a need for: (1) reported results to have provisions for two verifications, (2) reported results changed during on-line operations to generate an audit trail, and (3)

TABLE 5 Example Benefits Attributable to LIMS

Type of Benefit	Description	Comment
Tangible—Calculation of benefit is straightforward	Lab Throughput	Increase in the number of samples processed in the lab during a specified period
	Lab Turn Around	Decrease in the amount of time required to process a sample in the lab
	Labor Savings	Reduction in the number of analysts hours required to process a given number of samples in the lab
	Reduction in the number of errors encountered in the lab	Reduction of errors can reduce the amount of rework and can impact on the amount of time spent on laboratory investigations. Data entry errors are reduced by data capture programs. Fewer processing errors occur due to the introduction of more formal and better documented work procedures.
	Reduction in the cycle time to release manufactured products	Integration of LIMS with MRP/ERP systems enables release of product as soon as laboratory results are approved.
	Reduction in the amount of time spent on laboratory investigations	Immediate availability of data and consistency of process facilitates the laboratory investigation process
Intangible—Benefit calculation requires more creativity	Better Service Management	Benefits will accrue from the ability to integrate information more closely with suppliers and clients
	Access to Information/tools	Access to data and tools provided with LIMS will enable user to produce reports including new ways of looking at data. Data Access will also facilitate routine and non-routine problem solving.
	Easier to achieve and demonstrate regulatory compliance	Implementation of LIMS technology is typically coupled with improvements in work process and standard operating procedures. The LIMS will facilitate process improvements in some cases by forcing user to follow a consistent quality process. Access to data will make it easier to demonstrate quality processes are in place.
Other Benefits may accrue	Earlier correction of errors Laboratory Management	Tools provided by LIMS will enable more consistent work flow and management of laboratory operations
	Fewer lost samples	Sample tracking functionality can provide knowledge on the status and location of all samples
	Earlier intercepts of drifts in product quality	Closer coupling between the lab and manufacturing will provide quicker indication of quality trends

provision that archived data and test/requester tables be loaded into present system for retrieval of information. Retention periods for both raw data and LIMS resident data need to be evaluated and defined.

8.3.3 Validation Planning:

8.3.3.1 The rationale of validation planning is to ensure that computerized systems perform according to predetermined specifications in an accurate, reliable, and repeatable manner. The purpose of the validation plan is to establish the process to be followed and the documentation needed to provide evidence that the system has been installed and is operating according to approved requirements, specifications, design, corporate policies and regulatory and business needs. Validation planning is especially important in the industries that are regulated by GXP (GMP, GLP, GCP).

8.3.4 Risk Analysis:

8.3.4.1 There are several well documented reference materials associated with risk assessment. Users should evaluate projects against their business objectives and the risk associated with the LIMS project. Risk parameters include, but are not limited to, overall project risk, product risk based upon the

impact that the LIMS System will have on the product quality, people risk in the area of change management, and business risk.

8.3.5 System Evaluation and Selection:

8.3.5.1 *Request for Proposals (RFP)*—Issue a request for proposals (RFP) to LIMS vendors. The RFP should include a summary of your functional requirements, annual sample quantity, test complexity, process flows, sample work flow/model and business rules to define your specific needs.

8.3.5.2 The LIMS concept model and example RFP functions in this standard guide can be used to identify your requirements to the LIMS vendors. The sample workflow models found in **Figs. 4 and 5** and **Appendix X1** can be tailored to match your laboratories' requirements. Time required to write and issue a LIMS RFP can range from a week to a month or more.

NOTE 8—Custom LIMS can be built in-house. Custom-built LIMS are recommended only if unique requirements demand it. The cost of building and maintaining a LIMS in-house needs to be compared to the cost of purchasing a LIMS. The functions in commercial LIMS need to be compared to your specific laboratory functional requirements.

8.3.5.3 Evaluation & Selection—Quotations received from LIMS vendors should be evaluated against the functional requirements document. Objective judgments of the advantages and disadvantages of each LIMS product should be made. Weights can be assigned to each LIMS function for complex systems. Refer to LIMS checklist in [Appendix X1](#) as starting point and add your own functional requirements. The people who will be interacting with the LIMS should take an active role in the evaluation and selection steps. Site visits to installed systems are recommended. See section on LIMS database technology and hardware platforms for additional issues. Some guidelines for selecting a LIMS include:

(1) Select a LIMS where the database structure of the static tables/files (profiles, tests, calculations, specifications and related information) closely matches your current information structures and work flows.

(2) Select a LIMS that maintains the statuses that you need for running your laboratory.

(3) The criteria for LIMS selection should be driven by the software function. Hardware should be a second priority behind overall software functionality.

(4) Select a LIMS where the combination of the LIMS application and its underlying technology closely matches your laboratory work flow requirements and information structure.

(5) Select a LIMS based on a commercial database management system or database toolbox that is reliable, effective and supported external to your LIMS vendor (this is especially true if there is a chance that you may change your LIMS in the future). Proprietary LIMS database management systems may be required to meet specific performance requirements. Portability of data are a key factor in selecting a LIMS, including compatibility with an industry standard for accessing data.

(6) Select a LIMS based on database technology that permits the end-user to add/modify fields, indexes, relationships, tables, and codes.

(7) Select a LIMS where the database structure of the static tables/files (profiles, tests, calculations, specifications and related information) closely matches your current information structures and work flows. Move section to work flow team.

(8) Select a LIMS where the database structure of the dynamic tables/files matches the information types (numeric, date, memo) used in your laboratory. Move section to work flow team.

(9) Select a LIMS that permits third party tools to be used for report generation, export, import, and links to external systems, security, and monitoring beyond functionality built directly into the LIMS.

8.3.5.4 Purchase—The purchase order must contain conditions and provisions that are required by the end users. Typical items include delivery dates, acceptance testing, payment schedules, source code, software support and update policies, required documentation, training, installation, warranties, listing of all hardware and software. Formal contracts can be attached to the purchase orders.

8.3.6 Vendor Demonstrations:

8.3.6.1 Include a vendor demonstration as part of the selection process. The demonstration should be planned in

advance. Provide the vendor examples of types and samples and test methods and with a set of scripted scenarios including expected results that address areas of functionality that are critical to the successful implementation of the LIMS Typical examples of areas for demonstration could be complicated calculations or limit checking, automatic generation of samples and assignment of tests for skip lot testing, or complicated retest logic. Give the vendor at least two weeks notice prior to a vendor demonstration.

8.4 Phase 3—System Design:

8.4.1 Functional Requirements Analysis:

8.4.1.1 Depending on the system development life cycle being followed, functional requirements may be defined in a separate document from user (business) requirements. User requirements describe in business language, what the system is required to do to support the business process. Functional requirements describe in the language of the functionality of the system, how the system will satisfy the business requirements. In other words, Functional specifications are the translation of business requirements (what the system is required to do) into application-specific language (how the final system will satisfy business requirements). Functional requirements, as with business requirements, should be specific, measurable and realistic to ensure testing can easily confirm that requirements have been satisfied.

8.4.1.2 Where business and functional requirements are defined in separate documents, they should be linked using a traceability matrix to provide complete traceability between business and functional requirements.

8.4.1.3 Functional requirements may be documented in the form of a checklist of major features and functions of the system that will be configured and used (see the LIMS guide checklist in [Appendix X1](#)). The LIMS concept model can be used as a starting point in developing a list of LIMS functions. See [IEEE 830](#). Specific hardware and software standards in place at a particular lab or company may also need to be adhered to and therefore specified in functional requirements.

8.4.1.4 Functional requirements analysis is normally performed after selection of a commercial package based on business requirements is complete. It is the process of analyzing the existing and proposed systems. The existing system may be a manual, paper-based system or a legacy computer system or systems. The output of the process is a Functional Requirements Specification. This specification describes in detail the proposed system functions. Defining the functional requirements requires a thorough understanding of the business process(es), user requirements and proposed system capabilities. Functional requirements should include the following areas: business requirements, hardware, software, user interface, performance, availability, system interfaces, security, regulatory, system management, documentation, help, and training.

8.4.1.5 Requirements should be ranked in order of importance (for example, critical, important, nice to have) to facilitate decisions on whether certain functions may be implemented in initial phases or delayed until later phases.

8.4.1.6 Rapid prototyping development techniques may be used to assist users and the development team in elaborating

high-level functional requirements into a detailed set of functional specifications. When working with commercial off-the-shelf products, this process is also beneficial in helping constrain user expectations and system scope within the out of the box capabilities if avoiding excess configuration or customization is desired.

8.4.1.7 Development of detailed functional requirements may take anywhere from five to twenty days or more excluding prototyping time and review/approval cycles.

8.4.2 *System Design:*

8.4.2.1 The resulting deliverable is the system design document (SDD) that details the system design specifications. The SDD should provide enough information to build or configure the system software and network architecture.

8.4.2.2 During this phase users may again be interviewed to get specific details of process flows or elaborate on requirements for a definable portion of the business process. Process flows are constructed in layers so that an overview of the system processes are understood including the system interfaces as well as the details of specific area workflows so that adequate design specifications are documented for configuration/coding clarity.

8.4.2.3 When using prototyping techniques, specific information about how the system will be used and functions required by the users will be gathered and listed in a high-level system design document that will be used to guide the prototyping.

8.4.2.4 The system architecture as well as hardware and software may be defined at this stage as well. This could be defined in one section of the system design document or in a separate system architecture specification document.

8.4.2.5 Development of system designs may take anywhere from five to fifty days or more excluding prototyping time and review/approval cycles. The more design that is afforded up front in a project the smoother the configuration/coding will proceed.

8.4.3 *Unit and Integration Test Planning:*

8.4.3.1 An approach for testing the system under development should normally be defined during the design activities as well. Development testing for custom software code normally includes unit, integration and system testing of the configured system modules, interfaces and any custom developed modules. The test plan should include topics such as scope, test strategy, responsibilities, test preparation and, test procedures (including deviation management).

8.4.3.2 Development of unit and integration test plans may take anywhere from two to ten days or more excluding review and approval cycles.

8.5 *Phase 4—Build/Configure:*

8.5.1 *System Configuration and Development:*

8.5.1.1 In this phase the system software is configured, interfaces are configured or developed, any required customizations coded, hardware assembled and the components are integrated into a complete system based on the design specifications.

8.5.1.2 Configuration involves using the administrative and master data functions provided in the commercial software to prepare the system for use. These activities are normally

accomplished by setting values, selection of check boxes, filling-in forms or tables of values or other similar activities that define how base system functions perform. Commonly configured items include defining sample types, sample registration forms or result entry forms.

8.5.1.3 Customization is most commonly defined as software development activities that involve writing procedural programming code in the LIMS vendor's proprietary development language or other third party languages. Furthermore, customization may alter the way base system functions were intended to be used or add new functions not present in the vendor's base product. Customizations may be built to improve efficiency or quality by automating steps or actions normally executed by users in the system. Providing dynamic update of customer limits in the LIMS database through integration with a separate client database system to determine the appropriate customers for manufactured products would be an example of customization.

8.5.1.4 Configuration can normally be accomplished quickly and effectively through the administrative functions provided. Coming to a decision on how to best configure functionality make take several days to several weeks of working through options and assessing their impact on other system functions. Customization can take several days to several months to design, code and test depending on the complexity of the required functionality.

8.5.1.5 There are many methodologies for system configuration and development. [ISO/IEC 12207](#) can adequately explain the different approaches to software development. The approaches ranges from the traditional waterfall to phased evolutionary methodologies.

8.5.1.6 The system configuration and development may use a waterfall approach to development where the system design is fully developed and approved before proceeding on to system build activities. Rapid prototyping characterized by iterations of the design, build, and test activities has proven an effective methodology and is frequently used today. The prototyping is typically planned to be accomplished in three to five iterations with each iteration taking anywhere from two weeks to two months or more.

8.5.1.7 In prototyping, an initial high-level design is developed which should be used to guide the prototyping of the system. Through iterations of design, build and test, further details regarding requirements and system design are elaborated and documented in a system design specification. The detailed configuration parameters and detailed design specifications are updated simultaneously but may not be formally approved until the close of the final iteration.

8.5.1.8 Requirements are divided into logical groupings to form the development or prototype units. Requirements from the requirements specification should be met by configuration of the LIMS or custom code where necessary. Custom code modules should be developed using quality development standards. This includes establishing common standards for commenting code, defining variables, any naming conventions for variables or modules, indenting, and so forth. Standards should be established and used by the entire development team. A technical team lead or a senior developer should review code

for adherence to these standards. Standards should address the following topics: modules should have descriptive program headers including information such as revision date, (if appropriate, build number), programmer name, system name (revision number if appropriate), module name, module purpose, any required syntaxes for external references (inputs or outputs) to/from the module and any unique considerations for integration. A change history (names, dates, brief descriptions of changes) should be included in program headers. Code should be indented and include sufficient comments to enable other programmers to easily understand work with the code.

8.5.1.9 At the end of the each build iteration, before the prototype review, the code and development for the current prototype scope is frozen. The functionality in the prototype should be demonstrated to the users for their review. Any change/correction requests should be documented using a change control process, where requests are analyzed and submitted to the development team for inclusion in the final system build.

8.5.1.10 Development of system configuration and customizations may take anywhere from twenty-five days to several months or more including prototyping time.

8.5.2 *Unit, Integration/System Testing:*

8.5.2.1 At the conclusion of system configuration or prototyping, the final system may be built and undergo Unit and Integration Testing prior to the start of formal validation in regulated environments. Unit and Integration Testing is performed in the development environment by the development team. Each individual development unit is tested separately against defined design parameters in the system design specification. Integration testing is also performed by the development team on functionality provided by a cohesive group of units. In this phase, the development team logically combines individual development units and tests the resulting system as an integral unit. Integration testing will also test any interfaces between the LIMS and other laboratory or enterprise systems.

8.5.2.2 Unit and integration test scripts should be developed by the development team and should be reviewed and approved prior to execution. Having test scripts for a particular development unit/module developed and executed by a developer or test team member who did not develop the code for that unit/module is very effective in identifying errors and helping produce a robust system. Any anomalies encountered during testing should be documented and change control forms used to document software defects, proposed resolutions and then the failed test re-executed.

8.6 *Phase 5—Test/Commission:*

8.6.1 *Hardware & Software Installation/Qualification:*

8.6.1.1 *Validation*—The validation of LIMS is a mandatory step for regulated industries. Specific validation requirements exist for industries regulated by the FDA, EPA, and NRC. Properly managed execution of validation principles that represent the use of best engineering practices can save time, resources, materials and expenses in the long run. Traditional validation methodologies such as GAMP can add three to twelve months to the implementation time of a LIMS. The application of intended use methodologies such as C3Q or a risk-based approach can reduce time and cost within the

regulated framework. Documentation plays an important role in the validation process for LIMS. See Guide E 2066, IEE 829, IEEE 1012, and IEEE 1028.

8.6.1.2 *Documentation*—Documentation is critical to the operation of a LIMS. Documentation includes manuals supplied by the vendor and user-developed documents. Examples of vendor-supplied documentation include manuals, technical reference manuals, validation manuals; QC documentation and vendor staff curriculum vitae. User-developed documents include all standard operating procedures (SOPs), training documents, change control forms, definitions, acceptance-testing records, problem report logs, backup and recovery logs, audit reports, and security records. See IEEE 1063.

8.6.1.3 Standard operating procedures and system documentation, as a requirement for a holistic approach to validation, standard operating procedures (SOP) are necessary such as, but not limited to: Back-up and recovery SOP (include testing of systems and files on a regular basis); Use SOP defining how the user is to utilize the system and what type of information is being entered into the system as well as the data sources; system administration SOP defining governance of tasks such as assignment of security within the software, system security, logical security and physical security; and change control SOP defining not only what steps are necessary to make changes to the system but also include risk assessment of the changes and the impact to the system.

8.6.1.4 *LIMS Staffing Requirements*—Staffing requirements will vary depending on the size of the laboratory, number of users supported and the complexity of the LIMS implementation. Additional resources will be required during the Initial LIMS Implementation for analysis, design, configuration and testing tasks. A resource will also need to be dedicated to system administration and support during the operate and maintain phase. This resource will not need to be full time depending on the maturity of the LIMS. Many laboratories engage resources in ongoing LIMS support providing enhancements including additional functionality and reports. The cost of these resources is supported by the benefits achieved from the additional functionality. LIMS support can be provided either by laboratory or information technology staff. The ideal candidate combines both laboratory and computer experience. Many companies have been successful in retraining existing laboratory personnel to acquire new computer skills. LIMS support can also be outsourced.

8.6.1.5 *Training*—End user and system manager training is critical to successful LIMS and should be given highest priority and continued support. Although system management training is usually comprehensive, care must be given to provide sufficient end user training to avoid continual telephone or written queries to the laboratory. Training needs to cover all aspects of LIMS operation from user training on how to perform sample registration, enter results and report results, to system manager training on how to maintain complex computer systems. User qualification testing is becoming standard for regulated laboratories. Training documents maintained for each user can include personnel backgrounds, education, qualifications, job experience, job descriptions, and formal

testing of specific LIMS functions. Training certification can/should be maintained in LIMS.

8.6.1.6 Validation/Acceptance Testing-Validation—The validation of LIMS is a mandatory step for regulated industries. Specific validation requirements exist for industries regulated by the local and federal agencies and certifying bodies. Validation of LIMS can add three to twelve months to the implementation time. Documentation plays an important role in the validation process for LIMS. See Guide E 2066, IEE 829, IEEE 1012, and IEEE 1028.

8.6.1.7 Data Migration—Data migration requires a carefully thought out plan to assess the transfer of data and meta-data from one system to another. The plan should account for synchronization of field types, field length, field mapping and manual verification with a sampling schedule. Migrated data can also be considered part of static data loading (See next section). Migration of one system to another not only affords the opportunity for data migration but also the chance for driving process improvement.

8.6.1.8 LIMS are capable of maintaining information a broad range of business and laboratory data required for the effective operation of the laboratory. LIMS contains data that not only reflects the current operation state of the laboratory but also historical information on past performance and events. Where LIMS is introduced into an environment where a LIMS system has not previously been deployed, much of the information typically managed in the LIMS will exist in the form of paper documentation or disparate electronic sources such as documents, spreadsheets or specialized databases. When implementing the LIMS some of this data will provide the basis for configuring the LIMS system. Static data are always loaded into the LIMS as part of the LIMS deployment lifecycle. The decision on how to deal with historical dynamic data should be evaluated on the basis of risk. Appropriate strategies for dealing with this data include migration, preservation and archival. Where an existing LIMS is being replaced with a new solution, it may be possible to migrate data from the source LIMS to the new target LIMS deployment. In such cases the following steps should be considered: source data analysis, target database configuration, source to target mapping, data migration workflow planning, migration piloting and process validation, data migration and data verification. LIMS often provide embedded or commercial to assist with data load and data migration processes.

8.6.2 Static (Master) Data Loading:

8.6.2.1 Loading of test, calculation, specification, and other static (master) information—The loading of an individual laboratory's tests, calculations, specifications, and other static information into the LIMS database is usually the most time consuming step in implementing a LIMS. A large laboratory with hundreds of tests, calculations, and specifications can spend 6 to 24 plus months on entering and verifying master data. Smaller laboratories with fewer tests, calculations, and specifications can reduce the implementation time to one to six months. The ability to migrate data using automation (if it is possible) can greatly reduce costs in terms of time, transcription errors and verification. (See section on data migration). This area of planning is consistently the least clearly under-

stood or planned area in LIMS implementation. The failure to clearly quantify the costs and time associated with this single LIMS implementation phase can place the entire project at risk. The total cost in person hours required to enter the test, calculation, and specification information can exceed the total cost of hardware and software. Detailed planning and prototyping is recommended to maximize efficiency in this area. Research and contract labs may have less work than a large QA laboratory. Each laboratory needs to address this task on a case-by-case basis.

8.6.3 Rollout:

8.6.3.1 Rollout can occur in the big-bang or phase approaches. For smaller systems the big bang works well since the number of users are manageable. For larger installations a phased approach is the way to go since the limiting step is training resources. Other considerations are system performance, initial end user acceptance, system tuning and tweaking. System performance contributes greatly to the initial reactions for end user acceptance. If the end users find that the system is sluggish they will be unhappy and slow to adopt the technology. This would make the system administrator's job more difficult. Phasing facilitates managing the rollout. User surveys and positive posting of the results can contribute to better acceptance. This will also help the system administrators in glean information for tweaking.

8.6.3.2 Rollout also requires planning for future issues with the system, including tracking those issues, escalation to vendor customer support, request for bug fixes/ enhancements (for both the project and core product), system restores, and data restores.

8.7 Phase 6—Operation and Maintenance—Operation and maintenance tasks include data backup, data recovery, various database management and optimization tasks, user account maintenance, and general administration such as service contracts administration and software/hardware maintenance and upgrade.

8.7.1 LIMS Staff and Organizational Placement—A laboratory with a staff of around 50 will require a minimum of one full time person dedicated to the maintenance of the LIMS. Larger laboratories can have two to five full time staff supporting LIMS and laboratory automation. The LIMS staff generally supports lab automation including LIMS, data acquisition, and robotics. Organizations with data processing departments must decide where to locate the LIMS support staff, in the laboratory organization, or in the data processing organization. Factors to be taken into account, when deciding how best to support the LIMS, should include the accessibility and the responsiveness of the LIMS support staff to the laboratory staff. Small laboratories may absorb the LIMS staff functions with the existing laboratory staff. The computer and system skills required of the LIMS staff vary with the technology employed. Systems implemented with mainframes or mini-computers generally required staff resources with skills supporting those architectures. The ideal candidates for LIMS staff include personnel with both laboratory and computer experience. Finding suitable candidates with both laboratory and

computer experience can be difficult. Laboratories have been successful in retraining existing laboratory personnel to acquire new computer skills.

8.7.2 Security—Policies and procedures need to be established to document the users' access to data, and privileges to update, insert, and delete data. LIMS privileges should be assigned in terms of business needs, assuring data integrity, QA/QC considerations, business ethics, and customer confidence in privacy and safety of data. A procedure to document changes in users' privileges should be established. Procedure should take into account users' promotions, terminations, and demotions.

8.7.3 Backup—A backup policy needs to be established. Policy needs to contain type of backup(s) and frequencies of backup(s). Careful consideration should be given to the tolerance for data loss.

8.7.4 Disaster Recovery—A disaster recovery procedure should be established. Disaster recovery exercises should be conducted at a specified frequency.

8.7.5 Data Archive—The archive of LIMS data are periodically performed to manage system storage space and performance.

8.7.6 System Administration—The normal operation of a LIMS includes the routine sample registration, result entry, result verification, and reporting. Data integrity checking is a continuous task. Special system software, audit trails, and LIMS reports are used to monitor the fidelity of LIMS data and information. New instruments are connected to the LIMS for transferring information. Links to external systems are maintained and serviced. Preventive maintenance tasks are performed per a predefined schedule. Repairs are conducted on failed hardware units. Software support is conducted with the LIMS vendor using voice, FAX, mail, and modem support.

8.7.7 Regulatory Requirements—Environmental Laboratories should adhere to EPA's Good Automated Laboratory Practices (GALP). Laboratories falling under the FDA's regulations should be compliant with FDA's Good Laboratory Practice and Good Manufacturing Practice regulations and guidelines.

8.7.8 Maintenance and Support—Commercial LIMS generally have maintenance agreements and services that cover technical support with varying degrees of service. The service agreements can include written or implied provisions for software upgrades and training, and clear definitions of both

user and vendor support expectations for the life of the arrangement. The service agreement should spell out how disagreements over service will be mediated, and should be made a part of the contract with the LIMS vendor.

8.7.9 Change Control—Change control/configuration management plays an important role during LIMS operations. Changes in hardware, software, laboratory staff, and laboratory environment need to be carefully monitored and controlled. The vendor generally updates the LIMS software periodically. The LIMS support staff can also customize the LIMS software. LIMS software updates/customizations need to be tested/validated prior to live use of the new software. Ongoing training is needed to keep existing LIMS users current with new features and to train new LIMS users. Procedures for LIMS change control need to be in place at the start of implementation. Change control procedures should define persons authorized to approve changes (hardware and software). Standard forms should be developed to track and manage changes. Alternatively, commercially available change control software can be purchased. Customizations to the code should be documented and approved by a formal process. Customizations should include a scope of work detailing the business needs for the customization, the business rules the code should be compliant with detailed functional requirements. Information tracked during changes should include requirements to be met before approval of changes, revision numbers for all the code undergoing change, responsibilities for documenting testing, approving of changed versions, and moving changed versions to the production environment.

8.8 Phase 7—System Retirement (Replacement, Archive):

8.8.1 Data Archive or Migration:

8.8.1.1 Retirement or Replacement (or both) of a LIMS—Planning for the replacement of LIMS should begin early in the LIMS life cycle. Technology (software and hardware) changes very rapidly. The technology cycle is often shorter than the typical LIMS implementation cycle. The ease of changing from one LIMS to another is very important with a short technology cycle. Issues include how to retrieve, edit, and report on LIMS data collected from an older LIMS. Questions to be addressed include: do you convert the old data (recommended), maintain old hardware (high risk approach over mid to long time periods) to retrieve data on a limited basis, or dump all data to a third party system for archive?

APPENDIX

(Nonmandatory Information)

X1. LIMS FUNCTIONAL REQUIREMENTS CHECK LIST

X1.1 This LIMS functional requirements check List is provided to users of the ASTM LIMS Guide as they consider a LIMS purchase, upgrade or development initiative. While comprehensive in scope, it is important that the guide is used as guidance as they collect and document their own functional, technical, security and business requirements to address their specific project needs.

X1.2 A well-conceived and executed requirements process that encompasses a broad group of stakeholders who will interface with the LIMS is a crucial element of a successful LIMS project. Often the group of assembled stakeholders will reach far beyond laboratory staff and include quality, regulatory, production, engineering and management representatives. If the requirements gathering process, which typically consist of reviewing existing documentation, interviews with stakeholders and joint application design (JAD) sessions do not generate the required level of granularity for each requirement type, it is unlikely that the LIMS initiative will be successful.

X1.3 The LIMS functional requirements check List is set up as a spreadsheet. The spreadsheet has five columns and each column is described below.

X1.4 The first column is the cross ref. No. This column includes a cross-reference code to [Fig. 2](#). The cross-reference numbers are used throughout this document and should be used to obtain detailed information on meaning of each requirement as described in this standard guide.

X1.5 The second column “LIMS Function,” is organized to include primary LIMS functions including: work/process flows, data flows including verification, validation and release as well as LIMS product life cycle management considerations such as archive and retirement. Again, typically these elements would be further decomposed and customized for the projects specific requirements. Criteria to be considered while performing vendor and cost evaluation are also included.

X1.6 The third column, “Criticality,” indicates criticality of

each element is provided in the function list. In this case, criticality refers to the business needs for a particular function in the scorecard. For example criticality can be expressed as follows:

X1.6.1 High, medium, low

X1.6.2 Show-stopper, must-have, like to have

X1.6.3 Critical, important, wish

X1.7 It is important that while requirements are being collected, organized and segmented, they are ranked to reflect criticality as most LIMS will not provide all desired functionality out of the box or even after extensive configuration or customization. The application of a criticality factor to each function will assist in the product evaluation. For example, if a potential product solution cannot provide a high or required function, it will typically be eliminated from future consideration. If the non-conformance is on a low or “wish list” function, the user may elect to choose that product, but recognize the limitation identified during a gap analysis.

X1.8 The fourth column “Scope” is provided to address the project need in determining the functional areas the LIMS project will address. Input is typically limited to a simple Boolean value, that is, Y/N, T/F, 0/1, and so forth.

X1.9 The fifth column “Timing” is provided to capture the phase of the LIMS implementation where this functionality will be required. Complex LIMS systems are generally phased in with certain basic LIMS functions implemented in the early phases with additional functions added in subsequent phases. The implementing organization could define the rough phasing up front and indicate on the spreadsheet what phase each LIMS function is required.

X1.10 ASTM encourages users of the LIMS Guide to utilize this feature/function list as a resource as they begin to collect, segment, prioritize and document the functional, technical, security and business requirements of their specific LIMS initiative.

TABLE X1.1 LIMS Functional Requirements Check List

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
LC-1.1	Registration			
LC-1.1.1	Automated sample registration			
LC-1.1.1.1	Scheduled sample registration - calendar based			
LC-1.1.1.2	Scheduled sample registration - triggered event			
LC-1.1.1.3	Stability Sample registration			
LC-1.1.1.4	Pre-sample registration samples			
LC-1.1.1.5	Workflow triggered sample registration			
LC-1.1.1.6	Sample registration from external systems or files			
LC-1.1.1.7	Unique LIMS number automatically assigned to each sample			
LC-1.1.1.8	Sample registration reports			
LC-1.1.2	Manual sample registration			
LC-1.1.2.1	Single Sample registration			
LC-1.1.2.2	Using templates			
LC-1.1.2.3	Ad hoc			
LC-1.1.2.4	Multi-sample Registration			
LC-1.1.2.5	Using templates			
LC-1.1.2.6	Ad hoc			
LC-1.1.2.7	Sample registration reports			
LC-1.1.2.8	Modify test assigned to samples during sample registration			
LC-1.1.2.9	Unique LIMS number automatically assigned to each sample			
LC-1.1.2.10	Ad hoc sample registration and test assignments			
LC-1.1.3	Apply test group			
LC-1.1.3.1	Group or Batch sample registration			
LC-1.1.3.2	Sample registration from external systems or files			
LC-1.1.3.3	User definable sample registration methods			
LC-1.1.3.4	User definable sample registration screens			
LC-1.1.3.5	Ease of sample registration			
LC-1.1.4	Apply test specification			
LC-1.1.4.1	Templates to allow limits in tests to be pre-designated			
LC-1.1.4.2	Add or delete tests or profiles from logged in sample			
LC-1.1.4.3	Sample registration from external systems or files			
LC-1.1.4.4	Register sample receipt for pre-logged sample			
LC-1.1.5	Labels/Bar-coding			
LC-1.1.5.1	Print container label (with or without barcodes)			
LC-1.1.5.2	Print container requirements report from schedule			
LC-1.1.5.3	Print labels from logged in samples (with bar codes)			
LC-1.1.5.4	Print labels from schedule			
LC-1.1.5.5	Print sample receipt			
LC-1.1.5.6	Print sampling route list from schedule			
LC-1.1.5.7	User definable label formats			
LC-1.1.5.8	Ease of label functions			
LC-1.2	Sample Management			
LC-1.2.1	Receive Samples			
LC-1.2.2	Sample Distribution/Chain of Custody			
LC-1.2.2.1	Chain of custody			
LC-1.2.2.2	Sample routing			
LC-1.2.2.3	Sample storage and retrieval			
LC-1.2.2.4	Sample storage inventory management			
LC-1.2.2.5	Sample disposition /disposition management			
LC-1.2.2.6	On-line access to sample distribution, storage and safety information			
LC-1.2.3	Reporting			
LC-1.2.3.1	Section/Analyst Work lists: Print workgroup and analyst backlog			
LC-1.2.3.2	Specialized sample group worklist: Print backlog			
LC-1.2.3.3	Distribution Lists			
LC-1.3	Core Laboratory Workflow			
LC-1.3.1	Assigning Analyst			
LC-1.3.1.1	Select tests or samples			
LC-1.3.1.2	User definable work assignment methods			
LC-1.3.1.3	Analyst work lists (printed or electronic)			
LC-1.3.1.4	Distribution Lists			
LC-1.3.1.5	Chain of custody			
LC-1.3.1.6	Sample routing			
LC-1.3.1.7	Sample storage and retrieval			
LC-1.3.1.8	Analyst certification check			
LC-1.3.2	Assigning Instrumentation			
LC-1.3.2.1	Select tasks			
LC-1.3.2.2	Instrument sequence or control file generation			
LC-1.3.2.3	(Tray loading/transmit sequence to file)			
LC-1.3.2.4	Calibration and maintenance check			
LC-1.3.3	Manual Data Entry			
LC-1.3.3.1	Single sample by sample/test			

TABLE X1.1 *Continued*

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
LC-1.3.3.2	Multi sample by work list/ test backlog			
LC-1.3.3.3	Worksheet single/multiple sample data entry			
LC-1.3.3.4	User definable result entry method			
LC-1.3.3.5	Spreadsheet or auto-entry limit			
LC-1.3.3.6	Sample storage inventory management			
LC-1.3.3.7	Sample disposition /disposition management			
LC-1.3.3.8	On-line access to sample distribution, storage and safety information			
LC-1.3.4	Automated Data Entry			
LC-1.3.4.1	RS 232 Instruments			
LC-1.3.4.2	Acquisitions systems			
LC-1.3.4.3	Instrument control (Bi-directional)			
LC-1.3.4.4	Auto sampler control			
LC-1.3.4.5	Robotic Systems			
LC-1.3.4.6	Review auto entered results			
LC-1.3.4.7	File Transfers			
LC-1.3.4.8	User definable instruments import			
LC-1.3.4.9	Enter results from samples sent out and foreign systems			
LC-1.4	Review			
LC-1.4.1	Data Verification			
LC-1.4.1.1	Peer, Supervisory, QC Review/Verify tests			
LC-1.4.1.2	Reschedule a test, sample			
LC-1.4.1.3	Approve samples			
LC-1.4.1.4	Communicate status to external systems			
LC-1.4.1.5	User definable review and approval methods			
LC-1.4.1.6	Data Edit /Correction			
LC-1.4.1.6.1	Edit Sample information			
LC-1.4.1.6.2	Edit test results			
LC-1.4.1.7	Missing specs			
LC-1.4.1.8	Approximate specification checking<,>			
LC-1.4.1.9	User definable specs checking			
LC-1.4.1.10	Ad-hoc spec definition post sample registration			
LC-1.4.1.11	Specs based on test results			
LC-1.4.1.12	Warning to user for out of spec (audible, screen message, color, flag)			
LC-1.4.1.13	Custom user defined algorithm for specification checking			
LC-1.4.1.14	Reviewing and Approving Results			
LC-1.4.1.14.1	Review/Verify tests			
LC-1.4.1.14.2	Reschedule a test, sample			
LC-1.4.1.14.3	Approve samples			
LC-1.4.1.14.4	Communicate status to external systems			
LC-1.4.1.14.5	User definable review and approval methods			
LC-1.4.2	Data Validation			
LC-1.4.2.1	Field length			
LC-1.4.2.2	Data type			
LC-1.4.2.3	Data format			
LC-1.4.2.4	Calculations			
LC-1.4.2.4.1	Inter and Intra-test			
LC-1.4.2.4.2	Inter and Intra sample			
LC-1.4.2.4.3	Descriptive Statistics			
LC-1.4.2.4.4	Advanced Math Functions			
LC-1.4.2.4.5	User defined functions			
LC-1.4.2.4.6	Links to prior results			
LC-1.4.2.4.7	Trigger/Event functions			
LC-1.4.2.4.8	Library to math subroutines			
LC-1.4.2.4.9	Sample preparation factors			
LC-1.4.2.4.10	Linear Calibration			
LC-1.4.3	Workflow triggers			
LC-1.4.4	ERES Compliance			
LC-1.4.5	Data Limit Levels: One, two, three+			
LC-1.4.6	Limit of Instrument detection issues			
LC-1.5	Results Release			
LC-1.5.1	Individual test results			
LC-1.5.2	Individual sample results			
LC-1.5.3	Batch/ Lot results			
LC-1.6	Reporting			
LC-1.6.1	Ad hoc reports capability			
LC-1.6.2	Pre-formatted Certificate of Analysis			
LC-1.6.3	Business Reports			
LC-1.6.3.1	Single sample			
LC-1.6.3.2	Sample status reports			
LC-1.6.3.3	Group analysis reports			

TABLE X1.1 *Continued*

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
LC-1.6.3.4	Certificate of analysis report			
LC-1.6.3.5	Graphical results reports			
LC-1.6.3.6	User definable/Ad-hoc reports			
LC-1.6.3.7	3rd party reporting tool interface			
LC-1.6.3.8	Workload reports			
LC-1.6.3.9	Overdue reports			
LC-1.6.3.10	Instrument Loading Reports			
LC-1.6.3.10.1	Tests performed on instruments by operators			
LC-1.6.3.10.2	Tests by instrument for inst utilization and PM			
LC-1.6.3.11	Instrument Calibration management			
LC-1.6.3.12	Personal loading reports			
LC-1.6.3.13	Accounting reports			
LC-1.6.3.13.1	Cost allocation report: total costs allocated by account			
LC-1.6.3.13.2	Performance invoices			
LC-1.6.3.13.3	Cumulative charges per account			
LC-1.6.3.14	Capability catalog			
LC-1.6.4	Regulatory Reports			
LC-1.6.4.1	Audit trail reports			
LC-1.6.4.2	Incident Reporting			
LC-1.6.4.3	Electronic data deliverables			
LC-1.6.4.4	Chain of custody reports			
LC-1.6.4.5	Quality Assurance reports			
LC-1.6.4.6	Transaction log listing and maintenance			
LC-1.6.4.7	Inventory management			
LC-1.6.4.8	Instrument calibration reports			
LC-1.6.4.9	Exception reports			
LC-1.6.4.10	Reagent and standards inventory reports			
LC-2.3	Archival			
LC-2.3.1	Data/ meta data			
LC-2.3.2	Configuration/Custom Code			
LC-2.3.3	Data Restore from Archival			
LC-2.3.4	Disposal			
LC-2.3.5	Use: Historical Trending and Regulatory Support			
LC-2.1	Retirement			
LC-2.1.1	Planning			
LC-2.1.2	Data preparation			
LC-2.1.3	Software preparation			
LC-2.1.4	Hardware preparation			
SM	System Reliability and Maintenance Requirements			
SM-1	Software maintenance & updates			
	Frequency of updates			
	Install new release in parallel for testing			
	Migration of data to new release			
	Documentation on software update			
	Ease of update process			
SM-2	Issue Management			
	End-user issues			
	Escalation to vendor support			
SM-3	Training			
	End-user			
	Power user			
	Management			
SM-4	Change Control			
	Code change			
	Static table and specification			
	Test systems for development and change testing			
SM-5	Archival, Backup and Recovery			
	Periodic data archival			
	Periodic data backup			
	Periodic data restore from archival on test system			
	Periodic data restore from backup media to test system			
SM-6	Audit Trails			
	Result level edits			
	Sample level edits			
	LIMS database structure edits			
	Test Structures			
	Specifications			
	IMS methods			
SC	IMS Life Cycle Phases			
SC-1	Project Initiation/Requirements Analysis			

TABLE X1.1 *Continued*

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
SC-1.1	Database structure			
SC-1.1.1	lat File			
SC-1.1.2	Relational			
SC-1.1.3	Client Server/Distributed database			
SC-1.1.4	Match to current information structure/relationships			
SC-1.1.5	Match to current laboratory material/test/ specification structure			
SC-1.1.6	LIMS sample/result data base structure: Access to fit laboratory			
SC-1.1.6.1	Sample, result, material, profiles, test, result, and specification structure			
SC-1.1.6.2	Relationship between samples and tests			
SC-1.2	Database tools			
SC-1.2.1	User definable tables, fields, indexes, field expressions			
SC-1.2.2	Definable field authorities by data type, category, group, user			
SC-1.2.3	Import/Export LIMS modules			
SC-1.2.4	Automatic restructure of old data into new structure			
SC-1.3	Configuration Tools			
SC-1.3.1	Material definition			
SC-1.3.2	Test definition			
SC-1.3.2.1	Tests tied to specific materials			
SC-1.3.2.2	Standard tests			
SC-1.3.2.3	Ad-hoc test definition during sample registration			
SC-1.3.2.4	Pass/Fail tests			
SC-1.3.2.5	Free Comment tests			
SC-1.3.2.6	Menu choice tests			
SC-1.3.2.7	Numeric tests			
SC-1.3.2.8	Numeric with calculations			
SC-1.3.3	Result/Observation definition			
SC-1.3.4	Algorithm definition			
SC-1.3.5	Profile definition			
SC-1.3.6	Time Study definition			
SC-1.3.7	Specification definition			
SC-1.3.8	Revision Control			
SC-1.3.9	Audit Trail changes			
SC-1.3.10	Configuration reports			
SC-1.4	Links to General Purpose Tools			
SC-1.4.1	Word processing			
SC-1.4.2	Spreadsheet			
SC-1.4.3	Pop-Up calculator			
SC-1.4.4	Statistical analysis			
SC-1.4.5	Graphic Presentation			
SC-2.0	Design			
SC-2.1	Provide functional requirements			
SC-2.2	Provide design specifications			
SC-2.3	Provide unit test cases			
SC-2.4	Provide integrated test cases			
SC-3	Build/ Configure			
SC-4	Data Loading			
SC-4.1	Calculations			
SC-4.1.1	Inter and Intra-test			
SC-4.1.2	Inter and Intra sample			
SC-4.1.3	Descriptive Statistics			
SC-4.1.4	Advanced Math Functions			
SC-4.1.5	User defined functions			
SC-4.1.6	Links to prior results			
SC-4.1.7	Trigger/Event functions			
SC-4.1.8	Library to math subroutines			
SC-4.1.9	Sample preparation factors			
SC-4.1.10	Linear Calibration			
SC-4.2	Quality Control Monitoring			
SC-4.2.1	QC Templates			
SC-4.2.2	Automatic generation of control charts			
SC-4.2.3	Automatic Trend Analysis			
SC-4.2.4	Automatic calculation of % accuracy of controls, spikes, recovered & difference of duplicates			
SC-4.3	System maintenance			
SC-4.3.1	System data maintenance			
SC-4.3.2	Result data export			
SC-4.4	Information access -Online			
SC-4.4.1	SOP's			
SC-4.4.2	Method development			
SC-4.4.3	Method validation			

TABLE X1.1 *Continued*

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
SC-4.4.4	Supporting technical information			
SC-4.4.5	Historical QA data			
SC-4.5	Change Control			
SC-4.5.1	Code change			
SC-4.5.2	Static table and specification			
SC-4.6	Security			
SC-4.6.1	By group, users, data type, field, function facility, network			
SC-4.6.2	By Electronic identification (passwords) and signature (Biometric verification)			
SC-4.6.3	Automatic terminal logout			
SC-4.7	Data feed from Migration processes			
SC-5	Qualification, Verification and Validation			
SC-5.1	Data integrity			
SC-5.1.1	Data integrity report			
SC-5.1.2	Transitional/referential integrity			
SC-5.1.3	Data recovery after fault			
SC-5.1.4	Data integrity during concurrent development			
SC-5.2	Performance			
SC-5.2.1	Benchmark tests on live production systems			
SC-5.2.2	Number of concurrent users during testing			
SC-5.2.3	Number of concurrent users supported			
SC-5.2.4	Overall performance Summary			
SC-5.3	Numerical Representation			
SC-5.3.1	Rounding Issues			
SC-5.3.1.1	User definable rounding rules			
SC-5.3.1.2	Odd - Even			
SC-5.3.1.3	Five and greater			
SC-5.3.4	Comparison operators (<,>,+,-)			
SC-5.3.5	Data types: Text, integer, floating point, fixed point, scientific notation, logical, date, integer			
VCS	Vendor Consideration and Selection			
VCS-01	Vendor Rating			
	Voice Support			
	Modem Support			
	Help Desk Support			
	Training Support			
	Installation Support			
	Documentation			
	Established Software Development Standards			
	Formal Change Control			
	Software Revision Control			
	Software portability			
	Access to source code			
	Quality and skills of staff			
	Quantity of support staff for customer support			
	Quantity of staff dedicated to R&D on future LIMS functions			
	Ability of vendor to apply new technology to LIMS product			
	Financial Stability			
	Number of LIMS installed			
	Number of years in the LIMS business			
	Meet Federal, Local or International regulatory requirements			
	Is there a service level agreement?			
VCS-02	Cost (note: cost can be ranked relative to other vendors or listed in actual amounts)			
	Hardware costs			
	Hardware costs of ownership (service, maintenance contracts)			
	Software costs			
	Software cost of ownership (support, updates)			
	Cable costs			
	Implementation costs			
	Training costs			
VCS-03	Vendor Reference List (obtain the following information from each customer)			
	Software Revision No.			
	Hardware Platform			
	No. of concurrent users			
	Samples per year			
	Industry			
	Years of Operation			
	Contact Name			
	Phone			

TABLE X1.1 *Continued*

Cross Ref. No.	LIMS Function	Criticality	Scope [Y/N]	Timing
	Can system administrators provide feedback on LIMS support from vendor			
	Can external customers provide input into future LIMS functions			
	Observe LIMS in operation			
	Access to source code through Escrow Agreement			
	Amount of custom coding required to meet requirements			
VCS-04	Return on Investment Estimate			

ASTM International takes no position respecting the validity of any patent rights asserted in connection with any item mentioned in this standard. Users of this standard are expressly advised that determination of the validity of any such patent rights, and the risk of infringement of such rights, are entirely their own responsibility.

This standard is subject to revision at any time by the responsible technical committee and must be reviewed every five years and if not revised, either reapproved or withdrawn. Your comments are invited either for revision of this standard or for additional standards and should be addressed to ASTM International Headquarters. Your comments will receive careful consideration at a meeting of the responsible technical committee, which you may attend. If you feel that your comments have not received a fair hearing you should make your views known to the ASTM Committee on Standards, at the address shown below.

This standard is copyrighted by ASTM International, 100 Barr Harbor Drive, PO Box C700, West Conshohocken, PA 19428-2959, United States. Individual reprints (single or multiple copies) of this standard may be obtained by contacting ASTM at the above address or at 610-832-9585 (phone), 610-832-9555 (fax), or service@astm.org (e-mail); or through the ASTM website (www.astm.org).