12.4 THE RAPID PROTOTYPE PROJECT

T. LeFebvre*, M. Mathewson, T. Hansen

NOAA Research−Forecast Systems Laboratory

Boulder, CO

Todd Dankers

National Weather Service, Boulder, CO

1. INTRODUCTION

Over the past three years, staff from the Forecast Systems Laboratory (FSL) have been working closely with National Weather Service (NWS) field forecast offices in an effort to evolve as quickly as possible the grid−editing component of the Interactive Forecast Preparation System (IFPS). This component, called the Graphical Forecast Editor Suite (GFESuite), allows forecasters to define a weather forecast in gridded digital form. Once defined, the majority of NWS forecast products are then derived from this digital forecast database (LeFebvre 2000).

Prior to IFPS, forecasters produced most, if not all, of their forecasts in textual form. The paradigm shift from typing text products to expressing the forecast digitally represents one of the largest changes to the job of weather forecasting in decades. At the project's outset, very little experimentation had been done with respect to verifying whether such an approach would be viable in a forecast office.

To reduce the risk in implementing a system with IFPS's radically different paradigm, the NWS decided to deviate from its established software development, testing, and delivery process, and deliver software updates more frequently. Using this new methodology, a selected set of NWS forecast offices frequently communicate with the software developers. Forecaster's comments would then be incorporated into the software more quickly and the system could evolve in a much shorter time frame. This paper describes this rapid feedback/delivery process called the Rapid Prototype Process.

2. HISTORY

The grid−editing component of the IFPS project began in 1992 with a major specifications document that described the high−level functionality of the system. Once the infrastructure was built, the software developers invited forecasters from various regions of the NWS to travel to FSL, learn how to use the prototype, and then candidly comment on various aspects of the system including the user interface, ease of use/efficiency, meteorological soundness, and forecast methodology (Hansen et al., 2000). These valuable forecaster comments were then incorporated into the system design over the next 6−12 months, followed by another meeting where forecasters met with developers again to repeat the exercise and refine the system further. This process continued for several years. Developers incorporated the valuable feedback from forecasters and the system evolved in a positive direction.

The rate of progress, however, was relatively slow. Since meetings were held relatively infrequently (6 − 12 months), and because the grid editing system was never used in an operational setting, developers received little guidance from forecasters between meetings. The effort to integrate the grid−editing system with existing IFPS software further inhibited progress toward an effective grid editor. According to the NWS Strategic Plan, all NWS field forecast offices were expected to be producing a full suite of digital products by fall 2003. It became clear to many NWS managers that this goal would not be met if progress continued at the slow pace of the mid to late 1990s.

In June 1999, management representatives from each NWS region, NWS headquarters, as well as representatives from the software development organizations met in Norman, OK to focus on a plan that would accelerate the current rate of progress. At the meeting, attendees decided to implement a new experimental approach to software development. In this approach, FSL developers would release new software every 6−8 weeks instead the current practice of every 6−12 months. Closer communication between forecasters and developers would take place via telephone conference calls, an e−mail bulletin board, and periodic face−to−face meetings. NWS managers and developers believed that this enhanced communication and rapid delivery of software would evolve the GFESuite more quickly from a prototype to an operational system. The name chosen for this experimental activity was the Rapid Prototype Project (RPP).

1 * Corresponding author's address: Tom LeFebvre, NOAA Forecast Systems Laboratory, 325 Broadway R/FS6, Boulder, CO, 80305;
e−mail: lefebvre@fsl.noaa.gov

3. RAPID PROTOTYPE SOFTWARE

The RPP began with seven sites that accepted the prototype software, exercised it quasi−operationally, and provided feedback on a number of aspects. One site was chosen from each NWS region, and one from a national center, the Hydrometeorological Prediction Center (HPC) of the National Center for Environmental Prediction (NCEP).

An e−mail bulletin board (called the listserver) was established to allow anyone who registered to post questions and/or answers to technical problems. In addition, the listserver was frequently host to many forecast methodology discussions that steadily refined the software in a direction consistent with forecaster's requirements. In addition to the listserver, face−to−face meetings between forecasters and developers further enhanced information flow.

3.1 RAPID SOFTWARE DEPLOYMENT

One of the critical ingredients to RPP was rapid software deployment. Approximately every 6−8 weeks, FSL developers delivered a new software release by mailing a CD−rom disk containing the software to each RPP site. To accommodate offices that were interested in receiving these updates sooner, the software was also posted to a web−site for immediate download. Once received, forecasters serving as the RPP focal point installed the software on a local computer for review.

Delivering new software in this rapid fashion offered several advantages over waiting months between new releases. 1) Forecasters remained motivated to improve the system when they saw their suggestions implemented within weeks. 2) As new software bugs were discovered by field forecasters, developers made patches available that fixed the problem, usually within days. The Python programming language greatly helped in this manner, since it is a scripting language where no compiling or linking of the software is required. This feature of Python made field installation of patches as simple as installing a file in the appropriate location and restarting the software. 3) Immediate fixes to problems in the field allowed forecasters to quickly continue their informal evaluation work without having to wait weeks or months before a problem is corrected.

3.2 FORECASTER FEEDBACK

Rapid feedback from forecasters provided the catalyst needed to quickly deliver new features. Armed with copious comments from the system users, developers confidently made changes consistent with forecasters wishes. As new reports of bugs or suggestions for new features were made available, developers carefully recorded and organized this feedback into a bug report /change−request management system.

In addition to this spontaneous forecaster feedback, FSL's Evaluation Team occasionally issued a web−based survey to RPP participants that posed questions and invited comments on a diverse list of RPP subjects. (results of one of these surveys can be viewed at: http://www−md.fsl.noaa.gov/eteam/focal_point01/ rpp_2001_cat.html)

3.3 COMMUNICATION

In addition to the e−mail−based listserver communication, participants in the RPP met once per month via telephone conference call. At these calls each site typically reported their status along with any problems they were currently experiencing. In addition, RPP members prioritized tasks derived from the listserver feedback collected previously, factoring in an estimated delivery date for each task. Tasks were prioritized by votes cast by each RPP participant. Priorities derived from a consensus of the group were then recorded by the development team. This consensus priority was then used by developers to determine the order in which the tasks would be implemented.

Approximately every 6−8 months, RPP field participants would gather to meet with the developers over a period of several days to discuss requirements, refine existing tools, review new forecasting techniques, and brainstorm new ideas. These meetings typically began with a series of training exercises designed to educate each forecaster on new system features, tools, and user interface issues. With this knowledge, developers and forecasters then discussed ways in which these new features could be improved in order to make the forecast process more efficient and intuitive. Open discussion sessions allowed forecasters to present and explore new ideas. As with the listserver and monthly conference calls, results of these sessions were recorded and converted into tasks. These new tasks were added to the master list of tasks and prioritized by all RPP participants.

In general, each new software version focused on a particular feature of the overall system. This limited the scope of new changes so forecasters could better concentrate their efforts on particular areas of the system and review them.

4. HUMAN FACTORS

The RPP activities outlined above are essential ingredients to rapidly building a suite of software that meets the needs of forecasters. However, these activities alone are not sufficient. Factors such as skill level, team dynamics, motivation, and partnership are equally important for a process such as this one to be successful. In this section we explore some of these attributes we call "human factors".

4.1 DEVELOPMENT TEAM CHARACTERISTICS

One of the most important attributes of the RPP development team at FSL is that each member participates in all phases of software development from gathering requirements, designing the software, coding, testing, and providing support for the customers that ultimately use the software. Team members also write documentation and lead training sessions for the software they have designed and coded. With this continuity, the developer becomes aware of the context within which the software is used, resulting in a system that better meets the needs of the forecaster. This, in addition to the variety of challenging tasks, provides a high level of job satisfaction for the team members.

This "wearing all hats" approach requires a relatively high level of skill in the areas of software engineering and, in this particular case, the science of meteorology. FSL's development team includes experts in software engineering as well as meteorologists. The members interact to share their primary skills allowing developers to successfully translate the forecasters' requirements into code.

The RPP development team is diverse in other important ways. Some of the members are oriented toward the "big picture" while others tend to focus on the details. This adds depth and balance to the team. Some members like to design software using visual design tools such as the Unified Modeling Language (UML). Others are more comfortable with an "Extreme Programming" approach in which code is developed in small increments and the design evolves with each iteration (Hansen 2000). While the team has adopted guidelines for design and coding, few are rigidly enforced, providing a general acceptance of different operating styles. A member is free to adopt an appropriate style for each task and to operate in a way that is effective for that member and task.

FSL developers have adopted a team approach to development throughout the RPP process. Willingness by team members to listen to all ideas and then constructively critique them frequently leads to software designs that are more robust and tolerant to changing requirements. This approach is facilitated by a small team size (6−8 members) and a team leader who is also a member of the development team.

4.2 PARTNERSHIP BETWEEN DEVELOPERS AND FORECASTERS

A fundamental principle of the RPP is that the developers and forecasters share a common goal: to build a system that provides the forecasters the tools they need in order to meet the goals as defined in the NWS Strategic Plan. This common goal has fostered a partnership between the development team and NWS forecasters, who are equally responsible for the success of the RPP. The forecasters in this group have a high level of expertise and are in key positions within their forecast offices. They are highly motivated and talented individuals who have taken initiative to ensure success in the transition to digital forecasting.

Forecasters at RPP forecast offices have participated in the development process through a variety of activities. Field forecasters have used the smart tool infrastructure contained in the GFESuite to develop smart tools that aid forecast methodologies and improve the efficiency of those using the software. Through early October 2002, 187 smart tools and procedures have been posted on the NWS smart tool repository. (These smart tools may be viewed and downloaded at the web site: http://isl715.nws.noaa.gov/str). A component of GFESuite generates gridded surface weather elements from gridded numerical models (LeFebvre et al., 2002). Forecasters have identified and fixed deficiencies and provided enhancements to many of these model−based algorithms. Forecasters have made invaluable contributions to the GFESuite's text product infrastructure in developing various tabular and narrative alpha−numeric product formatters.

The development team is well−equipped to support forecasters with operational problems in the field. When each member of the team participates in developing software, they are better able to respond to problems that may arise. Using the listserver, forecasters have been willing to report any questions and problems they may be having. Many have submitted new ideas and issues for discussion. The development team welcomes all feedback knowing that it will lead to a better system. There is a high level of respect for the field forecasters and appreciation for their contributions. All listserver messages are answered promptly, further motivating the forecasters to give feedback.

There have been several occasions where an individual struggling with a system problem, used the listserver for the first time submitting a message in the form of a frustrated complaint. As with every listserver message, these were responded to promptly and, the problem was solved immediately, or, if that was not possible, a course of action was outlined. The individual invariably was appreciative and, henceforth, became a partner in identifying ways in which the system could be improved.

5. RESULTS

The RPP began with seven field sites, one from each region and one from NCEP. The focal points at each site installed the software locally and trained the staff to use the GFESuite software quasi−operationally. Focal points from many offices developed "job sheets", a set of specific instructions that provided forecasters possessing limited GFE experience with enough guidance to generate a limited set of experimental graphical products. As forecasters from each site gained experience, the number of these experimental products increased.

Observing the progress made at these original sites, 10 more forecast offices volunteered to be experimental RPP sites within a year bring the total to 17. Each new site brought with it a subtly different area of expertise. After the number of sites using the GFE more than doubled, the amount of feedback generated from the field increased proportionally. This feedback gave developers a wealth of information to further improve the system. Months later, new sites joined the RPP unofficially, training their staff and providing comments to the FSL development team. Eventually, any site that wanted to use GFESuite was allowed to do so.

During the first three years of the RPP process, FSL has delivered more than a dozen software releases to RPP field sites. Because of forecaster feedback, the development team identified and fixed more than 400 software bugs and performed over 700 enhancements including intersite coordination of forecast grids, smart tool framework, text formatter framework (Hansen et al., 2003), and daily forecast critique.

6. CROSS−POLLINATING THE RPP

Some have asked how the RPP can be applied in other project settings. The success of the RPP depends equally on project structure and human attributes. We have described the project structure which includes rapid deployment, rapid feedback, and communications mechanisms such as the listserver.

We have also identified some of the "human factors" that we believe are critical ingredients to success. These factors should be considered when building an RPP team. However, because of the complex interaction of these "human factors", building a successful team may take time. Thus, it might be advantageous to keep effective RPP teams intact to be "re−used" in other situations.

7. CONCLUSION

According to the majority of RPP participants, RPP has been a success. One forecaster commented in one of the training workshop surveys, "The RPP process should be a model for NWS software development and integration into operations. It works very, very well." Another RPP participant was impressed with the rate of progress stating, "I'm excited about the future of IFPS because of the great progress that we have made over the past few years."

What began as experimental prototype software rapidly evolved into a robust, reliable suite of code that met the needs of forecasters primarily because they played an important role in its development. Rapid software deployment coupled with rapid feedback provided an environment in which experimental techniques quickly evolved into dependable forecast tools and methodologies.

However, the RPP process is no substitute for a thorough software process that includes requirements gathering, thoughtful design, and extensive testing. These activities are essential in order to develop reliable software that works well for its users.

8. REFERENCES

Hansen, T. L., T. Dankers, C. Paxton, 2003: Text Formatting with the Graphical Forecast Editor. Preprints 19th International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Long Beach, CA, Amer. Meteor. Soc., in this preprint volume.

Hansen, T., 2000: Software approaches for the GFESuite. FSL Forum, N. Fullerton, Ed. Forecast Systems Laboratory, 12−15.

Hansen, T. L., M. Mathewson, M. Romberg, 2000: Forecast Methodology using GFESuite. Preprints, 17th International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Albuquerque, NM, Amer. Meteor. Soc.

LeFebvre, T. J., M. Mathewson, T. Hansen, M. Romberg, 2002: Initializing Gridded Fields from Numerical Models. Preprints, 17th International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Albuquerque, NM, Amer. Meteor. Soc.

LeFebvre, T., 2000: Designing digital for better products and services. FSL Forum, N. Fullerton, Ed. Forecast Systems Laboratory, 24−27.