SDD Evaluations - Object-Oriented Programming

Software Design Document Evaluation for the Team Consisting of

Ranjan Prasad
Bill Kilgallon
Weiqing Huang
Tom Clary
Stephen Andrew

Rating: SA

Preliminary Design

Good picture in the overview. Shows simply what is happening.

CSCI Overview - CSCI Architecture

Use of terminology - good - CC as an agent for the Command Interpreter and a container class. The Command Interpreter should have been listed first since the CC refers to it and it refers to no other object. A "defaults object" does not make a lot of sense as such - how about a defaults file manipulator? Interesting separation of Console for input and Display for output - fine. OID looks quite nice - no problem except that there are no non-procedural interfaces, which seems odd, but you *are* working in C++.

First significant error - you are not consistent between the names in 3.1.1 and the names in 3.1.1.1 - Display Interface on one and Display Handler on the other, File Formatter on one and Formatter on the other, no object on one and Database Handler on the other, and Defaults Object on one and no corresponding object on the other. Lots of detail on data flows and invocation between the objects - good.

Second significant error - your dependency diagram shows dependencies where there are no relationships in the OID. Command Interpreter depends on the Command Coordinator, Console Handler, and Display Handler only according to the OID. Dependencies in a dependency diagram are for direct dependencies - not indirect. The defaults object suddenly appeared.

CSCI Overview - System States

State table looks fine. Two states in the STD (Movement Processing and Search String Processing) are missing from the table. Names between the table and STD are not totally consistent (Initialize vs Initialization, for example). "Invalid Movement Attempted" is not an event related to the Command Interpreter according to the STD. State/associated CSC table is OK except that same problem with name consistency exists (File Formatter as a state vs File Formatting, for example). You suddenly used the acronym HCC without defining it, but you had it in the table of acronyms at the end, so OK.

CSCI Overview - Memory and Processing Time Allocation

Your guesses in 3.1.3.1 are fine, but the names do not always track with the definitions in 3.1.1 (Display Handler vs Display Interface, for example). You evidently omitted the Database Handler from 3.1.1.

The single biggest assistance you could have done for yourselves is to establish a data dictionary up front and used it.

CSCI Design

You covered all the CSCs mentioned previously - good. Requirements mapping the preliminary designs are OK overall. You used the conventions I showed you in the example - good. Command Coordinator is not complete in its OID symbol - you really need to meticulously spell out each method.

Detailed Design

Completely omitted - not a good sign.

CSCI Data

Omitted - not good. Surely you have plans for exported data?

CSCI Data Files

Fine - shows some research into the problem - good.

Requirements Traceability

Holes in this table are not good signs - some requirements are not being addressed. If they were not to be addressed (i.e., it did not make sense to), so state.

SDD Evaluations - Object-Oriented Programming

Software Design Document Evaluation for the Team Consisting of

Dave Curl
Joe Hoover
Jody Combs
Ken Richards

Rating: A

Preliminary Design

Picture - good.

CSCI Overview - CSCI Architecture

Very good, brief summary of the overall architecture. The selection of your object names looks fine. Identification of CS Parts is fine also. Your OID looks fine - what is the meaning of the dashed lines around Message_Handler? I note that Index_Handler is not included in the OID, and this is fine, especially if the CS_Browser has no interaction with the Index_Handler. From your write-up under 3.1.1, however, it looks like the Index_Handler belongs there. Dependency diagram is also fine. I note your name consistency - very good. There is no data flow to or from the Command method of the Shell_Handler - there should be. Also, it looks like there should be data flows to at least the Display and Print methods of the Buffer_Handler.

CSCI Overview - System States

Your state table looks fine. The names do not always track with the names in the STD (Shelling in the table vs Shell in the STD, for example). Most do, tho, so very good. Your STD is not nearly complete - no indication of the events causing the transition. In a Mealy STD you show events/resulting actions and in a Moore STD you show events - neither was shown here. Your state/associated CSC table is fine - names look consistent between the two tables under 3.1.2.1 and 3.1.2.3.

CSCI Overview - Memory and Processing Time Allocation

Your table looks fine.

CSCI Design

Good - your names track and the preliminary information seems complete. I compared the object names with 3.1.1. The method names did not always track, such as with the Buffer_Handler in 3.2.4 and the OID in 3.1.1.1. View/Change in the picture on 3.2.2 should be View_Change to match the OID et al.

Detailed Design

Very good - using PDL is fine for functional description. Not complete, but close - you left out the Index_Handler. Lots of thought went into this - good. PDL is relatively readable - good. Quite good overall.

No specifications are given - oversight. In detailed design, compilable specifications are an excellent product to give the implementers.

CSCI Data

While what you say is true, the details of the data structures and the like are not presented in the detailed design. It is really important for the data structures to be planned during design.

CSCI Data Files

You did not do an analysis of the structure and format of the data files for CS Parts. It would be really useful to have done this. Clearly an omission.

Requirements Traceability

Looks fine, but there is a gap for 3.8 in the SRS. If there is no requirement for implementation, you should so state in this table.

Software Design Document Evaluation for the Team Consisting of

Phil Barona
Dallas Marks
Carol Rollins
Colin Vogt

Rating: A

Preliminary Design

Picture - good.

CSCI Overview - CSCI Architecture

Names of top-level CSCs - good. Reusable components identified - good. Names are consistent between OID and 3.1.1 - good. Command_Dispatcher in figure 3.1.2 is shown as passive and in figure 3.1.3 is shown as active. Name consistency is maintained - good. Fonts are a little inconsistent in figure 3.1.3 (see File_Handler and Display).

CSCI Overview - System States

There is no state table - omission. STD looks fine.

CSCI Overview - Memory and Processing Time Allocation

No information - omission.

CSCI Design

Very good - names are consistent and OID symbols are in place.

Detailed Design

You could have used a flow chart or PDL in 4.1. Would have been better than just an English description.

Good - names are consistent.

COMMAND_TOKEN on page 19 is not shown in OID on page 9 - omission.

Your use of English as Ada comments is good in one sense (that there is a description of the functionality) but not in another (the description is not very detailed, leaving a lot to the implementer).

Syntax error on type BROWSER_TEXT_FILE (page 26) - you did not compile this package before putting it in the document - should have - catches lots of errors that way.

Your methods for Command_Dispatcher match those in its OID picture - good.

Your methods for File_Handler (pages 25-26) do not totally match the OID picture (page 11) - not good. Description is still a little high-level.

Syntax error in Index package spec - but methods match. You should have declared a new type for your STRING(1..64) objects. Your descriptions of the details are greatly improved for the Index package. Good overall.

Syntax error in package DISPLAY spec on page 32 (Screen_Height). No further details - possible omission.

Inconsistency between OID on page 15 (Get_Line) and spec on page 34 (Get_String). Words are enough in this case, so no design omission for using only the comments.

Type FILE for package Print_Handler is not defined - omission. Words are enough for design again.

CSCI Data

Incorrect statement - the exported data types et al from various packages are not documented here and sometimes not well in the packages.

CSCI Data Files

No documentation for CS Parts files - omission.

Requirements Traceability

Complete - good.

SDD Evaluations - Object-Oriented Programming

Software Design Document Evaluation for the Team Consisting of

Hitesh Jasani
Xianghong Liu
Satish Venkatesan

Rating: A

Preliminary Design

Picture - good.

CSCI Overview - CSCI Architecture

Listing of main CSCs with picture - good. Names consistent on page 6 - good. Definition missing from picture on page 6 - OK.

Names consistent between OID on 3.1.1.1 with listing on 3.1.1. Data flow looks reasonable in OID - OK.

Dependency diagram OK - names consistent.

CSCI Overview - System States

Listing of system states looks OK.

Quit and Unexpected Error Handler states in STD not shown in state table - omission. Editor and Shell not shown in STD. Names not consistent between table in 3.1.2.1 and STD (Editor Processing vs Editor Command Processing, for example). Not all transition events for Command Dispatch shown in state table. STD is not complete in events to transition back. Several omissions overall.

States/CSCs OK.

CSCI Overview - Memory and Processing Time Allocation

Looks fine.

CSCI Design

Additional CSCs shown that are not discussed in 3.1.1 - Error, Help, CS_Parts_File, Command_Handler (missing), Browser_h (missing). Inconsistent fonts (see title Display_Driver on page 22).

Mapping to requirements - good. Preliminary design - good overall.

Detailed Design

For 4.2, what do dispatch_command() and get_command() do? How are they to be implemented? Compilable class spec - good. Won't compile - missing ; at end - in general, you should compile these before putting them into an SDD - catches a lot of errors that way.

For 4.3, good overall. Class spec does not match OID diagram on page 17 (some methods are by different names, etc.). Writeups are adequate.

For 4.4, missing ; again. Methods OK. Writeup OK.

For 4.5, class spec and OID on page 19 do not match. Writeup lacks detail but OK.

For 4.6, class spec and OID on page 20 do not match. Writeup lacks detail - omission.

Display_Driver class is missing in detailed design.

For 4.7, class spec and OID on page 23 do not match. Writeup lacks detail - OK.

For 4.8, class spec and OID on page 24 do not match. No description at all - significant omission.

For 4.9, methods match! Good! But page 25 shows data exported, and class spec shows data hidden - error. Writeup adequate.

CSCI Data

OK. I did not note any global data associated with the classes - OK.

CSCI Data Files

Very good - you studied the CS Parts file structure.

Requirements Traceability

OK - looks complete.