Basic Project Management Terms

A

Abstract class: A class that has no direct instances, but whose descendants may have direct instances.

Acceptance testing: The process whereby actual users test a completed information system, the end result of which is the users' acceptance of it.

Action stubs: That part of a decision table that lists the actions that result for a given set of conditions.

Activation: The time period during which an object performs an operation.

Actor: An external entity that interacts with the system (similar to an external entity in data flow diagramming).

Adaptive maintenance: Changes made to a system to evolve its functionality to changing business needs or technologies.

Aggregation: A *part-of* relationship between a component object and an aggregate object.

Alpha testing: User testing of a completed information system using simulated data.

Application independence: The separation of data and the definition of data from the applications that use these data.

Application server: A "middle-tier" software and hardware combination that lies between the Web server and the corporate network and systems.

Application software: Software designed to process data and support users in an organization. Examples of application software include spreadsheets, word processors, and database management systems.

Association: A relationship between object classes.

Association role: The end of an association where it connects to a class.

Associative entity: An entity type that associates the instances of one or more entity types and contains attributes that are peculiar to the relationship between those entity instances.

Attribute: A named property or characteristic of an entity that is of interest to the organization.

Audit trail: A record of the sequence of data entries and the date of those entries.

B

Balancing: The conservation of inputs and outputs to a data flow diagram process when that process is decomposed to a lower level.

Baseline modules: Software modules that have been tested, documented, and approved to be included in the most recently created version of a system.

Baseline Project Plan (BPP): The major outcome and deliverable from the project initiation and planning phase that contains an estimate of the project's scope, benefits, costs, risks, and resource requirements.

Behavior: Represents how an object acts and reacts.

Beta testing: User testing of a completed information system using real data in the real user environment.

Binary relationship: A relationship between instances of two entity types.

Boundary: The line that marks the inside and outside of a system, and that sets off one system from other systems in the organization.

Break-even analysis: A type of cost-benefit analysis to identify at what point (if ever) benefits equal costs.

Build routines: Guidelines that list the instructions to construct an executable system from the baseline source code.

Business case: A written report that outlines the justification for an information system. The report highlights economic benefits and costs and the technical and organizational feasibility of the proposed system.

Business process reengineering (BPR): The search for, and implementation of, radical change in business processes to achieve breakthrough improvements in products and services.

C

Calculated (or computed or derived) field: A field that can be derived from other database fields.

Candidate key: An attribute (or combination of attributes) that uniquely identifies each instance of an entity type.

Cardinality: The number of instances of entity B that can (or must) be associated with each instance of entity A.

Class diagram: Shows the static structure of an object-oriented model: the object classes, their internal structure, and the relationships in which they participate.

Closed-ended questions: Questions in interviews and on questionnaires that ask those responding to choose from among a set of specified responses.

Code generators: CASE tools that enable the automatic generation of program and database definition code directly from the design documents, diagrams, forms, and reports stored in the repository.

Cohesion: The extent to which a system or subsystem performs a single function.

Component: An irreducible part or aggregation of parts that makes up a system; also called a *subsystem.*

Component diagram: Shows the software components or modules and their dependencies.

Computer-aided software engineering (CASE): Software tools that provide automated support for some portion of the systems development process.

Conceptual data model: A detailed model that shows the overall structure of organizational data while being independent of any database management system or other implementation considerations.

Concrete class: A class that can have direct instances.

Condition stub: That part of a decision table that lists the conditions relevant to the decision.

Configuration management: The process of ensuring that only authorized changes are made to a system.

Constraint: A limit to what a system can accomplish.

Context diagram: A data flow diagram of the scope of an organizational system that shows the system boundaries, external entities that interact with the system, and the major information flows between the entities and the system.

Cookie crumbs: A technique for showing a user where they are in a Web site by placing a series of "tabs" on a Web page that show a user where they are and where they have been.

Corrective maintenance: Changes made to a system to repair flaws in its design, coding, or implementation.

Coupling: The extent to which subsystems depend on each other.

Critical path: The shortest time in which a project can be completed.

Critical path scheduling: A scheduling technique whose order and duration of a sequence of task activities directly affect the completion date of a project.

Cross life cycle CASE: CASE tools designed to support activities that occur *across* multiple phases of the systems development life cycle.

D

Data: Raw facts about people, objects, and events in an organization.

Database: A shared collection of logically related data designed to meet the information needs of multiple users in an organization.

Data flow: Data in motion, moving from one place in a system to another.

Data flow diagram (DFD): A graphic that illustrates the movement of data between external entities and the processes and data stores within a system.

Data-oriented approach: A strategy of information systems development that focuses on the ideal organization of data rather than on where and how they are used.

Data store: Data at rest, which may take the form of many different physical representations.

Data type: A coding scheme recognized by system software for representing organizational data.

Decision table: A matrix representation of the logic of a decision, which specifies the possible conditions for the decision and the resulting actions.

Decomposition: The process of breaking the description of a system down into small components; also called *functional decomposition*.

Default value: The value a field will assume unless an explicit value is entered for that field.

Degree: The number of entity types that participate in a relationship.

Deliverable: An end product in a phase of the SDLC.

Denormalization: The process of splitting or combining normalized relations into physical tables based on affinity of use of rows and fields.

Design strategy: A particular approach to developing an information system. It includes statements on the system's functionality, hardware and system software platform, and method for acquisition.

Desk checking: A testing technique in which the program code is sequentially executed manually by the reviewer.

DFD completeness: The extent to which all necessary components of a data flow diagram have been included and fully described.

DFD consistency: The extent to which information contained on one level of a set of nested data flow diagrams is also included on other levels.

Dialogue: The sequence of interaction between a user and a system.

Dialogue diagramming: A formal method for designing and representing human—?computer dialogues using box and line diagrams.

Discount rate: The interest rate used to compute the present value of future cash flows.

Disruptive technologies: Technologies that enable the breaking of long-held business rules that inhibit organizations from making radical business changes.

E

Economic feasibility: A process of identifying the financial benefits and costs associated with a development project.

Electronic commerce (EC): Internet-based communication to support day-to-day business activities.

Electronic data interchange (EDI): The use of telecommunications technologies to transfer business documents directly between organizations.

Electronic performance support system (EPSS): Component of a software package or application in which training and educational information is embedded. An EPSS may include a tutorial, expert system, and hypertext jumps to reference material.

Encapsulation: The technique of hiding the internal implementation details of an object from its external view.

End users: Non-information-system professionals in an organization. End users often request new or modified software applications, test and approve applications, and may serve on project teams as business experts.

Enterprise solutions software or enterprise resource planning (ERP) system: A system that integrates individual traditional business functions into a series of modules so that a single transaction occurs seamlessly within a single information system rather than in several separate systems.

Entity: A person, place, object, event, or concept in the user environment about which the organization wishes to maintain data.

Entity instance (instance): A single occurrence of an entity type.

Entity-relationship diagram (E-R diagram): A detailed, logical, and graphical representation of the entities, associations, and data elements for an organization or business area.

Entity type: A collection of entities that share common properties or characteristics.

Environment: Everything external to a system that interacts with the system.

Event: Something that takes place at a certain point in time; it is a noteworthy occurrence that triggers a state transition.

External documentation: System documentation that includes the outcome of structured diagramming techniques such as data flow and entity-relationship diagrams.

Extranet: Internet-based communication to support business-to-business activities.

Feasibility study: Determines if the information system makes sense for the organization from an economic and operational standpoint.

F

Field: The smallest unit of named application data recognized by system software.

File organization: A technique for physically arranging the records of a file.

Foreign key: An attribute that appears as a nonprimary key attribute in one relation and as a primary key attribute (or part of a primary key) in another relation.

Form: A business document that contains some predefined data and may include some areas where additional data are to be filled in. An instance of a form is typically based on one database record.

Formal system: The official way a system works as described in organizational documentation.

Form and report generators: CASE tools that support the creation of system forms and reports in order to prototype how systems will "look and feel" to users.

Functional dependency: A particular relationship between two attributes. For a given relation, attribute B is functionally dependent on attribute A if, for every valid value of A, that value of A uniquely determines the value of B. The functional dependence of B on A is represented by A'B.

G

Gantt chart: A graphical representation of a project that shows each task as a horizontal bar whose length is proportional to its time for completion.

Gap analysis: The process of discovering discrepancies between two or more sets of data flow diagrams or discrepancies within a single DFD.

H

Hashed file organization: The address for each row is determined using an algorithm.

Help desk: A single point of contact for all user inquiries and problems about a particular information system or for all users in a particular department.

Homonym: A single attribute name that is used for two or more different attributes.

I

I-CASE: An automated systems development environment that provides numerous tools to create diagrams, forms, and reports; provides analysis, reporting, and code generation facilities; and seamlessly shares and integrates data across and between tools.

Identifier: A candidate key that has been selected as the unique, identifying characteristic for an entity type.

Incremental commitment: A strategy in systems analysis and design in which the project is reviewed after each phase and continuation of the project is rejustified in each of these reviews.

Index: A table used to determine the location of rows in a file that satisfy some condition.

Indifferent condition: In a decision table, a condition whose value does not affect which actions are taken for two or more rules.

Informal system: The way a system actually works.

Information: Data that have been processed and presented in a form suitable for human interpretation, often with the purpose of revealing trends or patterns.

Information center: An organizational unit whose mission is to support users in exploiting information technology.

Information systems analysis and design: The process of developing and maintaining an information system.

Inspections: A testing technique in which participants examine program code for predictable language-specific errors.

Installation: The organizational process of changing over from the current information system to a new one.

Intangible benefit: A benefit derived from the creation of an information system that cannot be easily measured in dollars or with certainty.

Intangible cost: A cost associated with an information system that cannot be easily measured in terms of dollars or with certainty.

Integration testing: The process of bringing together all of the modules that a program comprises for testing purposes. Modules are typically integrated in a top-down, incremental fashion.

Interface: Point of contact where a system meets its environment or where subsystems meet each other.

Internal documentation: System documentation that is part of the program source code or is generated at compile time.

Internet: A large worldwide network of networks that use a common protocol to communicate with each other; a global computing network to support business-to-consumer electronic commerce.

Interrelated components: Dependence of one part of the system on one or more other system parts.

Intranet: Internet-based communication to support business activities within a single organization.

J

JAD session leader: The trained individual who plans and leads Joint Application Design sessions.

Joint Application Design (JAD): A structured process in which users, managers, and analysts work together for several days in a series of intensive meetings to specify or review system requirements.

K/L

Key business processes: The structured, measured set of activities designed to produce a specific output for a particular customer or market.

Legal and contractual feasibility: The process of assessing potential legal and contractual ramifications due to the construction of a system.

Level-0 diagram: A data flow diagram that represents a system's major processes, data flows, and data stores at a high level of detail.

Level-*n* diagram: A DFD that is the result of *n* nested decompositions of a series of subprocesses from a process on a level-0 diagram.

Lightweight graphics: The use of small simple images to allow a Web page to be more quickly displayed.

Lower CASE: CASE tools designed to support the systems implementation and operation phase of the systems development life cycle.

M

Maintenance: Changes made to a system to fix or enhance its functionality.

Mean time between failures (MTBF): A measurement of error occurrences that can be tracked over time to indicate the quality of a system.

Modularity: Dividing a system into chunks or modules of equal size.

Multiplicity: Indicates how many objects participate in a given relationship.

Multivalued attribute: An attribute that may take on more than one value for each entity instance.

N

Normalization: The process of converting complex data structures into simple, stable data structures.

Null value: A special field value, distinct from 0, blank, or any other value, that indicates that the value for the field is missing or otherwise unknown.

O

Object: An entity that has a well-defined role in the application domain and has state, behavior, and identity.

Object class: A set of objects that share a common structure and a common behavior.

Object diagram: A graph of instances that are compatible with a given class diagram.

One-time cost: A cost associated with project start-up and development, or system start-up.

Open-ended questions: Questions in interviews and on questionnaires that have no prespecified answers.

Operation: A function or a service that is provided by all the instances of a class.

Operational feasibility: The process of assessing the degree to which a proposed system solves business problems or takes advantage of business opportunities.

Outsourcing: The practice of turning-over responsibility of some to all of an organization's information systems applications and operations to an outside firm.

P

Participatory Design (PD): A systems development approach that originated in northern Europe in which users and the improvement of their work lives are the central focus.

Perfective maintenance: Changes made to a system to add new features or to improve performance.

PERT chart: A diagram that depicts project tasks and their interrelationships. PERT stands for Program Evaluation Review Technique.

Physical file: A named set of table rows stored in a contiguous section of secondary memory.

Physical table: A named set of rows and columns that specifies the fields in each row of the table.

Picture (or template): A pattern of codes that restricts the width and possible values for each position of a field.

Pointer: A field of data that can be used to locate a related field or row of data.

Political feasibility: The process of evaluating how key stakeholders within the organization view the proposed system.

Present value: The current value of a future cash flow.

Preventive maintenance: Changes made to a system to avoid possible future problems.

Primary key: An attribute whose value is unique across all occurrences of a relation.

Primitive DFD: The lowest level of decomposition for a data flow diagram.

Process: The work or actions performed on data so that they are transformed, stored, or distributed.

Processing logic: The steps by which data are transformed or moved and a description of the events that trigger these steps.

Process modeling: Graphically representing the processes that capture, manipulate, store, and distribute data between a system and its environment and among components within a system.

Process-oriented approach: An approach to developing information systems that focuses on how and when data are moved and changed.

Project: A planned undertaking of related activities to reach an objective that has a beginning and an end.

Project closedown: The final phase of the project management process, which focuses on bringing a project to an end.

Project execution: The third phase of the project management process, in which the plans created in the prior phases (project initiation and planning) are put into action.

Project initiation: The first phase of the project management process, in which activities are performed to assess the size, scope, and complexity of the project and to establish procedures to support later project activities.

Project management: A controlled process of initiating, planning, executing, and closing down a project.

Project manager: A systems analyst with a diverse set of skills--management, leadership, technical, conflict management, and customer relationship--who is responsible for initiating, planning, executing, and closing down a project.

Project planning: The second phase of the project management process, which focuses on defining clear, discrete activities and the work needed to complete each activity within a single project.

Project workbook: An on-line or hard-copy repository for all project correspondence, inputs, outputs, deliverables, procedures, and standards that is used for performing project audits, orientation of new team members, communication with management and customers, identifying future projects, and performing postproject reviews.

Prototyping: Building a scaled-down version of the desired information system.

Purpose: The overall goal or function of a system.

R

Rapid Application Development (RAD): Systems development methodology created to radically decrease the time needed to design and implement information systems. RAD relies on extensive user involvement, Joint Application Design sessions, prototyping, integrated CASE tools, and code generators.

Recurring costs: A cost resulting from the ongoing evolution and use of a system.

Recursive foreign key: A foreign key in a relation that references the primary key values of that same relation.

Referential integrity: An integrity constraint specifying that the value (or existence) of an attribute in one relation depends on the value (or existence) of an attribute in the same or another relation.

Relation: A named, two-dimensional table of data. Each relation consists of a set of named columns and an arbitrary number of unnamed rows.

Relational database model: Data represented as a set of related tables or relations.

Relationship: An association between the instances of one or more entity types that is of interest to the organization.

Repeating group: A set of two or more multivalued attributes that are logically related.

Report: A business document that contains only predefined data; it is a passive document used only for reading or viewing. A form typically contains data from many unrelated records or transactions.

Repository: A centralized database that contains all diagrams, forms and report definitions, data structure, data definitions, process flows and logic, and definitions of other organizational and system components; it provides a set of mechanisms and structures to achieve seamless data-to-tool and data-to-data integration.

Request for proposal (RFP): A document provided to vendors to ask them to propose hardware and system software that will meet the requirements of a new system.

Resources: Any person, group of people, piece of equipment, or material used in accomplishing an activity.

Rules: That part of a decision table that specifies which actions are to be followed for a given set of conditions.

S

Scalable: The ability to upgrade seamlessly the capabilities of the system through either hardware upgrades, software upgrades, or both.

Schedule feasibility: The process of assessing the degree to which the potential time frame and completion dates for all major activities within a project meet organizational deadlines and constraints for affecting change.

Scribe: The person who makes detailed notes of the happenings at a Joint Application Design session.

Secondary key: One or a combination of fields for which more than one row may have the same combination of values.

Second normal form (2NF): A relation is in second normal form if every nonprimary key attribute is functionally dependent on the whole primary key.

Sequence diagram: Depicts the interactions among objects during a certain period of time.

Sequential file organization: The rows in the file are stored in sequence according to a primary key value.

Simple message: A message that transfers control from the sender to the recipient without describing the details of the communication.

Slack time: The amount of time that an activity can be delayed without delaying the project.

Source/Sink: The origin and/or destination of data; also called *external entities.*

Statement of Work (SOW): A document prepared for the customer during project initiation and planning that describes what the project will deliver and outlines generally and at a high level all work required to complete the project.

State transition: Changes in the attributes of an object or in the links an object has with other objects.

Structured English: Modified form of the English language used to specify the logic of information system processes. Although there is no single standard, Structured English typically relies on action verbs and noun phrases and contains no adjectives or adverbs.

Stub testing: A technique used in testing modules, especially where modules are written and tested in a top-down fashion, where a few lines of code are used to substitute for subordinate modules.

Support: Providing ongoing educational and problem-solving assistance to information system users. Support material and jobs must be designed along with the associated information system.

Synchronous message: A type of message in which the caller has to wait for the receiving object to finish executing the called operation before it can resume execution itself.

Synonyms: Two different names that are used for the same attribute.

System: A group of interrelated procedures used for a business function, with an identifiable boundary, working together for some purpose.

System documentation: Detailed information about a system's design specifications, its internal workings, and its functionality.

System librarian: A person responsible for controlling the checking out and checking in of baseline modules when a system is being developed or maintained.

Systems analysis: Phase of the SDLC in which the current system is studied and alternative replacement systems are proposed.

Systems analyst: The organizational role most responsible for the analysis and design of information systems.

Systems design: Phase of the SDLC in which the system chosen for development in systems analysis is first described independent of any computer platform (logical design) and is then transformed into technology-specific details (physical design) from which all programming and system construction can be accomplished.

Systems development life cycle (SDLC): The series of steps used to mark the phases of development for an information system.

Systems development methodology: A standard process followed in an organization to conduct all the steps necessary to analyze, design, implement, and maintain information systems.

Systems implementation and operation: Final phase of the SDLC, in which the information system is coded, tested, and installed in the organization, and in which the information system is systematically repaired and improved.

Systems planning and selection: The first phase of the SDLC, in which an organization's total information system needs are analyzed and arranged, and in which a potential information systems project is identified and an argument for continuing or not continuing with the project is presented.

System testing: The bringing together of all the programs that a system comprises for testing purposes. Programs are typically integrated in a top-down, incremental fashion.

T

Tangible benefit: A benefit derived from the creation of an information system that can be measured in dollars and with certainty.

Tangible cost: A cost associated with an information system that can be easily measured in dollars and with certainty.

Technical feasibility: The process of assessing the development organization's ability to construct a proposed system.

Template-based HTML: Templates to display and process common attributes of higher-level, more abstract items.

Ternary relationship: A simultaneous relationship among instances of three entity types.

Third normal form (3NF): A relation is in third normal form if it is in second normal form and there are no functional (transitive) dependencies between two (or more) nonprimary key attributes.

Time value of money (TVM): The process of comparing present cash outlays to future expected returns.

U

Unary relationship (recursive relationship): A relationship between the instances of one entity type.

Unified Modeling Language (UML): A notation that allows the modeler to specify, visualize, and construct the artifacts of software systems, as well as business models.

Unit testing: Each module is tested alone in an attempt to discover any errors in its code.

Upper CASE: CASE tools designed to support the systems planning and selection, systems analysis, and systems design phases of the systems development life cycle.

Use case: A complete sequence of related actions initiated by an actor; it represents a specific way to use the system.

Use-case diagram: A diagram that depicts the use cases and actors for a system.

User documentation: Written or other visual information about an application system, how it works, and how to use it.

W

Walkthrough: A peer group review of any product created during the systems development process; also called a *structured walkthrough.*

Web server: A computer that is connected to the Internet and stores files written in HTML (hypertext markup language) that are publicly available through an Internet connection.

Well-structured relation (table): A relation that contains a minimum amount of redundancy and allows users to insert, modify, and delete the rows without errors or inconsistencies.

Work breakdown structure: The process of dividing the project into manageable tasks and logically ordering them to ensure a smooth evolution between tasks.