ERJU SYSTEM PILLAR

# Systems Engineering Management Plan - Annex T Trade-Off Analysis

# Systems Engineering Management Plan - Annex T Trade-Off Analysis

| | |
|---|---|
| Author(s) | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) |
| Abstract | Describe guidelines for tradoff analysis in System Pillar |
| Config Item | System Engineering Management Plan |
| Document ID | SEMP Annexes/Annex T Trade-off Analysis#723403 Systems Engineering Management Plan - Annex T Trade-Off Analysis |
| Classification | Public |
| Status | Released |
| Version | 1.0 |
| Revision | 723403 |
| Last Change Date | 02.10.2025 |
| Copyright | Brussels: Europe’s Rail Joint Undertaking, 2025 |



## Document History

| | | |
|---|---|---|
| 0.1 05.03.2024 | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) | Reviewed version (initial) (initial) |
| 0.2 18.09.2025 | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) | Reviewed version including Findings from Review 0.1 |
| 1.0 26.09.2025 | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) | Approved version based on Review 0.2 |

**Approval by reviewers** (captured at end of 'In Review by System Pillar')

| | |
|---|---|
| Type of Approval | Document Review |
| Comments | **#1** by **SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF)** on 2025-09-26 16:12<br>Content was not changed since the last version. Document is considered reviewed in SEMP V3 |
| Approvals | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) : Approved , Dennis Kunz : Waiting , ANTOONS Gilles : Waiting |

Approval by approvers (captured at end of 'In Approval by System Pillar')

| | |
|---|---|
| Type of Approval | Document Approval |
| Comments | **#1** by **SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF)** on 2025-09-26 16:16<br>Content was not changed since the last version SEMP V3. Document is considered approved in SEMP V4<br>**#2 Approval comment** by **Renard, Marie Pierre (SMO RI MT FR ADC TGMTR3)** on 2025-09-30 09:23<br>Open comments needs to be resolved during next contract |
| Approvals | SANGO Marc (SNCF / DIR TECHNOLOGIES INNOVATION ET PROJETS GROUPE / IR DIR RECHERCHE - PSF) : Approved , Jorge Block : Waiting , Renard, Marie Pierre (SMO RI MT FR ADC TGMTR3) : Approved , Schmidt Steffen (I-NAT-GST-ERTM) : Waiting , SCHWAN Nico : Approved , KEFALAS, Georgios : Waiting |

# Table of contents

# 1 Introduction

## 1.1 Introduction to tradeoff analysis

Tradeoff studies are broadly recognized as the method for simultaneously considering multiple alternatives with many criteria or attributes, and as such are recommended by several standard bodies including SPPR-7763 - ISO/IEC/IEEE 15288:2023 or SPPR-7764 - INCOSE SEBok Decision Management Process .

## 1.2 Context and purpose in this document

In the context of the definition of a system, the tradeoff analysis method consists of providing a guideline to evaluate architecture's fitness with respect to multiple attributes . The attributes interact, and improving one often comes at the price of worsening one or more of the others. The conflicts between attributes can be resolved by engineering design principles of best practices, they can constrain the space of design possibilities or generate alternatives and candidate architectures.

Note : The term architecture is used in the sense of ARCADIA from the need space (Operational Analysis and System Analysis) to solution space (Logical Architecture and Physical Archidecture).

The purpose of this architecture tradeoff analysis method is to assess assess alternatives that could be chosen for any given decision by taking account all the influencing quality attributesthat are to be considered in decision making.

Note : The user of this method must define the scope of the tradeoff analysis: 1) identify the attributes which can be high-level attributes, such as Common Business Objectives or low-level derived requirements such as SPPR-2536 - Architecture Design Quality Attributes or quality attributes defined or refined by System Pillar Tasks/Domains, and 2) identify alternatives (potential options or solutions for a given objective or requirement).

## 1.3 Terms and abbreviation

**Guideline to add or change definitions (abbreviations or terms) in SP documents**.
SPPR-6669 - General guidelines to establish a list of Glossary definitions in a System Pillar deliverable
**[**SPPR-7579 **]**

The reused or new definitions and abbreviation in this document are provided hereafter.

### 1.3.1 List of Definitions

| Term | Status | Definition |
|---|---|---|
| SPPR-7586 - Architecture Tradeoff Analysis Method | Open | Architecture Tradeoff Analysis Method<br><br>Referenced by: SPPR-7571, SPPR-7572 |
| SPPR-7766 - Attribute | Open | Attribute is an inherent property or characteristic of an entity that can be distinguished quantitatively or qualitatively by human or automated means<br><br>Referenced by: SPPR-7571 |
| SPPR-7768 - Tradeoff points | Open | |

| Term | Status | Definition |
|---|---|---|
| | | Tradeoff points are architectural elements that affect different quality attributes simultaneously or architectural element to which multiple quality attributes are sensitive.<br><br>Referenced by: SPPR-7571 |
| SPPR-7767 - Quality attribute | Open | Quality attribute is a realized non-functional requirement used to evaluate the performance of a system<br><br>Referenced by: SPPR-7571 |
| SPPR-7769 - Sensitivity points | Open | Sensitivity points are any modeled values under a given individual quality attribute that are significantly affected by a change to the architecture<br><br>Referenced by: SPPR-7571 |

### 1.3.2 List of abbreviations

**ATAM -** Architecture Tradeoff Analysis Method

## 1.4 References

**SPPR-7763 -** ISO/IEC/IEEE 15288:2023 Systems and software engineering — System life cycle processes

**SPPR-7764 -** INCOSE SEBok Decision Management Process

**SPPR-7765 -** ISO/IEC/IEEE 15939:2007. Systems and software engineering - Measurement process.

**SPPR-7744 -** Rick Kazman et al. The architecture Tradeoff Analysis Method. Technical Report CMU/SEI-98-TR-008, July 1998

**SPPR-7783 -** In, H., Boehm, B., Rodgers, T., and Deutsch, M. 2001. Applying WinWin to quality requirements: A case study,
In Proceedings of the 23rd International Conference on Software Engineering (ICSE), pp. 555–564.

**SPPR-7782 -** Kazman, R., Asundi, J., and Klein, M. 2001. Quantifying the costs and benefits of architectural decision, In Proceedings of the 23rd International Conference on Software Engineering (ICSE), pp. 297–306.

**SPPR-7785 -** Saaty, T.L. 1980. The Analytic Hierarchy Process, New York, McGraw Hill.

**SPPR-7838 -** Pugh, S. (1981a). Concept selection: A method that works. In Review of design methodology. Proceedings international conference on engineering design. Heurista, Zürich, 497–506

## 2 Architecture Tradeoff Analysis Method

The ATAM is a spiral model of design depicted in Figure 1. The ATAM is like the standard spiral model in that each iteration takes one to a more complete understanding of the system, reduces risk, and perturbs the design. The method is divided into four main areas: 1) scenario & requirements gathering, 2) architectural views &scenario realization, 3) model building & analyses and 4) tradeoffs.

Generally, the method works as follows: Once a system's initial set of scenarios and requirements has been elicited and an initial architecture or small set of architectures is proposed (subject to environment and other considerations), each quality attribute is evaluated in turn, and in isolation, with respect to any proposed design. After these evaluations comes a critique step. During this step, tradeoff points, elements that affect multiple attributes, are found. After the critique, we can either: refine the models and reevaluate; refine the architectures, change the models to reflect these refinements and reevaluate; or change some requirements.

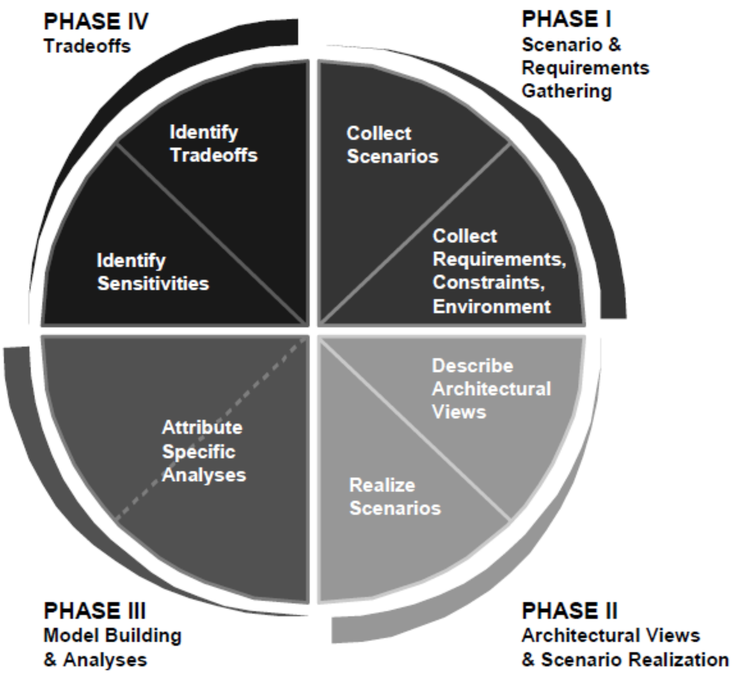



*Figure 1 Steps of the Architecture Tradeoff Analysis Method*

### 2.1 Steps of the method

This section looks at each of the ATAM steps (Figure 1) in more detail. It should be made clear that it is not expected these steps to be followed linearly. They can and do interact with each other: an analysis can lead to the reconsideration of a requirement; the building of a model can point out places where the architecture has not been adequately thought out or documented. This is why the steps are depicted as wedges in a circle: at the center of the circle every step touches (and exchanges information with) every other step.

#### 2.1.1 Step 1: Collect Scenarios

**Description**
The first and second steps in the ATAM method are interchangeable. Indeed, at times requirements exist before any architectural analysis has started. At other times scenarios drive the requirements. The analysis process can profitably begin with either. **[**SPPR-7770 **]**

**Guidelines**
Eliciting scenarios serves the same purposes as it does in system needs understanding processes

(*SPPROCESS/SEMP Annex D Processes/SEMP process group 02-Operational Analysis : 723403* and *SPPROCESS/SEMP Annex D Processes/SEMP process 03-System Analysis : 723403* ) to define both functional and quality requirements, to facilitate communication between stakeholders, and to develop a common vision of the important activities the system should support. **[**SPPR-7771 **]**

### 2.1.2 Step 2: Collect Requirements/Constraints/Environment

**Description**

In addition to the scenarios, the attribute-based requirements, constraints, and environment of the system could be identified according to the scope of tradeoff analysis. A requirement can have a specific value or can be more generic, as derived from scenarios of hypothetical situations. The environment must be characterized because subsequent analyses (e.g., performance or security) and constraints on the design space, as they evolve, are recorded as these too affect attribute analyses. This step places a strong emphasis on revisiting the scenarios from the previous step to ensure that they account for important quality attributes. **[**SPPR-7772 **]**

**Guidelines**

Requirements or constraints can be captured directly in a requirements management tool (Polarion by following these guidelines : *SPPROCESS/SEMP Annex R Requirements/SEMP Annex R2 - Requirements patterns syntax : 723403* and *SPPROCESS/SEMP Annex R Requirements/SEMP Annex R3 - Rules for writing textual requirements : 723403* ) or in a system analysis model (Arcadia Capella by using Arcadia/ Capella constraints or requirement add-on : SPPR-5197 - Define system constraints ) **[**SPPR-7773 **]**

### 2.1.3 Step 3: Describe Architectural Views

**Description**

The requirements, scenarios, and engineering design principles together generate candidate architectures and constrain the space of design possibilities. In addition, design almost never starts from a clean slate: legacy systems, interoperability, and the successes/failures of previous projects all constrain the space of architectures. **[**SPPR-7774 **]**

**Guidelines**

The details required for an analysis of each of the qualities are typically captured in distinct architectural views: for example a dataflow view (for reasoning about functional requirements), a process view (for reasoning about performance), a class view (for reasoning about sharing of object definitions), a component view (for reasoning about interaction of components and information hiding), etc. The result of this step leads to the description of candidate alternatives:

| | **Alternative 1** | **Alternative 2** | **Alternative 3** | **Alternative 4** | **...** |
|---|---|---|---|---|---|
| Description | | | | | |

**[**SPPR-7775 **]**

### 2.1.4 Step 4: Attribute-Specific Analyses

**Description**

Once a system's initial set of requirements and scenarios has been elicited and an initial architecture (or small set of architectures) is proposed, each quality attribute must be analyzed in isolation, with respect to each architecture. These analyses can be conducted in any order; no individual critique of attributes against requirements or interaction between attributes is done at this point. Allowing separate (concurrent) analysis is an important separation of concerns that allows individual attribute experts to bring their expertise to bare on the system.

**[**SPPR-7776 **]**

**Guidelines**

To do the attribute-specific analysis, the attribute-specific scenarios collected in Section SPPR-7747 - Step 1: Collect Scenarios are mapped onto the architectures described in SPPR-7749 - Step 3: Describe Architectural Views and then calculate whether this realization of the scenario meets the requirements set forth in SPPR-7773 - Guidelines Requirements or constraints can be captured directly in a requirement.

The ATAM allows the designer to focus on those attributes that are considered to be primary, and then introduce others later on. Though all analyses need not occur “up front and simultaneously,” the analyses for the secondary attributes can still occur well before implementation begins.

The result of the analyses leads to performance analysis about system behavior with respect to values of particular attributes and the critique of the analysis:

| | **Alternative 1** | **Alternative 2** | **Alternative 3** | **Alternative 4** | **...** |
|---|---|---|---|---|---|
| Description | | | | | |
| Performance analysis | e.g., Performance : “requests are responded to in XXms. average”;<br>e.g.: HW Cost : “the hardware will cost ZZ EUROS per platform”; | e.g., Safety/avaliability : “the mean time to failure is YY days”; | e.g., Security “the system is resistant to known attack scripts”;<br>SW Cost : “the software will require AA people per year to maintain”; | and so forth | ... |
| Critique of the analysis | .g., Alternative 1 has poor performance and availability. It is also the least expensive option (in terms of hardware costs; | e.g., alternative 2 has excellent availability, but at the cost of extra hardware. | and so forth | and so forth | ... |

*Table 1 Attribute-specific analysis*

**[**SPPR-7777 **]**

### 2.1.5 Step 5: Identify Sensitivities

**Description**
This step determines the sensitivity of individual attribute analyses to particular architectural elements; that is, one or more attributes of the architecture are varied. The models are then varied to capture these design changes, and the results are evaluated. Any modeled values that are significantly affected by a change to the architecture are considered to be sensitivity points. **[**SPPR-7788 **]**

**Guidelines**
Table 2 provides an example of application of change impact on individual attribute. The first columns represents
the individual quality attributes, whereas the last columns shows a type of change. Impact values are in the range of -2 to +2 for the proposed changes. The values are obtained by a team of domain experts. A change can have positive, negative or no impact on the individual attributes or multiple quatities.

- If a change does not have too much impact on the qualities the architecture has robust to change.
- If multiple attributes are sensitive to change qualities further investigation could be conducted to explore other means to restructure exising architecture.

| | **Alternative 1** | **Alternative 2** | **Alternative 3** | **...** |
|---|---|---|---|---|
| | Change of types of communication mechanisms in the scenarios | Change communication of control links in the scenarios | Change of computational units in the scenarios | Change of active data repo |

| | Alternative 1 | Alternative 2 | Alternative 3 | ... |
|---|---|---|---|---|
| | | | | sitori es |
| Quality attribute 1: Reliability | +1 | +1 | 0 | |
| Quality attribute 2: Availability | 0 | +2 | +2 | |
| Quality attribute 3: Maintainability | -1 | +1 | -1 | |
| Quality attribute 4: Safety | 0 | -1 | +1 | |
| Quality attribute 5: Security | | | | |
| Quality attribute 6: Interoperability | | | | |
| Quality attribute 7: Modularity | | | | |
| Quality attribute 8: Cost | | | | |
| Quality attribute 9: Performance (latency and throughput) | | | | |
| Quality attribute 10: Ecological (Green power usag, Energy usage optimization, ...) | | | | |
| Net Score of impact | 0 | +3 | +2 | |
| Critique of analysis | Robust to change | Positively impact on RAM and negatively impact on Safety => further analyzes are necessary | Positively impact on availability and safety; and negatively impact on maintenanbility => further analyzes are necessary | |

*Table 2 Sensitivity analysis*

More advanced sensitivity analysis tools are available on the market, such as Sensitivity Analyzer to evaluate how the parameters and states of a Simulink® model influence the model output or model design requirements.

**[**SPPR-7789 **]**

### 2.1.6 Step 6: Identify Tradeoffs

**Description**
The next step of the method is to criticise the models built in the previous step and to find the architectural tradeoff points. Although it is standard practice to critique designs, significant additional leverage can be gained by focusing this critique on the interaction of attribute-specific analyses, particularly the location of tradeoff points. A tradeoff point could lead to a tradeoff decisions. When this occurs, ATAM largely leaves the decision process to requirement negotiation. **[**SPPR-7784 **]**

**Guidelines for identification of tradeoff points**
Once the architectural sensitivity points have been determined, finding tradeoff points is simply the identification of architectural elements to which multiple attributes are sensitive.
**[**SPPR-7787 **]**

**Guidelines for tradeoff decision**
Stakeholders can use the WinWin model to assist their negotiation in resolving conflicts following a rigorous process WinWin Method . These methods have proved useful when consensus can be reached among stakeholders.

When relationships among design alternatives and quality attributes are too intricate to solve by informal negotiation, more formal quantitative methods need to be used. Two types of method exist in the architecture evaluation domain.

1. One is to use a method like Cost Benefit Analysis Method (CBAM Method) in which stakeholders can link all potential design decisions to their benefits through a response-utility function and then perform a value analysis to determine the best candidates. In the real world, the utility function of the responses can be difficult to solicit from stakeholders.
2. The other type of method is to use MCDM (Multi-Criteria Decision-Making) techniques, such as AHP Method, Pugh Matrix :
3. To help discriminate between the options the criteria can be weighted (optional). Typically the weighting is on a 1 to 5 scale with 1 the lowest and 5 the highest weighting.
4. Compare each candidate alternative against the reference architecture, quality attribute by quality attribute and decide a pair-wise score with:
5. "==" meaning "Same"
6. "+" meaning "Better"
7. "++" meaing "Much Better"
8. "-" meaning "worse"
9. "--" meaning "Much Worse"

| Quality attributes | Criteria Weights (optional) | Reference Alternative | Alternative 1 | Alternative 2 | Alternative 3 | ... |
|---|---|---|---|---|---|---|
| Example | 1 | Train Positioning | Trackside-centric alternative : track circuits | Trackside-centric alternative: axle counters | Onboard-centric alternative: track occupancy independently from trackside devices | |
| Quality attribute 1: Reliability | .. | == | ++ | + | + | |

| Quality attributes | Criteria Weights (optional) | Reference Alternative | Alternative 1 | Alternative 2 | Alternative 3 | ... |
|---|---|---|---|---|---|---|
| Quality attribute 2: Availability | .. | == | == | + | ++ | |
| Quality attribute 3: Maintainability | .. | == | - | - | ++ | |
| Quality attribute 4: Safety | 5 | == | + | + | - | |
| Quality attribute 5: Security | .. | == | = | = | -- | |
| Quality attribute 6: Interoperability | .. | == | - | - | + | |
| Quality attribute 7: Modularity | | == | | | | |
| Quality attribute 8: Cost | | == | | | | |
| Quality attribute 9: Performance (latency and throughput) | | == | | | | |
| Quality attribute 10: Ecological (Green power usag, Energy usage optimization, ...) | | == | | | | |
| SUM | | | | | | |

*Table 3 Sample Pugh Matrix*

**[**SPPR-7786 **]**

## 2.2 Iterations of ATAM

After the above steps have been completed, we can compare the results of the analyses to the requirements.

1. When the analyses show that the system's predicted behavior comes adequately close to its requirements, the designers can proceed to a more detailed level of design or to implementation. In practice, however, it is useful to continue to track the architecture with analytic models to support development, deployment, and maintenance.
2. In the event that the analysis reveals a problem, we now develop an action plan for changing the architecture, the models, or the requirements. The action plan will draw on the attribute-specific analyses and identification of tradeoff points. This then leads to another iteration of the method.

## 3 Process for tradeoff analysis

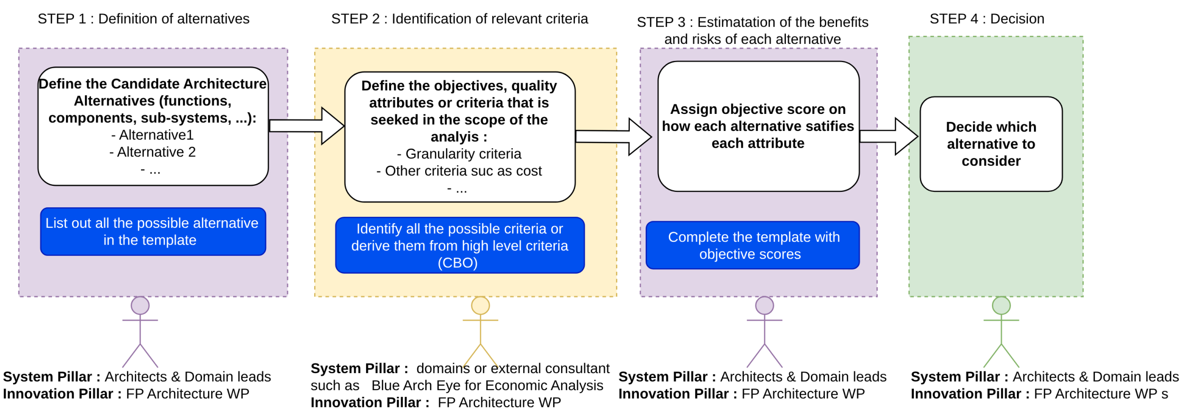
Tradeoff Analysis Process

The process for tradeoff analysis is presented hereafter

1. List out all the possible alternatives that could be chosen for any given decision
2. Identify and list out all the influencing objectives, attributes or criterias that are to be considered in making the decision
3. Provide a weighting to each attribute, giving it an importance score. This step is optional but if it is used it is suggested that the weighting of all infuencing attributes is completed before any scoring takes place.
4. Assign objective scores on how each alternative satisfies the attribute (best to do this on an attribute-by-attributes basis to estimate the benefits and risks of each alternative)

1. Decide which alternative to consider based on the highest resulting score, after all influencing attributes have been considered for each alternative.

Worflow to be applied for alternative assessment




**[**SPPR-8706 **]**

## 4 Template for tradeoff analyis record

The user of the tradeoff analysis process must define the scope of the tradeoff analysis:

1. Identify the attributes which can be high-level attributes, such as Common Business Objectives or low-level derived requirements such as SPPR-2536 - Architecture Design Quality Attributes or quality attributes defined or refined by System Pillar Tasks/Domains, such as those of granularity principle document .
2. Identify alternatives (potential solutions for a given objective or requirement)
3. Compare each candidate alternative against the other, attribute by attribute and decide a pair-wise score with:
4. "==" meaning "Same"
5. "+" meaning "Better"
6. "++" meaing "Much Better"
7. "-" meaning "worse"
8. "--" meaning "Much Worse

The record of tradeoff analysis can be summarized in a matrix by using the following steps:

| | **Attributes Weights (optional)** | **Alternative 1** | **Alternative 2** | **Alternative 3** | **Alternative 4** | **...** |
|---|---|---|---|---|---|---|
| Attribute 1 | | | | | | |
| | | | | | | |

| | **Attributes Weights (optional)** | **Alternative 1** | **Alternative 2** | **Alternative 3** | **Alternative 4** | **...** |
|---|---|---|---|---|---|---|
| Attribute 2 | | | | | | |
| Attribute 3 | | | | | | |
| Attribute 4 | | | | | | |
| ... | | | | | | |
| Resulting score | | | | | | |

*Table 4 Tradeoff analysis record*

**[**SPPR-8713 **]**

## 5 Status of the work, open points, issues

A decision-making process is not an accurate science; ergo, tradeoff studies have limits. The following concerns should be taken into account during the analysis.

- Subjective Criteria: personal bias of the analyst
- Uncertain Data: how can a systems engineer predict the evolution of a criteria over the time?