IEEE Std 1362™-1998 (R2007)
(Incorporates
IEEE Std 1362a-1998)

IEEE Guide for Information Technology—System Definition—Concept of Operations (ConOps) Document

Sponsor
Software Engineering Standards Committee
of the
IEEE Computer Society

Approved 19 March 1998
Reaffirmed 5 December 2007

IEEE-SA Standards Board

Abstract: The format and contents of a concept of operations (ConOps) document are described. A ConOps is a user-oriented document that describes system characteristics for a proposed system from the users' viewpoint. The ConOps document is used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements (for example, training, facilities, staffing, and maintenance). It is used to describe the user organization(s), mission(s), and organizational objectives from an integrated systems point of view.
Keywords: buyer, characteristics, concept of operation, concepts of operations document, ConOps,

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Print: ISBN 0-7381-0185-2 SH94615
PDF: ISBN 0-7381-1407-3 SS94615

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

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

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

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

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

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

Introduction

[This introduction is not a part of IEEE Std 1362-1998, IEEE Guide for Information Technology—System Definition—Concept of Operations (ConOps) Document.]

Purpose

This guide presents format and contents of a concept of operations (ConOps) document to be used when developing or modifying a software-intensive system. A software-intensive system is a system for which software is a major technical challenge and is perhaps the major factor that affects system schedule, cost, and risk. In the most general case, a software-intensive system is comprised of hardware, software, people, and manual procedures. To make this guide more readable, the term "system" will be used to mean a software-intensive system that includes elements to be developed or modified, in addition to software. The term "software system" will be used to mean a software-intensive system in which software is the only component to be developed or modified.

This guide does not specify the exact techniques to be used in developing the ConOps document, but it does provide approaches that might be used. Each organization that uses this guide should develop a set of practices and procedures to provide detailed guidance for preparing and updating ConOps documents. These detailed practices and procedures should take into account the environmental, organizational, and political factors that influence application of the guide.

The heart of the ConOps described in this guide is contained in Clauses 3 through 5.

— Clause 3 describes the existing system (manual or automated) that the user wants to replace;
— Clause 4 provides justification for a new or modified system and any restrictions on that system; and
— Clause 5 describes the proposed system.

The outlines for Clause 3 and Clause 5 are almost identical. This is not to say that the contents of the finished ConOps document will be identical. On the contrary, the contents should be very different. The outlines are the same to remind developers of the items that should be included and the actions to be taken.

Not all software projects are concerned with development of source code for a new software product. Some software projects consist of a feasibility study and definition of product requirements. Other projects terminate upon completion of product design or are only concerned with modifications to existing software products. Applicability of this guide is not limited to projects that develop operational versions of new products, nor is it limited by project size or scope. Small projects may require less formality than large projects, but all components of this guide should be addressed by every software project.

The ConOps approach provides an analysis activity and a document that bridges the gap between the user's needs and visions and the developer's technical specifications. In addition, the ConOps document provides the following:

— A means of describing a user's operational needs without becoming bogged down in detailed technical issues that shall be addressed during the systems analysis activity.
— A mechanism for documenting a system's characteristics and the user's operational needs in a manner that can be verified by the user without requiring any technical knowledge beyond that required to perform normal job functions.
— A place for users to state their desires, visions, and expectations without requiring the provision of quantified, testable specifications. For example, the users could express their need for a "highly reliable" system, and their reasons for that need, without having to produce a testable reliability requirement. [In this case, the user's need for "high reliability" might be stated in quantitative terms by the buyer prior to issuing a request for proposal (RFP), or it might be quantified by the developer during requirements analysis. In any case, it is the job of the buyer and/or the developer to quantify users' needs.]

— A mechanism for users and buyer(s) to express thoughts and concerns on possible solution strategies. In some cases, design constraints dictate particular approaches. In other cases, there may be a variety of acceptable solution strategies. The ConOps document allows users and buyer(s) to record design constraints, the rationale for those constraints, and to indicate the range of acceptable solution strategies.

Intended uses

This guide is intended for use in a variety of situations by a variety of users including the following:

— Acquirers using ISO/IEC 12207:1995, Information technology—Software life cycle processes, will find the current guide suitable for satisfying the requirements of 5.1.1.1:

"The acquirer begins the acquisition process by describing a concept or a need to acquire, develop, or enhance a system, software product or software service."

— Users who formerly applied MIL-STD-498, Software Development and Documentation, and related standards will find that the ConOps document described in this guide is very similar to the operational concept description (OCD) included in MIL-STD-498.

— Users of EIA/IEEE J-STD-016-1995, EIA/IEEE Interim Trial-Use Standard for Information Technology Software Life Cycle Processes Software Development Acquirer—Supplier Agreement will find that the ConOps document described in this guide is substantively identical to the OCD included in EIA/IEEE J-STD-016-1995.

— Other users will find this guide useful in facilitating communication among the various stakeholders in a project.

Software as part of a larger system

Software projects are sometimes parts of larger projects. In these cases, the software ConOps document may be a separate document or it may be merged into the system level ConOps document.

Overview

This guide contains four clauses. Clause 1 defines the scope of this guide. Clause 2 provides references to other IEEE standards that should be followed when applying this guide. Clause 3 provides definitions of terms that are used throughout the guide. Clause 4 contains an overview and a detailed specification of the ConOps document, including required components that should be included, and optional components that may be included in project plans based on this guide.

Responsible organization

Ideally, the ConOps document should be written by representatives of the user community. In practice, other individuals or organizations may write the ConOps (e.g., the buyer, a third party consultant, and/or the software developer). In these cases, it is essential that user representatives be involved in reviewing, revising, and approving the ConOps document. The primary goal for a ConOps document is to capture user needs, and to express those needs in the user's terminology.

Audience

This guide is intended for users and buyers of software systems, software developers, and other personnel who prepare and update operational requirements for software-intensive systems and monitor adherence to those requirements.

Evolution of plans

Developing the initial version of the ConOps document should be one of the first activities completed on a software project. As the project evolves, the nature of the work to be done and details of the work will be better understood. The ConOps document should be updated periodically to reflect the evolving situation. Thus, each version of the document should be placed under configuration control.

Terminology

This guide follows the 1996 edition of the IEEE Standards Style Manual. The terms *should, may, might,* and *suggest* are used to indicate actions that should be used to develop a good ConOps document but that are not mandatory. However, the authors of a ConOps document should consider using all aspects of this guide to insure a complete and effective document.

The ConOps document is sometimes called an operational concept document (OCD).

History

Use of a ConOps document was first documented in Lano, R. J., "A Structured Approach for Operational Concept Formulation," TRW SS-80-02, TRW Defense and Space Systems Group, Redondo Beach, CA, 1980. In 1992 the Software Systems Technical Committee of the American Institute of Aeronautics and Astronautics (AIAA), developed a standard for an OCD.

This ConOps guide originated in October 1993, as a Master of Science thesis at California State University, Sacramento, and was supported by the U.S. Office of Research and Development. It was accepted as MIL-STD-498, Data Item Description (DID), by the DoD-Std-2167A Harmonizing Working Group with few changes. MIL-STD-498-1995 became IEEE Std 1498-1995, which was redesignated J-STD-016-1995.

The IEEE Standards Board approved the project authorization request (PAR) for development of this guide in June 1993. The first draft was submitted to the Software Engineering Standards Committee (SESC) on 8 August 1995; it was returned on 1 November 1995 with a request that the guide be harmonized with certain other specified software engineering standards. The second draft was submitted to the SESC on 28 February 1996. This draft was balloted on 21 August 1996.

Participants

This guide was written by the IEEE Guide for a Concept of Operations Document Working Group, which is part of the IEEE Computer Society. The following three individuals are the authors of this guide:

Richard H. Thayer
Richard E. Fairley
Per Bjorke

Other individuals who supported the development of this guide are:

Jed Bartlett	Merlin Dorfman	Randy Paul
Boris I. Cogan	Rajko Milovanovic	Jane Radatz

The following persons were on the balloting committee:

Mikhail Auguston	Peter A. Berggren	Alan L. Bridges
Robert E. Barry	H. Ronald Berlack	Kathleen L. Briggs
Mordechai Ben-Menachem	Audrey C. Brewer	Thomas G. Callaghan

Stuart Ross Campbell
Leslie Chambers
Keith Chan
John P. Chihorek
S. V. Chiyyarath
Antonio M. Cicu
Theo Clarke
Darrell Cooksey
W. W. Geoff Cozens
Gregory T. Daich
Hillary Davidson
Neil Davis
Bostjan K. Derganc
Michael P. DeWalt
Dave Dikel
Charles Droz
John W. Fendrich
Julian Forster
Eva Freund
Juan Garbajosa-Sopena
Julio Gonzalez-Sanz
L. M. Gunther
John Harauz
Rob Harker
William Hefley
Manfred Hein
Mark Heinrich
Mark Henley
Umesh P. Hiriyannaiah
Fabrizio Imelio
George Jackelen
Vladan V. Javonovic
Frank V. Jorgensen
William S. Junk
George X. Kambic
David W. Kane
Judith S. Kerner
Robert J. Kierzyk
Motti Y. Klein
Dwayne L. Knirk
Shaye Koenig
Thomas M. Kurihara
J. Dennis Lawrence
Michael Lines
Dieter Look
David Maibor
Philip P. Mak
Tomoo Matsubara
Scott D. Matthews
Patrick McCray
Bret Michael
Alan Miller
Millard Allen Mobley
James W. Moore
Kartik C. Mujamdar
Mike Ottewill
Donald J. Pfeiffer
John G. Phippen
Peter T. Poon
Margaretha W. Price
Kenneth R. Ptack
Andrew P. Sage
Stephen R. Schach
Norman F. Schneidewind
Gregory D. Schumacher
Robert W. Shillato
Richard S. Sky
Alfred R. Sorkowitz
Donald W. Sova
Fred J. Strauss
Michael Surratt
Douglas H. Thiele
Booker Thomas
Patricia Trellue
Richard D. Tucker
Theodore J. Urbanowicz
Glenn D. Venables
Camille S. White-Partain
Charles D. Wilson
Paul R. Work
Weider D. Yu
Janusz Zalewski
Peter F. Zoll

The following individuals were part of the Life Cycle Data Harmonization working group for IEEE Std 1362a-1998:

Leonard L. Tripp, *Chair*

Edward Byrne
Paul R. Croll
Perry DeWeese
Robin Fralick
Marilyn Ginsberg-Finner
John Harauz
Mark Henley
Dennis Lawrence
David Maibor
Ray Milovanovic
James Moore
Timothy Niesen
Dennis Rilling
Terry Rout
Richard Schmidt
Norman F. Schneidewind
David Schultz
Basil Sherlund
Peter Voldner
Ronald Wade

The following persons were on the balloting committee for IEEE Std 1362a-1998:

Eduardo W. Bergamini
H. Ronald Berlack
Richard E. Biehl
Juris Borzovs
David W. Burnett
Michael Caldwell
Antonio M. Cicu
Francois Coallier
Virgil Lee Cooper
W. W. Geoff Cozens
Paul R. Croll
Geoffrey Darnton
Taz Daughtrey
Bostjan K. Derganc
Perry R. DeWeese
Leo Egan
Jonathan H. Fairclough
Richard E. Fairley
John W. Fendrich
Jay Forster
Kirby Fortenberry
Eva Freund
Roger U. Fujii
Marilyn Ginsberg-Finner
Julio Gonzalez-Sanz
Lewis Gray
L. M. Gunther
David A. Gustafson
John Harauz
Rob Harker
William Hefley
Debra Herrmann
Umesh P. Hiriyannaiah

David Johnson
Frank V. Jorgensen
William S. Junk
Ron S. Kenett
Judith S. Kerner
Robert J. Kierzyk
Thomas M. Kurihara
John B. Lane
J. Dennis Lawrence
Mary Leatherman
William M. Lively
Stan Magee
David Maibor
Robert A. Martin
Patrick D. McCray
James W. Moore
Pavol Navrat
Donald J. Ostrom
Lalit M. Patnaik
Mark Paulk
John G. Phippen
Alex Polack
Peter T. Poon
Kenneth R. Ptack
Larry K. Reed
Ann E. Reedy
Donald J. Reifer
Annette D. Reilly
Andrew P. Sage
Helmut Sandmayr
Stephen R. Schach
Norman F. Schneidewind
David J. Schultz
Robert W. Shillato
David M. Siefert
Lynn J. Simms
Carl A. Singer
Fred J. Strauss
Toru Takeshita
Richard H. Thayer
Douglas H. Thiele
Booker Thomas
Patricia Trellue
Glenn D. Venables
John W. Walz
Camille S. White-Partain
Scott A. Whitmire
P. A. Wolfgang
Paul R. Work
Janusz Zalewski
Geraldine Zimmerman
Peter F. Zoll

When the IEEE-SA Standards Board approved this standard on 19 March 1998, it had the following membership:

Richard J. Holleman, *Chair*
Donald N. Heirman, *Vice Chair*
Judith Gorman, *Secretary*

Satish K. Aggarwal
Clyde R. Camp
James T. Carlo
Gary R. Engmann
Harold E. Epstein
Jay Forster*
Thomas F. Garrity
Ruben D. Garzon
James H. Gurney
Jim D. Isaak
Lowell G. Johnson
Robert Kennelly
E. G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
Jim Logothetis
Donald C. Loughry
L. Bruce McClung
Louis-François Pau
Ronald C. Petersen
Gerald H. Peterson
John B. Posey
Gary S. Robinson
Hans E. Weinrich
Donald W. Zipse

*Member Emeritus

Kim Breitfelder
IEEE Standards Project Editor

Contents

IEEE Guide for Information Technology—System Definition—Concept of Operations (ConOps) Document

1. Scope

This guide prescribes the format and contents of the concept of operations (ConOps) document. A ConOps is a user-oriented document that describes system characteristics of the to-be-delivered system from the user's viewpoint. The ConOps document is used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements (e.g., training, facilities, staffing, and maintenance). It describes the user organization(s), mission(s), and organizational objectives from an integrated systems point of view.

This guide may be applied to all types of software-intensive systems: software-only or software/hardware/people systems. The concepts embodied in this guide could also be used for hardware-only systems, but this mode of use is not addressed herein. The size, scope, complexity, or criticality of the software product does not restrict use of this guide. This guide is applicable to systems that will be implemented in all forms of product media, including firmware, embedded systems code, programmable logic arrays, and software-in-silicon. This guide can be applied to any and all segments of a system life cycle.

This guide identifies the minimal set of elements that should appear in all ConOps documents. However, users of this guide may incorporate other elements by appending additional clauses or subclauses to their ConOps documents. In any case, the numbering scheme of the required clauses and subclauses should adhere to the format specified in this guide. Various clauses and subclauses of a ConOps document may be included by direct incorporation or by reference to other supporting documents.

2. References

This guide shall be used in conjunction with the following publications. In particular, the standards on requirements and plans should be consulted in preparing the ConOps. When the following standards are superseded by an approved revision, the revision shall apply.

IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology.[1]

[1]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Ln., P.O. Box 1331, Piscataway, NJ 08855-1331, USA.

IEEE Std 828-1998, IEEE Standard for Software Configuration Management Plans.

IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.

IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans.

IEEE Std 1058.1-1987 (Reaff 1993), IEEE Standard for Software Project Management Plans.

IEEE Std 1061-1992, IEEE Standard for Software Quality Metrics Methodology.

IEEE 1062, 1998 Edition, IEEE Recommended Practice for Software Acquisition.

IEEE Std 1074-1997, IEEE Standard for Developing Software Life Cycle Processes.

IEEE 1233, 1998 Edition, IEEE Guide for Developing System Requirements Specifications.

IEEE/EIA 12207.0-1996, IEEE/EIA Standard—Industry Implementation of ISO/IEC 12207:1995, for Information Technology— Software life cycle processes.

IEEE/EIA 12207.1-1997, IEEE/EIA Guide for Information Technology— Software life cycle processes—Life cycle data.

3. Definitions

The definitions listed here establish meanings within the context of this guide. Definitions of other terms that may be appropriate within the context of this guide can be found in IEEE Std 610.12-1990.

3.1 analysis: The process of studying a system by partitioning the system into parts (functions, components, or objects) and determining how the parts relate to each other.

3.2 buyer: (A) An individual or organization responsible for acquiring a product or service (for example, a software system) for use by themselves or other users. *See also:* **customer. (B)** The person or organization that accepts the system and pays for the project.

3.3 concept analysis: The derivation of a system concept through the application of analysis. *See also:* **analysis.**

3.4 concept of operations (ConOps) document: A user-oriented document that describes a system's operational characteristics from the end user's viewpoint. *Synonym:* **operational concept description (OCD).**

3.5 constraint: An externally imposed limitation on system requirements, design, or implementation or on the process used to develop or modify a system.

3.6 contract: In project management, a legally binding document agreed upon by the customer and the hardware or software developer or supplier; includes the technical, organizational, cost, and/or scheduling requirements of a project.

3.7 customer: (A) An individual or organization who specifies the requirements for and formally accepts delivery of a new or modified hardware or software product and its documentation; the customer may or may not be the ultimate user of the system. There are potentially many levels of customers, each with a different level of requirements to satisfy. The customer may be internal or external to the development organization for the project. *See also:* **user. (B)** An individual or organization who acts for the ultimate user of a new or modified hardware or software product to acquire the product and its documentation. *See also:* **buyer.**

3.8 developer: An organization that develops software products; "develops" may include new development, modification, reuse, reengineering, maintenance, or any other activity that results in software products, and includes the testing, quality assurance, configuration management, and other activities applied to these products. *Synonym:* **supplier.**

3.9 environment: The circumstances, objects, and conditions that surround a system to be built; includes technical, political, commercial, cultural, organizational, and physical influences as well as standards and policies that govern what a system must do or how it will do it.

3.10 functionality: The capabilities of the various computational, user interface, input, output, data management, and other features provided by a product.

3.11 mode: A set of related features or functional capabilities of a product, (e.g., on-line, off-line, and maintenance modes).

3.12 N^2 diagram: A system engineering or software engineering tool for tabulating, defining, analyzing, and describing functional interfaces and interactions among system components. The N^2 diagram is a matrix structure that graphically displays the bidirectional interrelationships between functions and components in a given system or structure.

3.13 operational concept description (OCD): *See:* **concept of operations (ConOps) document.**

3.14 priority: A rank order of status, activities, or tasks. Priority is particularly important when resources are limited.

3.15 problem domain: A set of similar problems that occur in an environment and lend themselves to common solutions.

3.16 request for proposal (RFP): A request for services, research, or a product prepared by a customer and delivered to prospective developers with the expectation that prospective developers will respond with their proposed cost, schedule, and development approach.

3.17 scenario: (A) A step-by-step description of a series of events that may occur concurrently or sequentially. **(B)** An account or synopsis of a projected course of events or actions.

3.18 software-intensive system: A system for which software is a major technical challenge and is perhaps the major factor that affects system schedule, cost, and risk. In the most general case, a software-intensive system is comprised of hardware, software, people, and manual procedures.

3.19 software life cycle: The system or product cycle initiated by a user need or a perceived customer need and terminated by discontinued use of the product. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase. These phases may overlap in time or may occur iteratively.

3.20 software system: A software-intensive system for which software is the only component to be developed or modified. *See also:* **software-intensive system.**

3.21 solution domain: The environment in which a solution or set of solutions resides. *See also:* **problem domain.**

3.22 supplier: *See:* **developer.**

3.23 system: (A) A collection of interacting components organized to accomplish a specific function or set of functions within a specific environment. **(B)** A group of people, objects, and procedures constituted to achieve defined objectives of some operational role by performing specified functions. A complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services, and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.

3.24 traceability: The identification and documentation of derivation paths (upward) and allocation or flowdown paths (downward) of work products in the work product hierarchy. Important kinds of traceability include: to or from external sources to or from system requirements; to or from system requirements to or from lowest level requirements; to or from requirements to or from design; to or from design to or from implementation; to or from implementation to test; and to or from requirements to test.

3.25 user: (A) An individual or organization who uses a software-intensive system in their daily work activities or recreational pursuits. **(B)** The person (or persons) who operates or interacts directly with a software-intensive system.

3.26 user need: A user requirement for a system that a user believes would solve a problem experienced by the user.

4. Elements of a ConOps document

This clause describes each of the essential elements of a ConOps document. These elements should be ordered in the sequence of clauses and subclauses shown in Table 1. Each version of a ConOps document based on this guide should contain a title and a revision notice that uniquely identifies the document. Revision information may include the project name, version number of the document, date of release, approval signatures, a list of subclauses that have been changed in the current version of the document, and a list of version numbers and dates of release of all previous versions of the document. The approved ConOps document should be placed under configuration control.

As indicated in Table 1, the preface of a ConOps document provides information that the writer wants the reader to know prior to reading the document. The preface should include the purpose of the document, the scope of activities that resulted in its development, who wrote the document and why, the intended audience for the document, and the expected evolution of the document.

A table of contents, a list of figures, and a list of tables should be included in every ConOps document, as indicated in Figure 1.

Title page
Revision chart
Preface
Table of contents
List of figures
List of tables
1. Scope
    1.1 Identification
    1.2 Document overview
    1.3 System overview
2. Referenced documents
3. Current system or situation
    3.1 Background, objectives, and scope
    3.2 Operational policies and constraints
    3.3 Description of the current system or situation
    3.4 Modes of operation for the current system or situation
    3.5 User classes and other involved personnel
    3.6 Support environment
4. Justification for and nature of changes
    4.1 Justication of changes
    4.2 Description of desired changes
    4.3 Priorities among changes
    4.4 Changes considered but not included
5. Concepts for the proposed system
    5.1 Background, objectives, and scope
    5.2 Operational policies and constraints
    5.3 Description of the proposed system
    5.4 Modes of operation
    5.5 User classes and other involved personnel
    5.6 Support environment
6. Operational scenarios
7. Summary of impacts
    7.1 Operational impacts
    7.2 Organizational impacts
    7.3 Impacts during development
8. Analysis of the proposed system
    8.1 Summary of improvements
    8.2 Disadvantages and limitations
    8.3 Alternatives and trade-offs considered
9. Notes
Appendices
Glossary

Figure 1—ConOps document outline

4.1 Scope (Clause 1 of the ConOps document)

Clause 1 provides an overview of the ConOps document and the system to which it applies.

4.1.1 Indentification (1.1 of the ConOps document)

This subclause contains the identifying number, title, and abbreviation (if applicable) of the system or subsystem to which this ConOps applies. If related ConOps documents for an overall system have been developed in a hierarchical or network manner, the position of this document relative to other ConOps documents should be described.

4.1.2 Document overview (1.2 of the ConOps document)

This subclause summarizes and expands on the purposes of motivations for the ConOps document. The intended audience for the document should also be mentioned. In addition, this subclause describes any security or privacy considerations associated with use of the ConOps. This subclause also outlines the remaining parts of this guide.

The purposes of a ConOps document will, in most cases, be:

— To communicate the user's needs for and expectations of the proposed system to the buyer and/or developer; or

— To communicate the buyer's or developer's understanding of the users' need and how the system shall operate to fulfill those needs.

However, a ConOps document might also serve other purposes, such as building consensus among several user groups, among several buyer organizations, and/or among several developers.

The audience of a ConOps document can be a variety of people.

— Users might read it to determine whether their needs and desires have been correctly specified by their representative or to verify the developer's understanding of their needs.

— Buyers might read it to acquire knowledge of the user's needs and/or developer's understanding of those needs.

— Developers will typically use the ConOps document as a basis for system development activities, and to familiarize new team members with the problem domain and the system to which the ConOps applies.

4.1.3 System overview (1.3 of the ConOps document)

This subclause briefly states the purpose of the proposed system or subsystem to which the ConOps applies. It describes the general nature of the system, and identifies the project sponsors, user agencies, development organizations, support agencies, certifiers or certifying bodies, and the operating centers or sites that will run the system. It also identifies other documents relevant to the present or proposed system.

A graphical overview of the system is strongly recommended. This can be in the form of a context diagram, a top-level object diagram, or some other type of diagram that depicts the system and its environment.

Documents that might be cited include, but are not limited to: the project authorization, relevant technical documentation, significant correspondence, documents concerning related projects, risk analysis reports, and feasibility studies.

4.2 Referenced documents (Clause 2 of the ConOps document)

This clause lists the document number, title, revision, and date of all documents referenced in the ConOps document. This clause should also identify the source for all documents not available through normal channels.

4.3 Current system or situation (Clause 3 of the ConOps document)

Clause 3 describes the system or situation (either automated or manual) as it currently exists. If there is no current system on which to base changes, this subclause describes the situation that motivates development of the proposed system. In this case, the following subclauses will be tailored as appropriate to describe the motivating situation.

This clause also provides readers with an introduction to the problem domain. This enables readers to better understand the reasons for the desired changes and improvements.

4.3.1 Background, objectives, and scope (3.1 of the ConOps document)

This subclause provides an overview of the current system or situation, including as applicable, background, mission, objectives, and scope. In addition to providing the background for the current system, this subclause should provide a brief summary of the motivation for the current system. Examples of motivations for a system might include automation of certain tasks or countering of certain threat situations. The goals for the current system should also be defined, together with the strategies, solutions, tactics, methods, and techniques used to accomplish them. The modes of operation, classes of users, and interfaces to the operational environment define the scope of the proposed system, which are summarized in this clause and defined in greater detail in subsequent clauses.

4.3.2 Operational policies and constraints (3.2 of the ConOps document)

This subclause describes any operational policies and constraints that apply to the current system or situation. Operational policies are predetermined management decisions regarding the operations of the current system, normally in the form of general statements or understandings that guide decision making activities. Policies limit decision-making freedom but do allow for some discretion. Operational constraints are limitations placed on the operations of the current system. Examples of operational constraints include the following:

- A constraint on the hours of operation of the system, perhaps limited by access to secure terminals
- A constraint on the number of personnel available to operate the system
- A constraint on the computer hardware (for example, must operate on computer X)
- A constraint on the operational facilities, such as office space

4.3.3 Description of the current system or situation (3.3 of the ConOps document)

This subclause will contain the major portion of the description of the current system. It provides a description of the current system or situation, including the following, as appropriate:

a) The operational environment and its characteristics;

b) Major system components and the interconnection among those components;

c) Interfaces to external systems or procedures;

d) Capabilities, functions, and features of the current system;

e) Charts and accompanying descriptions depicting inputs, outputs, data flows, control flows, and manual and automated processes sufficient to understand the current system or situation from the user's point of view;

f) Cost of system operations;

g) Operational risk factors;

h) Performance characteristics, such as speed, throughput, volume, frequency;

i) Quality attributes, such as: availability, correctness, efficiency, expandability, flexibility, interoperability, maintain-ability, portability, reliability, reusability, supportability, survivability, and usability; and

j) Provisions for safety, security, privacy, integrity, and continuity of operations in emergencies.

Since the purpose of this clause is to describe the current system and how it operates, it is appropriate to use any tools and/or techniques that serve this purpose. It is important that the description of the system be simple enough and clear enough that all intended readers of the document can fully understand it. It is also important to keep in mind that the ConOps document shall be written using the users' terminology. In most cases, this means avoidance of terminology specific to computers (i.e., "computer jargon").

Graphical tools should be used wherever possible, especially since ConOps documents should be understandable by several different types of readers. Useful graphical tools include, but are not limited to, work breakdown structures (WBS), N^2 charts, sequence or activity charts, functional flow block diagrams, structure charts, allocation charts, data flow diagrams (DFD), object diagrams, context diagrams, storyboards, and entity-relationship diagrams.

The description of the operational environment should identify, as applicable, the facilities, equipment, computing hardware, software, personnel, and operational procedures used to operate the existing system. This description should be as detailed as necessary to give the readers an understanding of the numbers, versions, capacity, etc., of the operational equipment being used. For example, if the current system contains a database, the capacity of the storage unit(s) should be specified, provided the information exerts an influence on the users' operational capabilities. Likewise, if the system uses communication links, the capacities of those links should be specified if they exert influence on factors such as user capabilities, response time, or throughput.

Those aspects of safety, security, and privacy that exert influence on the operation or operational environment of the current system should be described.

The author(s) of a ConOps document should organize the information in this subclause as appropriate to the system or situation, as long as a clear description of the existing system is achieved. If parts of the descriptions are voluminous, they can be included in an appendix or incorporated by reference. An example of material that might be included in an appendix would be a data dictionary. An example of material to be included by reference might be a detailed manual of operational policies and procedures for the current system.

4.3.4 Modes of operation for the current system or situation (3.4 in the ConOps document)

This subclause describes the various modes of operation for the current system or situation (e.g., operational, degraded, maintenance, training, emergency, alternate-site, peacetime, wartime, ground-based, flight, active, and idle modes). All of the modes that apply to all classes of users should be included. Important modes to include are degraded, backup, and emergency modes, if such exist. This is especially true if these modes involve different geographical sites and equipment that have significant impacts on the operational aspects of the system.

This subclause can be further divided into lower-level subclauses, one for each mode described. System processes, procedures, and capabilities or functions should be related to each mode, as appropriate, perhaps using a cross-reference matrix.

4.3.5 User classes and other involved personnel (3.5 of the ConOps document)

A user class is distinguished by the ways in which users interact with the system. Factors that distinguish a user class include common responsibilities, skill levels, work activities, and modes of interaction with the system. Different user classes may have distinct operational scenarios for their interactions with the system. In this context, a user is anyone who interacts with the existing system, including operational users, data entry personnel, system operators, operational support personnel, software maintainers, and trainers.

This subclause can be organized further, as follows, if it is helpful in communicating the content.

4.3.5.1 Organizational structure (3.5.1 of the ConOps document)

This subclause describes the existing organizational structures of the various user groups and user classes that are involved with the current system. Organizational charts are useful graphic tools for this purpose.

4.3.5.2 Profiles of user classes (3.5.2 of the ConOps document)

This subclause provides a profile of each user class for the current system. If some users play several roles, each role should be identified as a separate user class.

Each user class for the current system, including operators and maintainers, should be described in a separate subclause. Each of these should provide a description of the user class, including responsibilities, education, background, skill level, activities, and modes of interaction with the current system.

4.3.5.3 Interactions among user classes (3.5.3 of the ConOps document)

This subclause describes interactions among the various user classes involved with the current system. In particular, interactions among user groups, operators, and maintainers should be described. Interactions that occur among the users of the system, and between users and non-users, both within the organization and across organizational boundaries, if they are relevant to the operation of the existing system, should be described. Informal as well as formal interactions should be included.

4.3.5.4 Other involved personnel (3.5.4 of the ConOps document)

This subclause describes other personnel who will not directly interact with the system, but who have an influence on, and are influenced by, the present system. Examples include executive managers, policy makers, and the user's clients. Although these individuals do not have hands-on interaction with the system, they may significantly influence, and be influenced by, the new or modified system.

4.3.6 Support environment (3.6 of the ConOps document)

This subclause describes the support concepts and support environment for the current system, including the support agency or agencies; facilities; equipment; support software; repair or replacement criteria; maintenance levels and cycles; and storage, distribution, and supply methods.

4.4 Justification for and nature of changes (Clause 4 of the ConOps document)

Clause 4 of the ConOps document describes the shortcomings of the current system or situation that motivate development of a new system or modification of an existing system. This clause provides a transition from Clause 3 of the ConOps, which describes the current system or situation, to Clause 5 of the ConOps, which describes the proposed system. If there is no current system on which to base changes, this subclause should so indicate and provide justification for the features of the new system.

4.4.1 Justification for changes (4.1 of the ConOps document)

This subclause should:

a) Briefly summarize new or modified aspects of the user needs, missions, objectives, environments, interfaces, personnel, or other factors that require a new or modified system;
b) Summarize the deficiencies or limitations of the current system or situation that make it unable to respond to new or changed factors; and
c) Provide justification for a new or modified system.
    1) If the proposed system is to meet a new opportunity, describe the reasons why a new system should be developed to meet this opportunity.
    2) If the proposed system improves a current operation, describe the rationale behind the decision to modify the existing system (e.g., to reduce life cycle costs or improve personnel efficiency).
    3) If the proposed system implements a new functional capability, explain why this function is necessary.

4.4.2 Description of desired changes (4.2 of the ConOps document)

This subclause summarizes new or modified capabilities, functions, processes, interfaces, and other changes needed to respond to the factors identified in 4.1. Changes should be based on the current system described in Clause 3 of the ConOps document. If there is no existing system on which to base changes, this subclause should summarize the capabilities to be provided by a new system. This description should include the following, as appropriate:

a) *Capability changes*. Description of the functions and features to be added, deleted, and modified in order for the new or modified system to meet its objectives and requirements.

b) *System processing changes*. Description of the changes in the process or processes of transforming data that will result in new output with the same data, the same output with new data, or both.

c) *Interface changes*. Description of changes in the system that will cause changes in the interfaces and changes in the interfaces that will cause changes in the system.

d) *Personnel changes*. Description of changes in personnel caused by new requirements, changes in user classes, or both.

e) *Environment changes*. Description of changes in the operational environment that will cause changes in the system functions, processes, interfaces, or personnel and/or changes that should be made in the environment because of changes in the system functions, processes, interfaces, or personnel.

f) *Operational changes*. Description of changes to the user's operational policies, procedures, methods, or daily work routines caused by the above changes.

g) *Support changes*. Description of changes in the support requirements caused by changes in the system functions, processes, interfaces, or personnel and/or changes in the system functions, processes, interfaces, or personnel caused by changes in the support environment.

h) *Other changes*. Description of other changes that will impact the users, but that do not fit under any of the above categories.

4.4.3 Priorities among changes (4.3 of the ConOps document)

This subclause identifies priorities among the desired changes and new features. Each change should be classified as essential, desirable, or optional. Desirable and optional changes should be prioritized within their classes. If there is no existing system on which to base changes, this subclause should classify and prioritize the features of the proposed system.

a) *Essential features*. Features that shall be provided by the new or modified system. The impacts that would result if the features were not implemented should be explained for each essential feature.

b) *Desirable features*. Features that should be provided by the new or modified system. Desirable features should be prioritized. Reasons why the features are desirable should be explained for each desirable feature.

c) *Optional features*. Features that might be provided by the new or modified system. Optional features should be prioritized. Reasons why the features are optional should be explained for each optional feature.

Classifying the desired changes and new features into essential, desirable, and optional categories is important to guide the decision making process during development of the proposed system. This information is also helpful in cases of budget or schedule cuts or overruns, since it permits determination of which features must be finished, and which ones can be delayed or omitted.

4.4.4 Changes considered but not included (4.4 of the ConOps document)

This subclause identifies changes and new features considered but not included in 4.2 of the ConOps document, and the rationale for not including them. By describing changes and features considered but not included in the proposed system, the authors document the results of their analysis activities. This information can be useful to other personnel involved with system development, whether it be users, buyers, or developers should they want to know if a certain

change or feature was considered, and if so, why it was not included. In software especially, there are few, if any, outward signs of what has been changed, improved or is still unsafe or unsecure (e.g., in certain scenarios or workarounds).

4.4.5 Assumptions and constraints (4.5 of the ConOps document)

This subclause describes any assumptions or constraints applicable to the changes and new features identified in this clause. This should include all assumptions and constraints that will affect users during development and operation of the new or modified system. An assumption is a condition that is taken to be true. An example of an assumption is that the system workload will double over the next two years, thus a new system with higher performance is required. A constraint is an externally imposed limitation placed on the new or modified system or the processes used to develop or modify the system. Examples of constraints include external interface requirements, and limits on schedule and budget.

4.5 Concepts for the proposed system (Clause 5 of the ConOps document)

This clause describes the proposed system that results from the desired changes specified in Clause 4 of the ConOps document. This clause describes the proposed system in a high-level manner, indicating the operational features that are to be provided without specifying design details. Methods of description to be used and the level of detail in the description will depend on the situation. The level of detail should be sufficient to fully explain how the proposed system is envisioned to operate in fulfilling users' needs and buyer's requirements.

In some cases, it may be necessary to provide some level of design detail in the ConOps. The ConOps should not contain design specifications, but it may contain some examples of typical design strategies, for the purpose of clarifying operational details of the proposed system. In the event that actual design constraints need to be included in the description of the proposed system, they shall be explicitly identified as required to avoid possible misunderstandings.

NOTE — If some of the features of the proposed system are the same as the features of the original system, then the comment "no change" should appear after the subclause number and name.

4.5.1 Background, objectives, and scope (5.1 of the ConOps document)

This subclause provides an overview of the new or modified system, including, as applicable, background, mission, objectives, and scope. In addition to providing the background for the proposed system, this subclause should provide a brief summary of the motivation for the system. Examples of motivations for a system might include automation of certain tasks or taking advantage of new opportunities. The goals for the new or modified system should also be defined, together with the strategies, solutions, tactics, methods, and techniques proposed to achieve those goals. The modes of operation, classes of users, and interfaces to the operational environment define the scope of the proposed system, which are summarized in this subclause and defined in greater detail in subsequent subclauses.

4.5.2 Operational policies and constraints (5.2 of the ConOps document)

This subclause describes operational policies and constraints that apply to the proposed system. Operational policies are predetermined management decisions regarding the operation of the new or modified system, normally in the form of general statements or understandings that guide decision-making activities. Policies limit decision-making freedom, but do allow for some discretion. Operational constraints are limitations placed on the operations of the proposed system. Examples of operational constraints include the following:

- A constraint on the hours of operations of the system, perhaps limited by access to secure terminals;
- A limiting constraint on the number of personnel available to operate the system;
- A limiting constraint on the computer hardware (e.g., must operate on computer X); and
- A limiting constraint on the operational facilities, such as office space.

4.5.3 Description of the proposed system (5.3 of the ConOps document)

This subclause will contain the major portion of the description of the proposed system. It provides a description of the proposed system, including the following, as appropriate:

a) The operational environment and its characteristics;

b) Major system components and the interconnections among these components;

c) Interfaces to external systems or procedures;

d) Capabilities or functions of the proposed system;

e) Charts and accompanying descriptions depicting inputs, outputs, data flow, and manual and automated processes sufficient to understand the proposed system or situation from the user's point of view;

f) Cost of systems operations;

g) Operational risk factors;

h) Performance characteristics, such as speed, throughput, volume, frequency;

i) Quality attributes, such as: reliability, availability, correctness, efficiency, expandability, flexibility, interoperability, maintainability, portability, reusability, supportability, survivability, and usability; and

j) Provisions for safety, security, privacy, integrity, and continuity of operations in emergencies.

Since the purpose of this subclause is to describe the proposed system and how it should operate, it is appropriate to use any tools and/or techniques that serve that purpose. It is important that the description of the system be simple enough and clear enough that all intended readers of the document can fully understand it. It is important to keep in mind that the ConOps shall be written in the user's language. In most cases, this means avoidance of terminology specific to computers—in other words, "computer jargon."

Graphics and pictorial tools should be used wherever possible, especially since ConOps documents should be understandable be several different types of readers. Useful graphical tools include, but are not limited to, WBS, N^2 charts, sequence or activity charts, functional flow block diagrams, structure charts, allocation charts, DFDs, object diagrams, storyboards, and entity relationship diagrams.

The description of the operational environment should identify, as applicable, the facilities, equipment, computing hardware, software, personnel, and operational procedures needed to operate the proposed system. This description should be as detailed as necessary to give the readers an understanding of the numbers, versions, capacity, etc., of the operational equipment to be used. For example, if the proposed system contains a database, the capacity of the storage units should be specified, provided that information influences the users' operational capabilities. Likewise, if the system uses communication links, then the capacities of those links should be specified if they exert influence on user capabilities or response time.

Those aspects of safety, security, and privacy that exert influence on the operation or operational environment of the proposed system should be described.

The author(s) of a ConOps document should organize the information in this subclause as appropriate to the system or situation, as long as a clear description of the proposed system is achieved. If parts of the description are voluminous, they can be included in an appendix or incorporated by reference. An example of material that might be included in an appendix would be a data dictionary. An example of material to be included by reference might be a detailed manual of operation policies and procedures for the proposed system.

4.5.4 Modes of operation (5.4 of the ConOps document)

This subclause describes the various modes of operation for the proposes system (for example, regular, degraded, maintenance, training, emergency, alternate-site, peacetime, wartime, ground-based, flight, active, and idle modes). Include all of the modes that apply to all user classes. Important modes to include are degraded, backup, and emergency modes, if such exist. This is especially true if these modes involve different geographical sites and equipment that have significant impacts on the system.

This subclause can be further divided into lower-level subclauses, one for each mode described. System processes, procedures, and capabilities or functions should be related to each mode.

4.5.5 User classes and other involved personnel (5.5 of the ConOps document)

A user class is distinguished by the ways in which the users interact with the system. Factors that distinguish a user class include responsibilities, skill level, work activities, and mode of interaction with the system. Different user classes may have distinct operational scenarios for their interactions with the system. In this context, a user is anyone who will interact with the proposed system, including operational users, data entry personnel, system operators, operational support personnel, software maintainers, and trainers.

This subclause can be further divided into lower-level subclauses if it is helpful in communicating the content.

4.5.5.1 Organizational structure (5.5.1 of the ConOps document)

This subclause describes the organizational structures of the various user groups and user classes that will be involved with the proposed system. Organizational charts are useful graphic tools for this purpose.

4.5.5.2 Profiles of user classes (5.5.2 of the ConOps document)

This subclause provides a profile of each user class for the proposed system. If some users play several roles, each role should be identified as a separate user class.

Each user class for the proposed system, including operators and maintainers, should be described in a separate subclause. Each subclause should provide a description of the user class, including responsibilities, education, background, skill level, activities, and envisioned modes of interaction with the proposed system.

4.5.5.3 Interactions among user classes (5.5.3 of the ConOps document)

This subclause describes interactions among the various user classes that may be involved with the proposed system. In particular, interaction among user groups, operators, and maintainers should be described. Interactions that will occur among the users of the proposed system, and between users and non-users, both within the organization and across interfacing organizations, if they are relevant to the operation of the proposed system, should be described. Informal as well as formal interactions should be included.

4.5.5.4 Other involved personnel (5.5.4 of the ConOps document)

This subclause describes other personnel who will not directly interact with the system, but who have an influence on, and are influenced by, the present system. Examples include executive managers, policy makers, and the user's clients. Although these individuals do not have hands-on interaction with the system, they may significantly influence and be influenced by, the new or modified system.

4.5.6 Support environment (5.6 of the ConOps document)

This subclause describes the support concepts and support environment for the proposed system, including the support agency or agencies; facilities; equipment; support software; repair or replacement criteria; maintenance levels and cycles; and storage, distribution, and supply methods.

4.6 Operational scenarios (Clause 6 of the ConOps document)

A scenario is a step-by-step description of how the proposed system should operate and interact with its users and its external interfaces under a given set of circumstances. Scenarios should be described in a manner that will allow readers to walk through them and gain an understanding of how all the various parts of the proposed system function and interact. The scenarios tie together all parts of the system, the users, and other entities by describing how they interact. Scenarios may also be used to describe what the system should not do.

Scenarios should be organized into clauses and subclauses, each describing an operational sequence that illustrates the roles of the system, its interactions with users, and interactions with other systems. Operational scenarios should be described for all operational modes and all classes of users identified for the proposed system. Each scenario should include events, actions, stimuli, information, and interactions as appropriate to provide a comprehensive understanding of the operational aspects of the proposed system. Prototypes, storyboards, and other media, such as video or hypermedia presentations, may be used to provide part of this information.

In most cases, it will be necessary to develop several variations of each scenario, including one for normal operation, one for stress load handling, one for exception handling, one for degraded mode operation, etc.

Scenarios play several important roles. The first is to bind together all of the individual parts of a system into a comprehensible whole. Scenarios help the readers of a ConOps document understand how all the pieces interact to provide operational capabilities. The second role of scenarios is to provide readers with operational details for the proposed system; this enables them to understand the users' roles, how the system should operate, and the various operational features to be provided.

Scenarios can also support the development of simulation models that help in the definition and allocation of derived requirements, identification, and preparation of prototypes to address key issues.

In addition, scenarios can serve as the basis for the first draft of the users' manual, and as the basis for developing acceptance test plans. The scenarios are also useful for the buyer and the developer to verify that the system design will satisfy the users' needs and expectations.

Scenarios can be presented in several different ways. One approach is to specify scenarios for each major processing function of the proposed system. Using this approach, this clause would contain one subclause for each process. Each subclause would then contain several more lower-level subclauses, one for each scenario supported by that process. An alternative approach is to develop thread-based scenarios, where each scenario follows one type of transaction type through the proposed system. In this case, each subclause would contain one scenario for each interaction type, plus scenarios for degraded, stress loaded, and back-up modes of operation. Other alternatives include following the information flow through the system for each user capability, following the control flows, or focusing on the objects and events in the system.

Scenarios are an important component of a ConOps, and should therefore receive substantial emphasis. The number of scenarios and level of detail specified will be proportional to the perceived risk and the criticality of the project.

4.7 Summary of impacts (Clause 7 of the ConOps document)

This clause describes the operational impacts of the proposed system on the users, the developers, and the support and maintenance organizations. It also describes the temporary impacts on users, buyers, developers, and the support and maintenance organizations during the period of time when the new system is being developed, installed, or trained on.

This information is provided in order to allow all affected organizations to prepare for the changes that will be brought about by the new system and to allow for planning of the impacts on the buyer agency or agencies, user groups, and the support maintenance organizations during the development of, and transition to the new system.

4.7.1 Operational impacts (7.1 of the ConOps document)

This subclause should be further divided into lower-level subclauses to describe the anticipated operational impacts on the user, development, and support or maintenance agency or agencies during operation of the proposed system. These impacts may include the following:

- Interfaces with primary or alternate computer operating centers;
- Changes in procedure;
- Use of new data sources;
- Changes in quantity, type, and timing of data to be input into the system;
- Changes in data retention requirements;
- New modes of operation based on emergency, disaster, or accident conditions;
- New methods for providing input data if the required data are not readily available;
- Changes in operational budget; and
- Changes in operational risks.

4.7.2 Organizational impacts (7.2 of the ConOps document)

This subclause should be further divided into lower-level subclauses to describe the anticipated operational impacts on the user, development, and support or maintenance agency or agencies during operation of the proposed system. These impacts may include the following:

- Modification of responsibilities; responsibilities;
- Addition or elimination of job positions; positions;
- Training or retraining users; users;
- Changes in numbers, skill levels, position identifiers, or locations of personnel; personnel; and
- Numbers and skill levels of personnel needed for contingency operation at one or more alternate sites following an emergency, disaster, or accident.

4.7.3 Impacts during development (7.3 of the ConOps document)

This subclause should be further divided into lower-level subclauses that describe the anticipated impacts on the user, development, and support or maintenance agency or agencies during the development project for the proposed system. These impacts may include the following:

- Involvement in studies, meetings, and discussions prior to award of the contract;
- User and support involvement in reviews and demonstrations, evaluation of initial operating capabilities and evolving versions of the system, development or modification of databases, and required training;
- Parallel operation of the new and existing systems; and
- Operational impacts during system testing of the proposed system.

4.8 Analysis of the proposed system (Clause 8 of the ConOps document)

This clause provides an analysis of the benefits, limitations, advantages, disadvantages, and alternatives and trade-offs considered for the proposed system.

4.8.1 Summary of improvements (8.1 of the ConOps document)

This subclause provides a qualitative (and to the extent possible, quantitative) summary of the benefits to be provided by the proposed system. This summary should include the below items, as applicable. In each case, the benefits should be related to deficiencies identified in 4.1 of the ConOps.

— *New capabilities*. Additional new features or functionality.
— *Enhanced capabilities*. Upgrades to existing capabilities.
— *Deleted capabilities*. Unused, obsolete, confusing, or dangerous capabilities removed.
— *Improved performance*. Better response time, reduced storage requirements, improved quality, etc.

4.8.2 Disadvantages and limitations (8.2 of the ConOps document)

This subclause provides a qualitative (and to the extent possible, quantitative) summary of the disadvantages and/or limitations of the proposed system. Disadvantages might include the need to retrain personnel, rearrange work spaces, or change to a new style of user interface; limitations might include features desired by users but not included, degradation of existing capabilities to gain new capabilities, or greater-than-desired response time for certain complex operations.

4.8.3 Alternatives and trade-offs considered (8.3 of the ConOps document)

This subclause should describe major alternatives considered, the trade-offs among them, and rationale for the decisions reached. In the context of a ConOps document, alternatives are operational alternatives and not design alternatives, except to the extent that designs alternatives may be limited by the operational capabilities desired in the new system. This information can be useful to determine, now and at later times, whether a given approach was analyzed and evaluated, or why a particular approach or solution was rejected. This information would probably be lost if not recorded.

4.9 Notes (Clause 9 on the ConOps document)

This clause should contain any additional information that will aid understanding of a particular ConOps document. This clause should include an alphabetical listing of all acronyms and abbreviations, along with their meanings as used in this document, and a list of any terms and definitions needed to understand the document.

4.10 Appendices (Appendices of the ConOps document)

To facilitate ease of use and maintenance of the ConOps document, some information may be placed in appendices to the document. Charts and classified data are typical examples. Each appendix should be referenced in the main body of the document where that information would normally have been provided. Appendices may be bound as separate documents for easier handling.

4.11 Glossary (Glossary of the ConOps document)

The inclusion of a clear and concise definition of terms used in the ConOps document (but that may be unfamiliar to readers of the ConOps document) is very important. A glossary should be maintained and updated during the processes of concept analysis and development of the ConOps document. To avoid unnecessary work due to misinterpretations, all definitions should be reviewed and agreed upon by all involved parties.

Annex A

IEEE/EIA 12207.1-1997 Compliance Statement

(Informative)

A.1 Overview

The Software Engineering Standards Committee (SESC) of the IEEE Computer Society has endorsed the policy of adopting international standards. In 1995, the international standard, ISO/IEC 12207, Information technology—Software life cycle processes, was completed. That standard establishes a common framework for software life cycle processes, with well-defined terminology, that can be referenced by the software industry.

In 1995 SESC evaluated ISO/IEC 12207 and decided that the standard should be adopted and serve as the basis for life cycle processes within the IEEE Software Engineering Collection. The IEEE adaptation of ISO/IEC 12207 is IEEE/EIA 12207.0-1996. It contains ISO/IEC 12207 and the following additions: improved compliance approach, life cycle process objectives, life cycle data objectives, and errata.

The implementation of ISO/IEC 12207 within the IEEE also includes the following:

- IEEE/EIA 12207.1-1997, IEEE/EIA Guide for Information Technology—Software life cycle processes—Life cycle data;
- IEEE/EIA 12207.2-1997, IEEE/EIA Guide for Information Technology—Software life cycle processes—Implementation considerations; and
- Additions to 11 existing SESC standards (i.e., IEEE Stds 730, 828, 829, 830, 1012, 1016, 1058, 1062, 1219, 1233, and 1362) to define the correlation between the data produced by existing SESC standards and the data produced by the application of IEEE/EIA 12207.1-1997.

NOTE — Although IEEE/EIA 12207.1-1997 is a guide, it also contains provisions for application as a standard with specific compliance requirements. This annex treats IEEE/EIA 12207.1-1997 as a standard.

In order to achieve compliance with both this standard and IEEE/EIA 12207.1-1997, it is essential that the user review and satisfy the data requirements for both standards.

When this standard is directly referenced, the precedence for conformance is based upon this standard alone. When this standard is referenced with the IEEE/EIA 12207.x standard series, the precedence for conformance is based upon the directly referenced IEEE/EIA 12207.x standard, unless there is a statement that this standard has precedence.

A.1.1 Scope and purpose

Both this standard and IEEE/EIA 12207.1-1997 place requirements on a ConOps document. The purpose of this annex is to explain the relationship between the two sets of requirements so that users producing documents intended to comply with both standards may do so.

A.2 Correlation

This clause explains the relationship between this standard and IEEE/EIA 12207.0-1996 in the following areas: terminology, process, and life cycle data.

A.2.1 Terminology correlation

Both this standard and IEEE/EIA 12207.0-1996 have similar semantics for the key terms of change, constraints, environment, modes of operation, policy, and system.

A.2.2 Process correlation

This standard places no requirements on process.

A.2.3 Life cycle data correlation and concept of operations documents

The information required in a ConOps document by this standard and the information required in a ConOps document by IEEE/EIA 12207.1-1997 are similar. It is reasonable to expect that a single document could comply with both standards. Both documents use a process-oriented context to describe the content of a ConOps document.

A.2.4 Life cycle data correlation between other data in IEEE/EIA 12207.1-1997 and this standard

This sublause correlates the life cycle data other than a ConOps document between IEEE/EIA 12207.1-1997 and this standard. It provides information to users of both standards.

Table A-1—Life cycle data correlation between other data in IEEE/EIA 12207.1-1997 and IEEE Std 1362-1998

Information item	IEEE/EIA 12207.0-1996 subclause	Kind	IEEE/EIA 12207.1-1997 subclause	IEEE Std 1362-1998 subclause
System architecture and requirements allocation description	5.3.3.1, 5.3.3.2	Description	6.25	4.5.3

A.3 Document compliance

This clause provides details bearing on a claim that a ConOps document complying with this standard would also achieve “document compliance” with the ConOps document described in IEEE/EIA 12207.1-1997. The requirements for document compliance are summarized in a single row of Table 1 of IEEE/EIA 12207.1-1997. That row is reproduced in Table A.2.

Table A-2—Summary of requirements for a ConOps document excerpted from Table 1 of IEEE/EIA Std 12207.1-1997.

Information item(s)	IEEE/EIA 12207.0-1996 subclause	Kind	IEEE/EIA 12207.1-1997 subclause	References
ConOps	5.1.1.1	Description	6.3	IEEE Std 1362-1998 EIA/IEEE J-STD-016, F.2.1 Also see the following for guidance on the use of notations: ISO 5806: 1984 ISO 5807: 1985 ISO 8631: 1989 ISO 8790: 1987 ISO 11411: 1995

The requirements for document compliance are discussed in the following subclauses:

— A.3.1 discusses compliance with the information requirements noted in column 2 of Table A.2 as prescribed by Clause 5.1.1.1 of IEEE/EIA 12207.0-1996.

— A.3.2 discusses compliance with the generic content guideline (the "kind" of document) noted in column 3 of Table A.2 as a "description." The generic content guidelines for a "description" appear in 5.1 of IEEE/EIA 12207.1-1997.

— A.3.3 discusses compliance with the specific requirements for a ConOps document noted in column 4 of Table A.2 as prescribed by 6.3 of IEEE/EIA 12207.1-1997.

— A.3.4 discusses compliance with the life cycle data objectives of Annex H of IEEE/EIA 12207.0-1996 as described in 4.2 of IEEE/EIA 12207.1-1997.

A.3.1 Compliance with information requirements of IEEE/EIA 12207.0-1996

The information requirements for a ConOps document are those prescribed by 5.1.1.1 of IEEE/EIA 12207.0-1996. In this case, those requirements are substantially identical with those considered in A.3.3 of this standard.

A.3.2 Compliance with generic content guidelines of IEEE/EIA 12207.1-1997

The generic content guidelines for a "description" in IEEE/EIA 12207.1-1997 are prescribed by 5.1 of IEEE/EIA 12207.1-1997. A complying description shall achieve the purpose stated in 5.1.1 and include the information listed in 5.1.2 of IEEE/EIA 12207.1-1997.

The purpose of a description is:

> IEEE/EIA 12207.1-1997, subclause 5.1.1: Purpose: Describe a planned or actual function, design, performance, or process.

A ConOps document complying with this standard would achieve the stated purpose.

Any description complying with IEEE/EIA 12207.1-1997 shall satisfy the generic content requirements provided in 5.1.2 of that standard. Table A.3 of this standard lists the generic content items and, where appropriate, references the subclause of this standard that requires the same information. The third column lists information that shall be added in order to comply with the generic content requirements.

Table A-3—Coverage of generic description requirements by ConOps document listed in IEEE Std 1362-1998

IEEE/EIA 12207.1-1997 generic content	Corresponding subclauses of IEEE Std 1362-1998	Additions to requirements of IEEE Std 1362-1998
a) Date of issue and status	—	Date of issue and status should be provided in the Revision Chart or as an additional clause after the Glossary, Configuration management.
b) Scope	4.1 Scope (Clause 1 of the ConOps document)	—
c) Issuing organization	—	Issuing organization should be identified and referenced in the Revision Chart or as an additional clause after the Glossary, Configuration management.
d) References	4.2 Referenced documents (Clause 2 of the ConOps document)	—
e) Context	4.3 Current system or situation (Clause 3 of the ConOps document)	—
f) Notation for description	—	The definition of, or appropriate references to a definition of, the notation used for the system overview graphics should be included in 1.3 of the ConOps document if the system graphics are included.
g) Body	4.4 Justification for and nature of changes (Clause 4 of the ConOps document) 4.5 Concepts for the proposed system (Clause 5 of the ConOps document) 4.6 Operational scenarios (Clause 6 of the ConOps document) 4.7 Summary of impacts (Clause 7 of the ConOps document)	—
h) Summary	4.8 Analysis of the proposed system (Clause 8 of the ConOps document)	—
i) Glossary	4.11 Glossary (Glossary of the ConOps document)	—
j) Change history	—	Change history for the ConOps document should be provided or referenced in an additional clause after the Glossary, Configuration management.

A.3.3 Compliance with specific content requirements of IEEE/EIA 12207.1-1997

The specific content requirements for a ConOps document in IEEE/EIA 12207.1-1997 are prescribed by 6.3 of IEEE/EIA 12207.1-1997. A complying ConOps document shall achieve the purpose stated in 6.3.1 and include the information listed in 6.3.3 of IEEE/EIA 12207.1-1997.

The purpose of the ConOps document is:

> IEEE/EIA 12207.1-1997, subclause 6.3.1: Purpose: Describe, in users' terminology, how the system should operate to meet the users' needs for the system.

A ConOps document complying with IEEE/EIA 12207.1-1997 shall satisfy the specific content requirements provided in 6.3.3 of that standard. Table A.4 of this standard lists the specific content items and, where appropriate, references the subclause of this standard that requires the same information. The third column lists information that shall be added in order to comply with the specific content requirements.

Table A-4—Coverage of specific ConOps document requirements by requirements by ConOps document listed in IEEE Std 1362-1998

IEEE/EIA Std 12207.1-1997 specific content	Corresponding subclauses of IEEE Std 1362-1998	Additions to requirements of IEEE Std 1362-1998
a) Generic description information	4.1 Scope (Clause 1 of the ConOps document)	See Table A.2
b) Description of current situation or system	4.3 Current system or situation (Clause 3 of the ConOps document)	—
c) Justification for and nature of changes	4.4 Justification for and nature of changes (Clause 4 of the ConOps document)	—
d) Concepts of the proposed system	4.5 Concepts for the proposed system (Clause 5 of the ConOps document)	—
e) Operational scenarios	4.6 Operational scenarios (Clause 6 of the ConOps document)	—
f) Summary of impacts	4.7 Summary of impacts (Clause 7 of the ConOps document)	—
g) Analysis of the proposed system	4.8 Analysis of the proposed system (Clause 8 of the ConOps document)	—
h) Priorities, assumptions, constraints, advantages, limitations, alternatives, and trade-offs considered	4.8.1 Summary of improvements 4.8.2 Disadvantages and limitations 4.8.3 Alternatives and tradeoffs considered	—

A.3.4 Compliance with life cycle data characteristics

In addition to the content requirem

ents, life cycle data shall be managed in accordance with the objectives provided in Annex H of IEEE/EIA 12207.0-1996.

NOTE — The information items covered by this standard include plans and provisions for creating software life cycle data related to the basic types “requirements data” and “user data” in H.4 of IEEE/EIA 12207.0-1996. Requirements data provides for the following: expected functionality, operational context, performance constraints and expectations, basis for qualification testing, and key decision rationale. User data provides the following: software overview, system access information, commands and responses, error messages, operational environment, and key decision rationale.

A.4 Conclusion

The analysis suggests that any ConOps document complying with this standard and the additions shown in Table A.3 and Table A.4 will comply with the requirements of a ConOps document in IEEE/EIA 12207.1-1997. In addition, to comply with IEEE/EIA 12207.1-1997, any document shall support the life cycle data objectives of Annex H of IEEE/EIA 12207.0-1996.