



San Juan Software Coding Standards

Coding Rules

- ✓ All source code is ANSI C compliant.
- ✓ Source code is written in a consistent style throughout the product.
- ✓ All source code compiles without errors or warnings.
- ✓ Care is made to write clear, understandable, unambiguous code.
- ✓ Comments reflect the design of the code; they are clear and concise.
- ✓ All function, type and global variable names are chosen so as not to pollute or conflict with the global namespace. See Naming Conventions.
- ✓ All constants are defined with a macro. No magic numbers.
- ✓ Header files are not nested.
- ✓ Curly braces { } are on a line by themselves. Code is indented one tab-stop and one line below the curly brace. Tab-stops are 4 columns.
- ✓ All macro parameters use parenthesis () to avoid unexpected behaviors.
- ✓ All files include the standard San Juan Software copyright header.

Naming Conventions

- ✓ All variables of global scope begin with some variant of “TFS” to avoid polluting namespace.
- ✓ Variable and function names are meaningful; functions reflect actions, variables are nouns. Examples: TfsFormatDrive() is a function, iTfsBlockSize is a variable.
- ✓ Function names, including macros, are mixed case and begin with Tfs. Example: TfsOpenFile.
- ✓ Variable names and types are mixed case and begin with their type. Example: iBlockSize.
- ✓ Types are indicated with “i” for integer types, “p” for pointer types and “s” for structures and unions, “b” for byte and “f” for flags, “g” for globals. Example: giTfsWhichMetaRoot.
- ✓ Functions indicate which subsystem they belong to by their prefix. Example: TfsIMapBranchBlock() belongs to the iMap subsystem.
- ✓ Macro constants are upper case and begin with TFS. Example TFS_META_FIELDS.

For more information, visit the San Juan Software website at www.sanjuansw.com

Or contact San Juan Software
 sales@sanjuansw.com
 (360) 378-8669

San Juan Software
 221 Wood Duck Lane
 Friday Harbor, WA 98250
 USA