1.6 Use of color in this standard

This standard uses a minimal amount of color to enhance readability. The coloring is not essential and does not affect the accuracy of this standard when viewed in black and white. The places where color is used are the following:

- Cross references that are hyperlinked to other portions of this standard are shown in underlined-blue text (hyperlinking works when this standard is viewed interactively as a PDF file).
- Syntactic keywords and tokens in the formal language definitions are shown in **boldface-red text**.

1.7 Contents of this standard

The organization of the remainder of this standard is as follows:

- Clause 2 provides references to other applicable standards that are assumed or required for this standard.
- Clause 3 defines terms, acronyms, and abbreviations used throughout the different specifications contained in this standard.
- Clause 4 describes the fundamental syntactic and semantic components of the ***e*** language.
- Clause 5 defines ***e*** data types and describes their usage.
- Clause 6 defines how structs, subtypes, and fields function within this standard.
- Clause 7 describes the constructs used to define units and their use in a modular verification methodology.
- Clause 8 describes the principles and usage of ***e*** templates types. Template types in ***e*** define generic structs and units that are parameterized by type. They can then be instantiated, giving specific types as actual parameters.
- Clause 9 defines ***e*** unit interfaces and their usage within this standard.
- Clause 10 defines test generation and constraint functions within this standard.
- Clause 11 defines how temporal constructs can be used for specifying and verifying behavior over time in a ***e*** program.
- Clause 12 defines the syntax and semantics of TEs and describes their usage to track temporal behavior.
- Clause 13 defines two other struct members that can be used for temporal coding of events.
- Clause 14 defines what time-consuming actions are and how to use them.
- Clause 15 describes how to define, extend, and use coverage constructs.
- Clause 16 describes how to modify the grammar of the ***e*** language.
- Clause 17 shows the various constructs that print an expression, check for errors in the design under test (DUT), or add exception handling and diagnostics to an ***e*** program.
- Clause 18 defines how to declare and use ***e*** methods within an ***e*** program.
- Clause 19 describes how to create and assign values to ***e*** variables.
- Clause 20 shows how to perform basic packing and unpacking of scalars, strings, lists, and structs in this standard.
- Clause 21 describes how to use ***e*** control flow actions.
- Clause 22 defines how to create dependencies between ***e*** files and use the preprocessor directives in this standard.
- Clause 23 contains the syntax and descriptions of the ***e*** statements used to create packages and modify access control.