# Evaluation

## Method of Evaluation

### What method did we choose?

We chose a variant of the BTH 4-hour Architecture Evaluation method, BTH method.

### Why did why choose this method?

We chose the BTH method because it was the only evaluation method we were familiar with and it was the only one that we found that did not require weeks of evaluation or huge teams of evaluaters.

### What is the BTH 4-hour Architecture method?

The BTH 4-hour Architecture method is a method developed at Blekinge Institute of Technology. The BTH method is very much like the Architecture Tradeoff Architecture Method and the Software Architecture Analysis Method with the main difference that the BTH method is much shorter. It revolves around creating different scenarios that the architecture should be able to handle, or explainedwhy it does not in fact need to handle the certain scenario. The main purpose of the scenarios is to define the limits and requirements of the evaluated system, and in some cases to develop starategies to counter eventual problems or drawbacks in the system.

When using the BTH method one should have an exterior part to ask the questions and help formulate scenarios. However since we were unable to procure another group for our evaluation we had to improvise and evaluate ourselves. Unfortunatly since one is sometimes blind to one’s own work we risked to miss several possible scenarios that would have aided us greatly, and we did not gainthe insights one may receive when collaborating with another group.

### Other methods we considered

We considered both the Architecture Tradeoff Analysis Method, ATAM, and Cost Benefit Analysis Method, CBAM, methods, and the reason as to why we considered these alternatives, and only these, was because they were the only other evaluation methods described in the main course literature. There were other methods described during the lectures but we felt we did not know enough about them to take them into consideratoion.

### Reasons as to why we did not choose the other methods

The reason why we did not choose the ATAM evaluation method was because it required resources we did not have, namely time and a larger team. With ATAM you practically start with an hiatus of two to three weeks after a day of evaluation, and after the hiatus it requires up to a week of continued work, and this was, and is, time that we do not have. And this alone made ATAM impossible for us to use, but beyond the time ruirement ATAM also requires a large evaluation team and prefferably stakeholders, persons we do not have available for obvious reasons.

The main reason as to why we did not choose to use CBAM was because it mainly concerns itself with budget and profits, the economical aspect of the whole project. And since we have been told not to concern ourselves with the organizational aspects of the project, and the fact that we do not have a budget nor a customer we felt that this method was highly irrelevant.

## Quality Attributes Tested

The three quality attributes that we found most important to evaluate were *System dependability*, *Input types* and *Maintainability*.

We felt that the system dependability was the most important attribute to be tested at this point, probably will continue to be, as our system’s purpose is to ensure the integrety and stability of other systems.

We chose evaluate input types since it is one of the major issues in the system description:

“*The system shall be able to simulate various inputs, such as, but not limited to, file input,*

*command line input, socket communication, and graphical input. It must be easy to incorporate new input mechanisms in the system.”*

With this in the description we felt that it was well worth more of our attention.

The maintainability attribute was chosen because we envision that the MIB will be used by ther companies for many yeasrs to come and thus it is important to consider the maintainabilityaspects of our system.

## Scenarios

### System dependability

#### The tested system uses as much as 10 gigabyte if primary memory

According to our research this is not a feasable quantity of required primary memory for any program running on a standard computer both presently and in the percievable future. This scenario did however make us add an assumption about the memory requirements of the tested system.

#### The tested system crashes during testing

The testing wrapper will acquire the last saved state of the tested system from the checkpoint module and then restore the tested system from there and continue the testing.

#### The tested system crashes during testing and keeps on crashing at the same moment every time

As we discussed this scenario it became clear that we needed a new issue card, that we named “Multiple Crash Issue”. The wrapper around the tested system will count the amount of crashes and map them to the chackpoints. If the tested system crashes a predetermined amount of times between two checkpoints the MIB will terminate the current test and note this in the test log.

This issue however had already been considered in the first assignment but not properly documented, as such the views have not required any polishing because of this “new” issue card.

### Input types

#### Input stream from a blu-ray disc

Our input system is able to handle the streaming from a blu-ray disc as the maximum transfer rate of a blu-ray disc, 54 MB/s, is well below the capacity of any standard computer.

### Maintainability

#### The customer returns after buying the system and requsts two new testing techniques

By maintaining good semantic coherence thorughout the wrapping module as well as annticipating any expected changes in testing techniques. Thus changes will be kept to a minimum and localized to the testing wrapper.

#### The costumer returns after buying the system requests two new input types

By generalizing the input modules we make it simple to implement new input ttypes and changes the impact on the rest on the systemm will be minimal.

#### The test is finished and the output needs to be validated

When the testing is completed the data from the test will be transferred from the data log to the output validation system. Within the output validation system the output will be controlled and validated against predefined output in the validation module. The report module wil then compile all the data together with the validated output into a final readable report.

## Evaluation Conclusion

We have found that, with our current knowledge of architecture structures and with the results of our self evaluation, our current structure upholds the quality requirements as set by the system description. As such we have found no doing an architecture transformation.