
Abstract
ET: Enhanced tool (solution strategy)
The effectiveness of <heuristics for guiding the programming process> in supporting the design of <software> has been demonstrated. An enhanced tool / method is described for the design of <software> based on <semi-automatically transforming the specification into an implementation>. Examples are provided confirming the effectiveness of its support for <software development> in design.

Why ET?
The authors note several extant “conceptualizations” (actually heuristics or mindsets) that have improved programming craft in the past. They then build on these conceptualizations to produce “Transformational Implementation,” which involves representing the specification in a computer-understandable format, and then (under the guidance of a human) incrementally transforming the specification into an optimized implementation.

Question - [Method/means of development]
<model-type / solution strategy>
What is the most efficient way of producing an efficient, bug-free implementation of a specification?

<artifact-type>
The artifact being produced is a software implementation. The question is the best way to ensure that the implementation is correct and optimized.

Results - [Procedure / technique]
<model-type / solution strategy>
The authors propose to represent the specification in a computer-readable format which can be successively transformed into an actual implementation. The programmer is responsible for identifying the areas to be transformed and the transformation to execute.

Validation – [Example]
<model-type / solution strategy>
According to the authors, this new approach will help to separate optimization from implementation, thereby allowing the best possible execution of each by programmers. They attempt to demonstrate this on a toy example, a piece of software that effects some changes on a piece of text. The authors have not implemented a real tool to support the necessary transformations, nor have they indeed even yet built a catalog of all the transformations that such a tool would need to support. Consequently, the toy example leaves a feeling that the authors’ proposed approach is still in a formative state, and that the approach would benefit from additional development and validation.