

INTERACTIVE TROUBLESHOOTING GUIDE FOR ACCELERATORS

R. Legg, B. Dunham, S. Higgins, R. Kazimi, D. Kehne and T. Oren

Thomas Jefferson National Accelerator Facility, 12000 Jefferson Ave., Newport News, USA

0205 9100 016 490 4

Abstract

Modern accelerators face the challenge of supporting increased machine complexity and higher levels of utilization while relying on downsized maintenance and support groups. To improve system availability, reduce reliance on system experts and provide a systematic approach to problem solving, an interactive troubleshooting guide has been implemented for the CEBAF injector at Jefferson Lab. This guide, which uses a tree structure with appropriate hypertext links, traces problems from a set of "symptoms," through a series of diagnostic tests, to a specific corrective action. This guide is used in conjunction with an active parameter monitoring system that is part of the EPICS control system tool kit. The monitoring system generates operator alarms when injector RF or magnet parameters fall outside pre-set windows. Operators receive immediate feedback when injector components vary from their standard values and the troubleshooting guide provides a systematic means to identify and ultimately correct the problem.

1 INTRODUCTION

CEBAF, the Continuous Electron Beam Accelerator Facility[1], is a five-pass, recirculating, superconducting RF linac designed to provide a 4 GeV, 200 μ A CW beam of exceptional quality. It is made up of an injector, two 400 MeV linacs and nine recirculation arcs having a total beamline length of more than 4.5 kilometers. There are presently over 40,000 control points and more than 120,000 database records in the control system[2].

During commissioning of the injector, problems affecting the electron beam parameters occurred on a regular basis. In order to minimize disruptions to the program, these problems had to be corrected as quickly as possible. To address this need, a group of personnel with injector expertise was assembled and placed on-call 24 hours a day on a rotating basis. Once the commissioning phase was completed, however, the use of accelerator physicists to troubleshoot routine injector problems proved to be an intolerable drain on resources. A way for the operations crews to troubleshoot problems without the intervention of system experts was needed.

The requirements for the system were that it should not require any special knowledge of beam dynamics, it should be easy to use with a point-and-click interface and hypertext links, and it should have an underlying structure that would guide the user from a specific symptom through a set of tests that would help identify the specific problem and, finally, to some corrective action. The goal of the

guide was not to eliminate the experts, but to answer the most frequently asked questions that account for 80-90% of the injector problems.

2 THE PROCESS

The first step in the creation of the troubleshooting guide was to define a set of symptoms that could then be passed on to the original on-call personnel for "solution." To gather the symptoms a search of the Daily Activity Log, the log book kept by the machine operations group, was conducted. This search led to three broad categories of problems commonly encountered in the injector. 1) an irregular or 'bad' spot on the beam viewer at a dispersed point immediately prior to injection into the first linac, 2) an injector orbit that was offset by more than ± 0.5 mm, and 3) "Something's wrong with the injector"—a generic problem category. This last problem usually indicated an odd beam envelope downstream from the injector, suggesting that the injector beam phase space was no longer matched into the first linac.

The second step in the process was to pass the list of "symptoms" out to the experts, who then generated flowcharts that specified a set of diagnostic tests to determine the possible cause for each symptom. As a cross check of the experts, an additional search of the Daily Activity Logs generated a list of actual injector problems that had been encountered in the control room and how they had ultimately been solved.

The next step was to take the flowcharts for diagnosing the symptoms and break them down into steps while seeking commonality between the different flow charts. Once identified, these common elements could be written only once and then, using hypertext links, shared between the various diagnostic tests. This shared module approach led, however, to one long term problem: it made the overall "tree" a hyper-object with a structure that was not intuitively obvious without a diagram. As different people made changes to the tree to upgrade the guide or reflect changes in the injector itself, over time, the inconsistencies grew in number. This non-intuitive structure made maintenance of the guide difficult for new authors assigned to the task.

The result of the flowcharts was a list of the precise procedures needed to diagnose the symptoms. Many of the procedures already existed among the 100 or so that had been written for the commissioning effort, although most were not in the form needed. In those cases, the directories of procedures were searched and the appropriate documents were rewritten. In some cases, new software was written to allow the required tests to be done more easily. Some of

the procedures for diagnostic tests were written from scratch. Once written, these procedures went through a formal review process by the operations department and were corrected to reflect the recommended changes. As these modules were completed and corrected, they were collated for use in the final tree structure.

At this point, we had a set of written procedures within a tree framework leading from a set of symptoms to possible solutions. Finally, it was time to decide on a delivery system for the troubleshooting guide—a way to get the information to the end user in the control room. We needed a system that allowed hypertext pointers from one document to another and a point and click interface that would be easy for the operators to use. The initial thought was to use the Worldwide Web and HTML to post the troubleshooting guide as a series of linked pages; however, the setup procedures written during the commissioning phase were in Framemaker^o and a translation to HTML format would be very time consuming and make updating the documents more cumbersome. If the original procedures were not translated, however, they would not be available to a web browser, and if the operator needed the original procedure they would be forced to open the original word processing application.

While the problem of what document delivery system should be used was studied, a preliminary test of the troubleshooting guide was performed using HTML. A copy of the tree was drawn out by hand and the corresponding HTML files were set up on a local web server. This system was deliberately very crude, with no graphics and little discussion, but it did provide the hypertext pointers and the basic Symptom-to-Diagnostic Test-to-Solution structure. This allowed us to look for dead ends and inconsistencies in the procedures that didn't show up while looking just at the tree or at the procedures alone. This test did not, however, settle the question of how to implement the final version and also raised the issue of change control for the pages in the guide. During the creation of the troubleshooting guide, multiple authors worked simultaneously, with multiple updates occurring in some instances.

As the prototype system was going through testing, another group was working with the Alarm Handler, an EPICS tool, to generate a system that would passively monitor the signals generated by the injector and send the operators an audible and visual signal when the injector parameters went out of tolerance. The Alarm Handler was already being used to monitor the major accelerator systems during operation, and in its simplest form provided a set of window comparators looking at control signals defined within a tree-like hierarchy. When a signal moves outside a user-defined range, the Alarm Handler trips and the fault propagates up the tree to the top node and sets off an alarm. The injector Alarm Handler differs by comparing the signals in the injector to a pre-defined set of values. It ignores the question of whether the hardware is working properly and is only concerned with whether the signals are

at a preset value. The allowed beam orbit through the injector becomes an alarm parameter, along with the magnet and RF settings. The idea was not only to provide a troubleshooting guide to solve problems after they occurred, but to give the operators a monitoring capability that would alarm as the beam or the hardware began to drift away from the nominal settings. This system was implemented and has worked well within its limitations.

3 IMPLEMENTATION

The tests of the original HTML implementation of the troubleshooting guide revealed several problems with using the HTML format. There were questions about the stability of HTML as a language, since vendors freely added their own extensions to the language and these extensions were not serviced by all web browsers[3]. There were also concerns about document management, particularly with the large number of image and text files required to create the HTML pages. There was also the issue of control system security if operators were going to access the Worldwide Web from X terminals on the controls subnet. If the guide was made local to the subnet, then no external browser could view it. Finally, in the spring of 1996 the available tools for generating HTML were not well suited to such a large word processing task and long term maintenance of the raw HTML files was seen as a potential problem. To address these concerns, Adobe Acrobat^o was adopted to act as a delivery system.

Acrobat is freeware published by Adobe Systems, Inc. and is widely used to distribute on-line documents. This conference's proceedings, for example, will be published using Acrobat. Acrobat runs on most platforms. It handles Portable Document Files (PDF)—an Adobe standard—that are created from any Postscript file using an Acrobat distiller[4]. Acrobat supports hypertext links within the document and uses a controlled file standard. FrameMaker and Acrobat are a powerful combination; hypertext links and cross references created in FrameMaker are automatically converted to hypertext links in the PDF files during the Postscript-to-PDF conversion process. The on-screen version of a PDF document looks just like the final printed version, allowing for off-line review and use of the document. Basic document control elements such as page numbers and header and footer information match the original document, and the fidelity of graphics and type fonts is also maintained. In addition, Acrobat acts as a plug-in for all major web browsers, allowing PDF files to be read as Worldwide Web documents (see Figure 1).

One by one, Acrobat addressed our concerns. Since it uses a proprietary controlled file specification (the PDF file format), unlike HTML, there is no question about the long term stability of the file format and how it will be "extended" by other vendors. Acrobat is widely accepted as an emerging standard in the on-line publishing industry. It leverages the popularity of the Postscript page description language, and provides a convenient, reliable means of converting existing graphically rich document

files for on-line display, without having to convert to

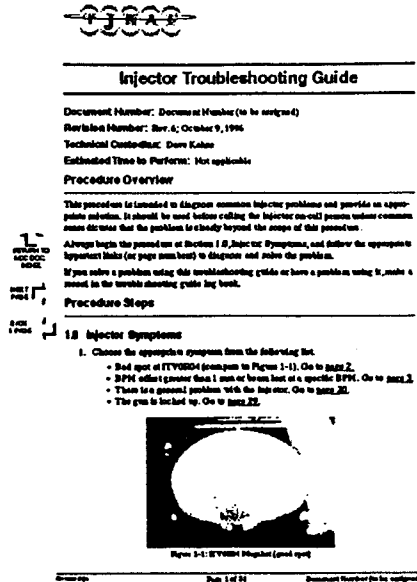


Figure 1

HTML. Document control becomes much easier because one person controls the distiller that converts the files into PDF format. This person acts as a final quality assurance step in the documentation creation and revision process and provides a level of document control by tracking revisions and maintaining consistency between documents.

The authors have access to the original word processor documents in their original format and thus can use familiar, powerful document editing and formatting tools when document revisions are necessary. When changes are needed, the original document is updated and then reconverted to a PDF file—a short, very predictable process that avoids the unpredictable, labor-intensive conversion to HTML. The question of network security is also no longer a problem since Acrobat runs in a stand alone configuration on the controls network. These same PDF files can also be viewed remotely from a web browser. The only thing passed between the networks are the read-only PDF files.

4 PROBLEMS

There are two types of problems associated with the injector troubleshooting guide and the injector Alarm Handler: 1) the elements in the injector do not reproduce, and 2) the injector itself changes. The first problem is

typified by the performance of a 100 keV 90° bending magnet. This magnet runs at such a low field that it is very difficult to remove hysteretic effects at the 0.01% level necessary to insure that the orbit through the magnet is reproduced on a day-to-day basis. This error, though small, changes the downstream orbit, which then sets off the Alarm Handler. To prevent this, the values in the Alarm Handler are updated when the device setpoints are changed, and efforts continue to improve the reproducibility of the devices.

The second problem occurs as a result of ongoing work to make the injector more useful and reliable, such as the addition of a polarized electron source. Such changes can fundamentally change the tree structures of the troubleshooting guide and, since no single person is responsible for maintaining the troubleshooting guide, there is a need for periodic general updates of the document. As changes are made to the underlying hyper tree, it is important to carefully consider how these changes will affect the procedure as a whole. Past difficulties with maintaining the troubleshooting guide have lead to the creation of an overall flow chart of the document. Though it is never seen by the operators using the troubleshooting guide, the flow chart is maintained as a way to identify potential problems with the hypertext link structure and simplify document maintenance.

5 RESULTS

The injector Alarm Handler and the troubleshooting guide have proven to be a very successful combination and have reduced the amount of time spent troubleshooting the thermionic injector. This trend is evidenced the fact that the number of physicists supporting the thermionic injector has been cut in half without affecting injector performance and reliability.

REFERENCES

- [1] Work supported by USA DOE#DE-AC05-84ER40150
- [2] K. White, D. Kehne, "Operational Experience with the CEBAF Control System", Proceeding of ICALEPS 1995
- [3] Jesse Berst, "Three Overlooked Problems That Threaten the Internet", ZDNet AnchorDesk, Nov. 25, 1996
- [4] Acrobat Distiller, Adobe Systems Incorporated, 1585 Charleston Road, Mountain View, CA 94039