NUWC-NPT Technical Document 10,889
18 May 1995

Algorithm Description Guide for the Advanced Engineering Framework Improvement Group (AEFIG)

Version 1.0

J. A. Ionata
Submarine Sonar Department

Naval Undersea Warfare Center Division
Newport, Rhode Island

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	18 May 1995	Final

4. TITLE AND SUBTITLE

Algorithm Description Guide for the Advanced Engineering Framework Improvement Group (AEFIG), Version 1.0

5. FUNDING NUMBERS

PR V0223

6. AUTHOR(S)

J. A. Ionata

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Undersea Warfare Center Detachment
39 Smith Street
New London, Connecticut 06320-5594

8. PERFORMING ORGANIZATION REPORT NUMBER

TD 10,889

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Program Executive Office (PEO(USW)-ASTO, Code E1)
2531 Jefferson Davis Highway
Arlington, Virginia 22242-5169

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This guide is intended to begin the process of standardizing the way in which algorithms are documented within the Advanced Development community at the Naval Undersea Warfare Center Division. The sections of this guide provide the outline for an Algorithm Description Document to be created for each algorithm. The format of the generated document should follow that of this guide as closely as is practicable.

14. SUBJECT TERMS

Software Engineering
Software Process Improvement
System Engineering

15. NUMBER OF PAGES

10

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

TABLE OF CONTENTS

FOREWORD

This guide is intended to begin the process of standardizing the way in which algorithms are documented within the Advanced Development community at the Naval Undersea Warfare Center. The sections of this guide provide the outline for an Algorithm Description Document to be created for each algorithm. The format of the generated document should follow that of this guide as closely as is practicable. An Algorithm Description Document should provide all the information necessary for a system implementer to incorporate the algorithm into the system being developed. Since the implementer is not likely to be an expert in the "science" behind the algorithm, this document must be as complete and as clear as possible. The Algorithm Description Document also provides basic information for test, integration, and maintenance activities.

The algorithm description should be independent of specific implementation details. Reliance on processor type, bus structure, memory / disk configuration, programming language and operating system should be avoided. Such specifics *are* appropriate for describing prototype implementations, however.

One of the more difficult design decisions will be choosing the appropriate level of detail for what we describe as a "single" algorithm. The nature of complex systems requires that we build systems from less complex, but certainly non-trivial, components. Similarly, most algorithms will be composed of steps that are more complex than basic binary operators. For example, a Finite Impulse Response (FIR) filter may be an algorithm worthy of its own Algorithm Description Document. Once defined, however, the FIR filter may be used as a basic step in a more complex beamformer algorithm. As a general rule, an Algorithm Description Document will be at the lowest practical level. Defining "algorithms" as low-level, primitive, functions gives the system implementers maximum flexibility and opportunity for re-use.

This document is set up in the format that must be used for preparing an Algorithm Description Document. It also includes a comment form. Your cooperation and feedback would be appreciated and would help make this guide as useful and realistic as possible. Please use the form at the end of this guide for your comments and suggestions.

ALGORITHM DESCRIPTION GUIDE FOR THE ADVANCED ENGINEERING FRAMEWORK IMPROVEMENT GROUP (AEFIG)

1. ADMINISTRATIVE INFORMATION

Section 1 identifies the algorithm, its origin and authors and gives credit to others who contributed to the work.

NAME OF ALGORITHM

This section identifies the common name of the algorithm. Choose the name most people would recognize the algorithm by.

AUTHOR(S) AND ORGANIZATION(S)

Identify the authors of the algorithm and their organizations, whether within the Naval Undersea Warfare Center (NUWC) Division Newport, Detachment New London, or outside. Where possible, provide other contact information such as telephone and FAX numbers, electronic mail addresses, etc.

DATE AND VERSION CONTROL INFORMATION

Ultimately, the Advanced Development community at NUWC will have its own, common configuration control system. Until such a system is available, use whatever project-specific version control information is available. For smaller projects, the author and date are usually sufficient. If the algorithm is from (or for) a larger system, some form of version number is often available.

ACKNOWLEDGMENTS

Give credit in this section to others (beside the author) who have contributed to the work. Also include whatever programmatic information is required by the sponsoring activity.

REFERENCES

Include references as appropriate to document the origins of the algorithm. Include NUWC publications as well as journal articles and textbooks that describe the theory behind the algorithm.

2. PURPOSE OF ALGORITHM

Section 2 describes the algorithm in terms of its purpose (what does it do?), its scope (how "big" a part of a system does it represent?) and its expected performance gains (why is this algorithm better than what is used now?).

FUNCTIONAL DESCRIPTION

Provide a functional description, in English, of **what** the algorithm does. **How** the algorithm does it will be described in detail throughout the rest of document. Augment the text with whatever diagrams or other graphical information that will help to make the purpose of the algorithm clear. [For details of how the problem is broken down into steps, refer to Process Flow in section 6 - Processing (Design).]

SCOPE

Describe the scope of the problem that this algorithm solves, as it relates to other algorithms. Does this algorithm do the complete job? Is it a special purpose piece of the answer that is intended to be combined with other algorithms? Or is it a general utility function that could be used within many higher level algorithms?

PERFORMANCE

Describe the expected performance gains of this algorithm over other ways of doing the same job. Is there a theoretical reason why this approach is optimal? What are the measures of effectiveness that are improved by using this algorithm over some other algorithm? Include any analysis that supports the conclusions drawn.

3. ASSUMPTIONS

Section 3 describes any underlying assumptions made about the implementation or environment in which the algorithm will operate.

IMPLEMENTATION DEPENDENCIES

Ideally, this section should contain the word "none." Realistically, there may be some assumptions made about the hardware or software that will implement the algorithm. Highlight the impact of violating these assumptions during the implementation of the system. For example, if an algorithm was written for parallel processor using a Single Instruction, Multiple Data (SIMD) paradigm, will it still work on a Multiple Instruction, Multiple Data (MIMD) machine if the individual processors are not synchronized?

ENVIRONMENTAL DEPENDENCIES

Identify any assumptions made about the physical or logical environment in which the algorithm will operate. Is the algorithm specific to a particular acoustic environment, mission phase, or tactical configuration?

4. INPUTS

Section 4 describes all inputs to the algorithm, including data and control signals.

IDENTIFICATION OF INPUT PORTS

Identify the various input streams ("ports") to the algorithm. Do controls and other data all enter from a common point in the processing flow, or are there multiple input points within the algorithm?

CONTROLS AND PARAMETERS

Identify any controls, signals, and parameters that affect the algorithm. (Include any status variables or condition flags set by other algorithms.) Are these items assumed to be static throughout the operation of the algorithm, or can they change? If changes occur, do they immediately interrupt the processing, or are changes allowed only at certain points? For each item, identify the associated input port.

INPUT DATA ITEMS

Identify each data item processed by the algorithm. For each item, identify the associated input port. Each item should be described in terms of the amount of data read, the type of data, the format of the data, and the accuracy required. Be as general as possible without violating any necessary assumptions. For example, do not specify IEEE floating point format if any numeric format with appropriate accuracy and dynamic range will do. However, if the algorithm takes advantage of a particular number format by doing bit-wise manipulation, state that clearly!

TIMING OF INPUT DATA

Describe the order and frequency of arrival of data and signals. Is the data periodic? Are there buffering requirements to ensure that data is processed in a certain order? Are certain data items synchronized with other items from a different input port?

RATIONALE FOR INPUTS

The reason for most data items is often obvious from the function of the algorithm. However, some choices of parameters or "ancillary" inputs are not so obvious. Describe the reason for including any input items that may cause confusion.

5. OUTPUTS

Section 5 describes all outputs generated by the algorithm, including data, special signals and controls.

IDENTIFICATION OF OUTPUT PORTS

Identify the various output streams ("ports") to the algorithm. Do controls and other data all exit from a common point in the processing flow, or are there multiple output points within the algorithm?

SIGNALS AND ERROR CONDITIONS

Identify any special signals, controls, or error conditions generated by the algorithm. (Include any status variables or condition flags set by the algorithm.) For each item, identify the associated output port.

OUTPUT DATA ITEMS

Identify each data item produced by the algorithm. For each item, identify the associated output port. Each item should be described in terms of the the amount of data generated, the type of data, the format of the data, and the accuracy.

TIMING OF OUTPUT DATA

Describe the order and frequency of the generated data and signals. Is the data periodic? Are there buffering requirements to ensure that data is output in a certain order? Are certain data items synchronized with other items on a different output port?

RATIONALE FOR OUTPUTS

The reason for most data items is often obvious from the function of the algorithm. However, some choices of signals or "ancillary" outputs are not so obvious. Describe the reason for including any output items that may cause confusion.

6. PROCESSING (DESIGN)

Section 6 describes the general design of the algorithm in terms of processing steps and the flow of execution.

BASIS FOR ALGORITHM

Describe the mathematical, physical science, or other basis for the algorithm. Point to available references (whose full citations are listed in section 1). Include information as to how the algorithm was determined to be "correct." (The implementation, or any future changes to the algorithm, may need to be evaluated according to these same criteria.)

PROCESS FLOW

Describe the sequence of operations that make up the algorithm. Use a graphical representation where possible. How is the overall problem broken down into procedural steps or objects? Be sure that any "primitive" steps used are well defined. (i. e. they should have their own Algorithm Description Document or be basic enough to be found in common, vendor-supplied math libraries.) For each step, describe the accuracy required (if it's important to the algorithm) and any other constraints, such as timing.

Describe the relationship among the steps. Is there a strict order in which the steps must be carried out? Think in terms of the possibilities for parallel and distributed processing. For example, if a calculation is performed independently on each of 100 beams, indicate "for each beam" instead of "DO BEAM = 1, 100" which implies order is important.

Describe the rationale for the process flow. Explain any choices which may not be obvious or which could cause confusion.

TIMING REQUIREMENTS

Explain any overall timing requirements that the algorithm must meet. These are most likely based on physical considerations or on keeping up with input data rates. Additional requirements having to do with integrating the algorithm into a system are the responsibility of the system implementers.

DATA CONSIDERATIONS

In general, choosing specific data structures is the responsibility of the software implementers. However, if particular data structures (objects, buffers, arrays, complex types, etc.) are important to the algorithm, explain the significance of each. Include any relationships involving data likely to be shared with other algorithms.

7. RESOURCE ESTIMATES

Section 7 describes the estimated amounts of computational resources required to support the algorithm.

PROCESSOR

How much "raw" processing power (in terms of instructions per second, floating point operations per second, or some other measure) is required to support this algorithm?

DATA STORE

How much storage space is required for data? Divide into primary (memory) and secondary (peripheral device) space if that's important to the algorithm.

PROGRAM STORE

Estimate the total size of the program store in terms of source lines code, memory size, or some other metric.

BANDWIDTH

Estimate the I/O bandwidth required to support the data flow among the various steps.

8. PROTOTYPE / MODEL IMPLEMENTATION

Section 8 describes any prototypes or model implementations of the algorithm completed as part of the algorithm development. Prototypes could include, for example, a non-real-time implementation or subset implementation on commonly available, commercial hardware. Such prototypes could be written in a "standard" language like C or using an environment like MATLAB. Model implementations include things like previous Advanced Development or Full Scale Developments of algorithms very similar to (but perhaps not exactly equivalent to) the algorithm being described.

SOFTWARE ENVIRONMENT

Describe the software environment used to implement the algorithm. Include information on programming language or tools, operating system, etc. If source code is available, provide information on where/how it can be accessed. Include specific version information if applicable. If necessary, provide any source code listings in an appendix.

HARDWARE ENVIRONMENT

Describe the hardware environment used to implement the prototype or model. Include the type and number of processors used, bus or network structure, memory, and storage devices.

SAMPLE INPUT AND OUTPUT

Provide information on sample inputs and output data sets used as part of the prototype / model efforts. Describe underlying assumptions and limitations related to the data. If necessary, provide listings in an appendix.

PERFORMANCE RESULTS (VERIFICATION)

Describe the performance of the prototype or model. How was it measured or tested? Describe any internal "test points" used to verify the prototype or model was correctly implemented.

CORRECTNESS OF PROTOTYPE (VALIDATION)

Describe any analysis done to "prove" the algorithm, as implemented in the prototype or model, is valid in terms of the basis for the algorithm (as described in section 6).

9. BIBLIOGRAPHY

This guide was compiled by using appropriate sections from the following documents:

[ANSI 830] American National Standards Institute/Institute of Electrical and Electronics Engineers Standard, ANSI/IEEE Standard 830-1984, "IEEE Guide to Software Requirements Specifications," 30 September 1983.

[ANSI 1016] American National Standards Institute/Institute of Electrical and Electronics Engineers Standard, ANSI/IEEE Standard 1016-1987, "IEEE Recommended Practices for Software Design Descriptions," 12 March 1987.

[DOD 2167A] Department of Defense Standard, DOD-STD-2167A, "Defense System Software Development," 29 February 1988, and associated Data Item Descriptions: Software Development Plan (SDP), DI-MCCR-80030; Software Requirements Specification (SRS), DI-MCCR-80025; Interface Requirements Specification (IRS), DI-MCCR-80026

[SAIC 94] Science Applications International Corporation, "Specifications for an Algorithm Description Document," Version 1.0, 3 October 1994. (Prepared for the Naval Undersea Warfare Center under contract N66604-94-C-1831.)

COMMENT FORM

Algorithm Description Guide
for the Advanced Engineering Framework Improvement Group (AEFIG)
Version 1.0

An updated version of the Algorithm Description Guide will be released in September of 1995. We welcome any comments which will help us to improve the document. Please provide your inputs via hard-copy or electronic mail using the information format provided below.

James A. Ionata
Naval Undersea Warfare Center, Code 2153
39 Smith Street
New London, CT 06320

Phone (203)440-4905
FAX (203)440-5987

email ionata@nl.nuwc.navy.mil

Name ______________________________ Date __________

Organization ______________________________

Street Address ______________________________

____

____

Email Address ______________________________

Telephone ____________________ FAX ____________________

Overall value:
( ) Excellent
( ) Good
( ) Fair
( ) Not Useful

Explanation:
____
____
____
____

General Comments on Document: ______________________________

____

____

Specific Comments on Sections:

Section:	Page:	Comments:
______	______	______________________________
______	______	______________________________
______	______	______________________________

INITIAL DISTRIBUTION LIST

Addressee	No. of Copies
Program Executive Officer (R. Zarnich, PEO(USW), Code E1)	1
Defense Technical Information Center	1