Review of Advanced Perl by M-J Dominus
by Adam Turoff
1/17/02

I was working on a very thorough set of answers for this manuscript. Unfortunately, I didn't finish it last week, nor last weekend, nor earlier this week. I'm leaving for London in a couple of hours, so here's a tidbit to tide you over until I can complete the more extensive set of answers.

The short version is that I liked it a lot; it's one of the most interesting, most complete and most approachable pieces of technical writing I've come across in a very long time. I look forward to seeing the entire manuscript as soon as possible.

Content and Organization
1. Please examine the revised table of contents. Do you agree with the ordering of topics and the relative emphasis? Please be specific.

The TOC is well-done. The only issue I have is the size of the finished manuscript if all of these topics are to be covered. I think it would benefit readers if the book could be made available in two significant chunks, or possibly omitting chapters 10 (perhaps), 11 and maybe even 13. I don't understand what chapter 12 is supposed to cover, so it is difficult for me to see if this would qualify in the same category.

This material is very useful, and I expect that a large portion of readers will get the majority of their value out of the first nine or ten chapters; the remaining three-four chapters are quite interesting from an academic point of view, but are likely to be used rarely in practice.

Again, the primary reason for cutting back on the scope of the book would be to bring it to market faster, not because the more advanced topics covered towards the end of the outline are any less worthy of being discussed.

Chapters 1-4
For each of these chapters please consider and answer the questions below.
1. How well are the concepts developed ?particularly for the reader who may come to this material with no previous experience with these sort of techniques?

For the most part, the concepts are built up from basic principles in a manner that is easy to grasp for the typical reader. This manuscript is refreshing in the fact that it does not dwell overmuch on classic examples from computer science and discusses real world problems that regular programmers can comprehend -- such as parsing HTML, spidering the web, and scouring filesystems.

Occasionally, the discussion of classic computer science problems finishes too quickly, or isn't presented with enough background material. For example, the very first problem in the book discusses binary representations of integers. It does not discuss even briefly what binary representations are, nor why this is an interesting problem to solve. The reader is expected to infer the nature and value of the problem being solved, and understand the recursive nature of the solution being explained here. A little more explanation leading up to this problem would help here.

[Compare this to the discussion of RGB -> CYMK, which is handled much better.]

Also, the towers of hanoi problem is revisited after it is solved, almost apropos of nothing; the way the problem resurfaces (when a relevant technique is being discussed) is quite jarring and interrupts the reader's flow of thought. (It took me a few seconds to connect the dots.)

2. Are the illustrations and examples clear and concise? Are there sufficient numbers or types?

The examples are reasonably clear. The tower of hanoi problem could benefit from a few diagrams. None of the examples were overdone or exceedingly lengthy.

The discussion of iterators was quite long, and might benefit by being split into two chapters. Also, particularly with Chapter 4, the same basic problem is discussed so many times, with so few lines changing each time, I believe it would help the reader if the entire solution were presented periodically (or at the end of the chapter) to see exactly what is changing to produce which iterator behaviors.

3. Please evaluate the level-are they too difficult or too easy? Do they progress in difficulty in a measured and understandable way?

The basic examples refer to classic problems in computer science. While this is unfortunate (especially when trying to reach a lay audience not well-versed in these classic problems), the discussion is presented in a manner that de-emphasizes the mathematical nature of the problem and focuses on the practical nature of the solution. This approach is quite unique, and should help elluciate these topics to an audience that has never fully understood them before.

That said, the level of the material is never too easy, and is only occasionally too difficult -- mostly through lack of exposition and explanation. For example, the discussion of factorials tries to explain combinations and permutations in a manner that the average non-mathematically inclined reader can understand. But the discussion of fibbonaci numbers isn't quite at the same approachable level – all of the examples are there, but something just isn't clicking the same way that it clicked for factorials...

Iterators are a very difficult concept to get a cross, yet the level at which they are being discussed is quite easy to understand. So much so that I finished the chapter wondering why more Perl programs aren't written this way.

4. Is the chapter technically accurate and up to date? Are there concepts that require further illustration or demonstration?

The topics that require further illustration tend to be earlier in the chapter, when the solution to purely mathematical problems is being discussed.

Iterators are a very deep topic, and could probably be expanded and split into two chapters; I don't know where this split should occur though, other than in the middle.

There were a couple of fragments in the manuscript that need to be filled in: there is no discussion of state machines in the chapter of dispatch tables (something I've heard Dominus speak about at length).

5. On a scale of 1-10, with 10 being the highest mark, please include an overall rating for the chapter.

Chapters 1-4: 8.25

(a few loose ends need to be closed, but if it were printed as it stands today, it would be an incredible asset to the domain of Perl books)

Writing

1. What is your assessment of the authors' overall writing style? Any particular suggestions here?

Overall, very approachable without being preachy or condescending. Quite easy to understand topics that are typically reserved for advanced study in computer science curricula.

2. Please describe your overall reaction to the manuscript.

Very favorable. I can't wait to see this book in print.

Thank you for the opportunity to read this manuscript.

Z.