U.S. DEPARTMENT OF COMMERCE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NATIONAL WEATHER SERVICE
OFFICE OF SCIENCE AND TECHNOLOGY
METEOROLOGICAL DEVELOPMENT LABORATORY

MDL SOFTWARE DEVELOPMENT APPROACH

DRAFT
February 13, 2004

TABLE OF CONTENTS

MDL SOFTWARE DEVELOPMENT APPROACH

1.0 INTRODUCTION

This document defines a common approach for software development projects within the Meteorological Development Laboratory (MDL). This approach will be kept consistent with the emerging NWS Information Technology and Science and Technology directives. The approach defines guidelines for establishing documented and repeatable processes that are measured and optimized to increase product quality and development productivity. This document identifies development, testing, and management requirements for the approach and assigns responsibilities for meeting those requirements.

This document is based on experience with large and small software projects. Much of the experience is drawn from Advanced Weather Interactive Processing System (AWIPS) development and our work with government contractors like RS Information Systems (RSIS), Science Applications International Corporation (SAIC), and Northrop Grumman Information Technology (NGIT).

1.1 PROJECTS

The following MDL software development projects fall under the umbrella of MDL's Software Development Approach.

- AWIPS
  - Storm Scale Prediction
    - Hydrometeorological Monitoring
    - Watch/Warning Decision Assistance
  - Product Generation
    - IFPS Implementation and Enhancement
    - Digital Product Development
- Digital Database Development
- Statistical Update Development
- Storm Scale Prediction
  - Statistical Forecast Development
  - Advance Meteorological Applications
  - Operations and Software Support
- Forecast Evaluation
- Coastal Marine Prediction

2.0 REQUIREMENTS

MDL management, development, and support staff are committed to adopting, tailoring and implementing the following development requirements.

1. Standard Software Process Definition - Each project shall define a standard software process that adopts the following requirements. Each project can tailor its process to meet unique customer needs.

   a. Software Project Planning - Each project shall develop a software project plan that defines tasks, level of effort, staffing and schedule.

   b. Software Project Management - Each project shall track progress against the approved software project plan. Corrective action will be taken when milestones are not being adequately achieved. Any changes are made with the involvement of all affected projects.

   c. Requirements Management - Each project shall document the requirements that have been allocated to software and manage changes to these requirements throughout the life of the project. The allocated requirements are reviewed by project management and all affected projects.

   d. Reviews - Each project shall employ reviews to quality control internal development products (e.g., code walkthroughs, peer reviews), project management (e.g., project status), and for critical development checkpoints (e.g, design review).

   e. Documentation - Each project shall prepare documentation to communicate and archive valuable development information. This information is used by management, system integrators, users, developers and maintenance personnel.

   f. Testing - Each project shall establish a repeatable testing approach that effectively identifies and removes defects. The approach follows a bottom up approach, beginning with unit testing and proceeding upward as units are integrated into an application and system.

   g. Standards - Each project shall adopt software coding standards for the applicable software languages.

   h. Software Configuration Management - Each project shall implement software configuration management functions throughout the project's life cycle for all externally deliverable products and designated internal software products.

2. Standard Software Process Documentation - Each project shall document its software development process.

3. Standard Software Process Implementation - Each project shall adopt and follow the documented standard software process.

4. Process Improvement - Each project shall periodically review its software development process for strengths and weaknesses. Lessons learned will be identified and utilized to improve future development efforts. This review will be performed yearly and will be used to update the projects software development process. Metrics will be collected to measure effectiveness of the software development process.

5. Quality Assurance (QA) - Management shall verify that each project is following their software development process, standards, guidelines, and procedures throughout all phases of the life cycle.

6. Training - MDL management shall provide training to build the skills base of the development organization.

7. Hardware - MDL management shall maintain and monitor the computers and system resources necessary to allow each project to meet their development objectives and goals.

8. Waivers - MDL management shall grant waivers to requirements when it is in the best interest of the project or organization. Waivers are requested when a portion of the general requirement cannot be followed for a period of time. Occasionally, a permanent waiver to a general requirement may be necessary. Requests for waivers are submitted via the organization chain of management to the management. Any waivers are documented, reviewed by the Project Managers and approved by the MDL Director.

3.0 RESPONSIBILITIES

This section describes the responsibilities of project management within MDL for software development. Figure 3-1 shows the project organization and relationship to MDL management.

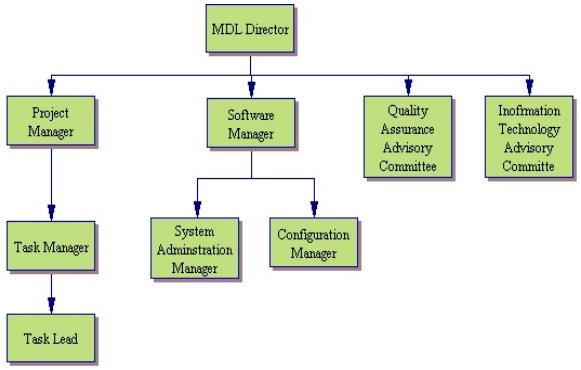

Figure 3-1. MDL Software Development Organization

Director, MDL - The MDL Director has overall responsibility for the software development effort.

Project Manager - The Project Manager is responsible for software process definition, documentation and implementation, planning and management, and process improvement. Within MDL, the MDL Branch Chiefs usually serve as Project Managers.

Task Manager - For larger projects, a Task Manager is assigned by the Project Manager. The Task Manager is responsible for project management and planning, and executing elements of the software development process (e.g., requirement management, software design and peer reviews).

Task Lead - The Task Lead is responsible for leading the development of individual tasks assigned by the Project/Task Manager. The Task Lead performs the requirement analysis and design, assists with coding, attends peer reviews of developers' code and test procedures, and coordinates the testing of the software.

Software Manager - The Software Manager is responsible for assisting the Project Manager define, document and implement a Software Development Process. The Software Manager ensures that each software project is managed according to its defined software process and conducts regular status reviews with the Project Manager. The Software Manager is responsible for the collection of metrics to measure performance and coordinates the activities to assess, develop, maintain, and improve these processes.

Configuration Manager - The Configuration Manager is responsible for ensuring that all externally deliverable products, designated internal software work products, and support products are managed in accordance with the project's software development process. The Configuration Manager will define and implement a configuration management plan which supports the software development process using configuration management tools to coordinate version and change management, and manage software releases. The Configuration Manager supports the Project Manager with status reports, but reports to the Software Manager.

System Administration Manager - The System Administration Manager is responsible for maintaining the hardware and necessary system support software that make up the MDL development and test systems.

Quality Assurance (QA) Advisory Committee - The QA Advisory Committee advises management on quality assurance aspects of the software development life cycle to ensure the production of high quality software and guidance products for our customers. The team is composed of one person from each MDL Branch.

Information Technology (IT) Advisory Committee - The Information Technology (IT) Advisory Committee advises management on the status of IT equipment and practices within MDL, and recommends improvements. The team is composed of one person from each MDL Branch.

Table 3-1 shows the primary responsiblity for the requirements in section 2.0 of this document. Table 3-2 shows the primary responsibilities for each MDL project.

Table 3-1. Requirement Responsibilities

Policy	**Responsibility**
Standard Software Process Definition	Project Manager/Software Manager
Software Project Planning	Project Manager
Software Project Management	Project Manager/Task Manager
Requirements Management	Task Manager/Task Lead
Review	Task Manager/Task Lead
Documentation	Task Manager/Task Lead
Testing	Task Manager/Task Lead
Software Configuration Management	Configuration Manager
Standard Software Process Documentation	Project Manager/Software Manager
Standard Software Process Implementation	Project Manager
Process Improvement Responsibility	Software Manager
Quality Assurance	Software Manager, QA Advisory Committee
Training	QA Advisory Committee
Hardware	System Administration Manager, IT Advisory Committee
Waivers	MDL Director

Table 3-2. Project Responsibilities

Project	**Project Manager**	**Task Manager**
AWIPS	Storm Scale Prediction - Stephen Smith Product Generation - Matt Peroutka	Tom Filiaggi Mark McInerney Jim Calkins
Digital Data	David Ruth	Tim Boyer
Statistical Update	Bob Glahn	Judy Ghirardelli
Statistical Forecast	Paul Dallavale	Paul Dallavale Kathryn Hughes Rebecca Cosgrove
Forecast Evaluation	Wil Shaffer	Valery Dagostaro
Coastal Marine Prediction	Wil Shaffer	Wil Shaffer
	Responsibility	**Name**
All Projects	MDL Director	Bob Glahn
	Configuration Manager	Joel Moeller
	Software Manager	Ed Mandel
	System Administration Manager	Steve Olson

4.0 SUPPORTING DOCUMENTATION

The MDL Software Development Approach consists of this document and the following companion documents.

MDL Software Development Processes - A compilation of documents which define the software development process for each project such that the unique needs of that project and customer are met. Each document describes the process and procedures used to analyze, design, code, test, and manage the projects entire software development life cycle.

MDL Software Standards, Guidelines and Procedures - A compilation of documents define standards, guidelines, and procedures to be used in the execution of the software development process. These documents are applicable to all MDL development organizations and cover software coding, documentation, review, testing, project management, and configuration management practices.

MDL Quality Assurance (QA) Advisory Committee Web Site - This website was created by the QA Advisory Committee to provide a wide range of services to the MDL development community. This site is a repository for all software development process documentation and provides links to developer training courses, configuration management practices, and system administration information.