RTD-Log
FTR-meeting 09.03.01 13:15 - 15:00 in C477
Minutes

Present: Matti Johansson, Marko Moisio, Juha Saarelainen,
Minna Vasankari, Dan Patrick Doepel,
instructor Mika Karlstedt,
projectmaster Raine Kauppinen,
customer Tiina Nikander
Absent: –

The purpose of the meeting was to review the design document. A list of the remarks is presented in the following.

- General:
  - The order of the module sections in Module plan should be changed.
  - It is not clear which module takes the responsibility of turning the log flow from TU to RU. Or how the turning is done.
  - The same subtitles should be found in every module section, even if the subsection contains nothing more than a "this subsection is empty" -remark.
  - The "Description in pseudocode" subsection of the module sections could be renamed to plain "Description" so that it allows broader explanations.
  - The information in section 6 is repeated earlier in the document. One solution might be to transfer the section somewhere earlier and then use references to it.
  - Explaining and giving reasons for things is desirable. Even the issues handled already in the requirements specification document can be repeated here to some extent.
- Introduction:
  - 1.1:

    * 1. sentence of the second paragraph: *seven* *important* sections
  - 1.2:
    * The table doesn't need a subsection of its own. It should also be named so that it can be referenced later.
    * Log system: The * system this... ("whole" removed)
    * Log manager: The *Log* system excluding SSS
    * The explanations for the terms DB and DBMS are missing

- Architecture:
  - 2.1:
    * Last word of the first paragraph: unit-¿Unit
    * The arrows should be referenced nicely in ascending order.
    * The sentence referencing to arrow 5 in the last paragraph should be rewritten. It is now unclear.
    * The disk is missing from figure 1 and that makes it difficult to interpret the picture.
  - 2.2:
    * Why does the system not run on PU? (last sentence of the first paragraph in 2.2.1)
    * "Log system" should be replaced with "Log manager" in the title of figure 3 and the reference to it in 2.2.2.

- Data plan:
  - 3.1:
    * Active and passive files should be explained.
    * RU is missing from the state list in the first sentence of the last paragraph.
  - 3.2:
    * File manager should be explained. Some general information about important files could be given already in the Architecture section.

        * It would be more important to clarify the purpose and function of the index here than discuss its internal structure.
    - 3.3:
        * It would be more flexible if the Control module would pass the adjustable parameters to the modules.

- Interface plan:
    - 4.1:
        * There are actually two different designs for the interfaces in this document. The semaphore using should be included in the interface functions. The BUF_empty -function may then not be needed at all.
        * 4.1.2: first paragraph, last sentence: Log Mediator should be DBMS.
        * Only one of the DB_write_image and read_log -functions may be needed, but it must be explained why these operations need or need not to be separated. The design of this interface must be reconsidered.

- Module plan:
    - The "Purpose"-subsection could be removed and its contents could follow the module section title right away.
    - Also the Module plan -section could use an introduction paragraph like the other sections have.
    - Returned error codes should be listed about each module. Some errors can be handled without stopping the whole system and that should be discussed too. Overall there should be explained more what is done when an error occurs.
    - Which module updates for example the global state variable? This should be made clear about each of the global variables: who uses it and who updates it.
    - When should the index and the image be stored? And by which module?

- The pseudocode resembles too much C-code.
- Control Module:
  * The file status should be asked again after the log flow has been turned.
  * Also this module should return error codes.
- SSS:
  * Subsections 5.2.3 and 5.2.4 could be moved into the Data plan
  * It should be made clear which variables in the File Manager are really constantly updated. For example size_of_active_log_file should not be that kind of variable.
  * 5.2.4 last paragraph: TU becomes PU already after the log file is turned.
  * 5.2.4: it should be explained more clearly why this variable is needed.
  * Who writes the index_write_mark?
  * There is no pseudocode for the initialization of the SSS.
  * The index will not fit into one single message. It should be explained how this is handled.
- Log saving module:
  * 5.3.1 secomd paragraph: ...Log mediator *could* use a semaphore
  * Clearing must be done also in TU.
  * The pseudocode is far too much like C and contains errors:
    · the condition of the while-loop
    · type = delete
    · the condition of the if clause in the end of page 20
  * There are unnecessary '_'-marks.
  * There is no need to mention the names of the actually called functions.
  * Error handling should be more specific.
- Image saving module:

    * 5.4.3 first paragraph: there should be a reference to the interface function.
    * 5.4.5 second paragraph, 4. sentence should be written more clearly. The index's logic of picking the oids should be explained here.
    * Last sentence on page 23: how is the goal of continuing the image saving optionally in TU handled in the Control module?
    * pseudocode: ask_index_for_next_unwritten_oid but the index module only seems to give the next oid
    * It may be neater not to put an actual log record to the records of type IMAGE_FIRST_RECORD and IMAGE_LAST_RECORD.
    * 5.4.6: other modules *may* return error codes. What codes?
  - Index module:
    * read_next_index returns oids to the Recovery module and to the Image saving module but with different logic. Does this work?
    * Return values should be defined values, not plain numbers.
    * The second parameter of the realloc-call should be fixed in function IM_reallocate_index on page 30.
    * IM_read_index does not return 0. In the same function the nextindexid should be compared with 0, not NULL. Also the whole logic of searching from the index is a bit strange.
    * The pseudocode in function IM_read_nextindex is a little too high-level.
    * When the index has been walked through once, the reading function should return a special value. Or there should be a separate reset-function which starts the walkthrough and then the normal reading function can return nulls when there is nothing more to read.
  - Storage module:
    * Sending functions "block" only when they must wait for a response.
    * SM-file_status_retrieve gets a pointer to the file status as a parameter, but the file status was supposed to be global.

    - The file-id parameter of the SM_index_write function is needed although it always has the same value. But this should be explained. Now it looks as if this had a conflict with the SSS index_write code on page 17.
    - The SM_receive_log function doesn't block. Why? Are the log records returned in the same order as they are on the disk?
    - The Image saving module should use the image_write function, not the log_write function as it now does.
    - Variables do not need to be declared as in send_message. Why are the header and data sent separately? Some possible error situations for this function:
      - data_length is wrong
      - malloc may fail
    - Does receive_message function guarantee that each message is delivered to the right receiver? Each receiver could be given its own message buffer.
  - Recovery module:
    - The receive_logs function may be unnecessary, since it only calls another function. There is no need to explain the internal functions in detail, if they don't call other modules' functions.
    - SM_receive_log does not return OK but receive_logs tries to use that value.
    - It should be explained why get_log_by_index is needed.
    - The values of the Global variable in RM_recover should be defined, not plain numbers.
    - There is a copy-paste error on the second row on page 48.
    - The first paragraph of 5.7.4 is unclear.
  - Total size estimate:
    - This should be a table containing an entry for each module. Also the designers of each module could be mentioned here. The LOC value will probably increase when all the error handling is taken into account.
- Communication protocol:

  - Subsection 6.2 should also contain a figure.

- Testing plan:
  - For example validation testing is missing entirely.

- Implementation restrictions:
  - 8.2:
    * the database stub will be needed anyway in testing.
    * All the test stubs must be returned and documented as part of the final product. This will also increase the LOC value.
    * A testing motor could also be useful.

- The document was approved with changes, but the group must carefully review it after the changes are made.

- The final version of this document was supposed to be ready next week, but that deadline is transferred one week forward.

- Everyone should start correcting their own parts of the document, except Juha who will be absent next week.

The next meeting is on Tuesday 13.3. at 12:15 in C475.