Performance Modeling of a Surveillance Mission

Xavier Quayzin, APSYS

Emmanuel Arbaretier, APSYS

Key Words: Altarica, Reliability Modeling, R&M Applications in Aerospace

SUMMARY & CONCLUSIONS

When you have to select the number of aircraft to realize a mission, not being confident in your performance could be expensive and hazardous. By using performance modeling and Monte Carlo simulation, you can define the performance's distribution for the option you may choose or find the optimal solution. This paper details the way to obtain such a result and the added value the method has from an operational and financial point of view compared to a traditional method based on spreadsheet calculations.

A key conclusion of this analysis is that the performance level of the mission depends on the number of operators available to use the system. This is important in the sense that the customer often provides the operators, and their number is fixed. Therefore your maximum performance is defined by this number and not by the number of systems you can exploit. Therefore, even if your system is highly reliable and resilient, it will be impossible to achieve a high level of availability without the right concept of operations and concept of maintenance in place. Furthermore these aspects must be thought about from the first stages of the project in a systems approach as they will determine your performance and your financial performance for a Service Level Agreement (SLA) based contract.

This study also demonstrates that the current performance modeling software and languages are sufficient to obtain the statistical distribution of the performance of the system from the initial concept and that could allow companies to optimize their offer and to be aware of the risk that they take. Moreover, the use of tools based on spreadsheet can't be used for this kind of simulation as they don't allow you to model share resources, time dependent phenomenon, conditional logic.

1 INTRODUCTION

In the defense and security domains, more and more contracts include a Service Level Agreement (SLA) on the performance of the mission for example in the case of a surveillance mission: the ratio effective hours of surveillance to total mission's hours. These times are quite easy to compute during the in-service phase. But it is during the design and often the bidding phase that the decisions, which are going to impact on the mission's performance, are made. Therefore, it is crucial to find the best way to consider the issue of performance modeling from the initial concept phase.

To investigate this assertion, a 24h a day, 7 days a week surveillance mission carried out by helicopters is chosen as an example to guide our analysis. To model the performance of the mission, reliability and maintainability data of the helicopters are taken into account: MTBF, MTTR, preventive maintenance, corrective maintenance and associated costs. The core of this paper will describe the different assumptions made and their justification, how the models have been built in excel and in a simulation tool based on the ALTARICA language: SIMFIA. After designing the model it is possible to run a Monte Carlo simulation and observe the statistical distribution of the performance of the mission. In that case, the performance is expressed as the ratio of number of surveillance's hours to number of hours in the year and the number of mission interruption for more than 2 hours. These two indicators are the two conditions of the SLA. Thus this result is translated in costs according to the SLA to which the acquisition cost, exploitation cost, preventive maintenance cost, and corrective maintenance cost are added to obtain the through life cycle cost of the mission.

A similar approach is taken using a spreadsheet to compare both results.

2 THE ALTARICA LANGUAGE

First of all, to have a more precise idea of what is modeled in this sample cases, let's introduce the Altarica language, [1] [2].

Altarica language is very suitable for the description of a wide number of systems; it is mainly devoted to reliability and dependability studies.

Altarica is dedicated to the study of critical systems from both a functional and dysfunctional point of view. The functional viewpoint includes all of the techniques that aim to reinforce the conviction that the system meets its specification. The dysfunctional viewpoint includes those that aim to analyze what happens when something goes wrong, i.e. how and with which probability the system may fail.

It generalizes some of the most widely used description formalisms such as finite state automata, Petri nets or block diagrams. It is characterized by:

- Hierarchical description: the system is made of several sub systems representing the other complex or elementary natural components.
- Mode automata.

1-4244-2509-9/09/$20.00 ©2009 IEEE

- In / out flow.
- Discrete events.
- Transitions: Each component possesses different states (working, failure, standby…). Occurrence of events (failure, repair, start back up…) modifies this state then transition is fired. The firing time of this transition can be deterministic or random with a given distribution.
- Ability to model share resources such as spares, maintenance team and equipment [3].

In our case, we used the software tool: SIMFIA to translate automatically the graphical representation of the system in Altarica language. In the SIMFIA interface, we can run a stochastic simulation, in that case a Monte Carlo simulation, to assess the different performances of the mission [4] [5].

3 DESCRIPTION OF THE MISSION

All the following figures are fictional and don't represent any existing aircraft and mission.

3.1 Introductions

The mission models in that case study is a surveillance mission carried out with helicopters.

The following operational parameters are defined:

- Surveillance 24h / 24h.
- 6 helicopters.
- Maximum of 3 flying crews available at any time.
- No limit on the number of flight per aircraft per day.
- Duration of the operation: 1 year.

3.2 Mission profile

The maximum number of flying crew available to flight at the same time means that a maximum of 3 aircrafts can be in flight at the same time, which can be expressed in normal conditions by in 1 helicopter in surveillance, 1 in Flight to, 1 in Flight back.

The duration of the different phases of the mission are:

- Pre-flight check : 0.17h.
- Flight to surveillance position : 1h.
- Surveillance : 2h.
- Flight back to platform : 1h.
- Maintenance if required.
- Refuel and mission preparation: 1.5h.
- Wait.

3.3 Helicopter performances

Two types of failure are taken in account in the model: critical failure and degraded failure.

Critical failures are failures that end the flight and for which the helicopter needs to go to maintenance immediately. The failure rate associated with this kind of failure is 1E-6 per flight hour. The Mean Time To Repair (MTTR) for these failures is 4 hours.

Degraded failures are failures that allow to finish the mission but need to be repaired at the end of the mission before the next flight. The failure rate associated with this kind of failure is 1E-3 per flight hour. The Mean Time To Repair (MTTR) for these failures is 0.5 hours.

In terms of preventive maintenance, in addition to the pre-flight check, we take into account in the model a check every 15 flight hours. This check has a duration of 0.33h.

3.4 Costs

The following costs are defined to build the financial model of the mission's variable cost per year:

- Preventive maintenance: $50 per check.
- Corrective maintenance, critical failure reparation: $2500.
- Corrective maintenance, Degraded failure reparation: $100.

3.5 Service Level Agreement (SLA)

The following two conditions defined the SLA of the surveillance mission:

- Condition 1: Availability target (surveillance time / mission time): $500,000 per % below 95%.
- Condition 2: mission interruption of more than 2h: $100,000.

4 MODELING OF THE MISSION

4.1 Modeling with SIMFIA / Altarica Language

To model this case, we used the software tool SIMFIA. SIMFIA allows to generate the equivalent Altarica code to the modeled system. It will be used to make stochastic simulations (Monte Carlo simulation) and to generate sequences. Altarica language is based on the formalism of the automats with discrete states. It is a hierarchical language, where almost all syntactic constructions have a graphical representation

A set of experiments showed that the Altarica language is very suitable for the description of a great number of systems. Although Altarica is mainly devoted to the reliability and safety studies, it is a language of object description. It generalizes the most used formalisms of description such as Petri or queuing networks, state charts or blocks diagrams.

The first step of the modeling is to create a representation of the different components of the mission (see figure 1). In addition to the 6 helicopters, two "virtual" blocks are created. On the left, the block "Global mission state" is a monitoring block, which is there to register the state of the mission at any time in order to calculate the penalties. This block is divided in two functions. The first one computes the availability of the mission, the second one monitors the interruption time and increments the penalty cost associated to the second condition of SLA. On the right, the block "Request" reproduces the logic to select and request a helicopter to assure the best availability. It basically follows the state of each helicopter and tries to optimize the availability of the mission within the constraint of no more than 3 flying crews.

The helicopter can be divided in three different sub-blocks which represent the helicopter itself, block "H1", the mission of the helicopter, block "Mission_H1", and the maintenance cost of the helicopter, block "H1_cost".

The block "H1" represents the helicopter. The two main roles of this block is to simulate the state of the helicopter and to calculate the cost of the maintenance corrective. Figure 2 (see next page) represents the automata associated to the helicopter. The helicopter starts in the state "Etat=OK" and stays in this state as long as a no failure occurs. In terms of failure, two types are modeled: critical "Etat = KO" or degraded "etat = deg". occurs when a critical failure occurs. If "Etat = KO", then the helicopter waits for reparation. It goes to the state start reparation (start_rep_c) as soon as the maintenance spares and facilities are available. Then the reparation stats and it is modeled with a dirac of 2h (MTTR). At the end of the maintenance operation, the state of the helicopter is "OK" and the cost of the maintenance is added to the exploitation cost variable. If a degraded failure occurs, the reparation scenario is the same, except for the duration and cost. The outputs of this block are a variable for the cost of corrective maintenance which is an input for the block "H1_cost" and a variable with the three different states of the helicopter (OK, deg., KO) which is an input of the block "Mission_H1".

The block "Mission_H1" models the 6 different states of mission the helicopter can follow as shown by figure 3. All different paths are modeled and the associated transition described in terms of impact on the surveillance, time, availability of the helicopter and cost. The outputs of the block "Mission_H1" are:

- A variable for the cost of preventive maintenance which is an input for the block "H1_cost"
- A number of variables which indicate in which phase of the mission the helicopter is. These variables are input for the blocks "H1", "Request", and "Global mission state".

The block "H1_cost" follows the cost of the preventive and corrective maintenance. Each time a maintenance operation is carried out a cost variable is incremented. The role of "H1_cost" is to add the corrective maintenance variable calculated by "H1" and the preventive maintenance calculated by "mission_H1" to compute an aggregate variable corresponding of the maintenance cost of the helicopter 1.

Table 1 presents the results of the Monte Carlo simulation of this model.

Further simulation with various number of helicopters shows us that increasing the number of helicopters doesn't improve the performance and that the limiting factor is the number of flying crews. For example with 3 flying crews simultaneously you can't have more than 1 helicopter in surveillance. If instead of 3 flying crews, you have 4 flying crews you can have two helicopters in surveillance at the same time and by doing so you increase your availability substantially. Figure 5, below, demonstrated that if you can guarantee to have 6 aircrafts available at any time, you need 4 flying crews to achieve a high level of mission availability. However, the strategy with 4 flying crews implies more flight hours and therefore more maintenance and more costs. A further case study will be to run this model with 4 flying crews and see if the additional costs off set the reduction in penalties cost.

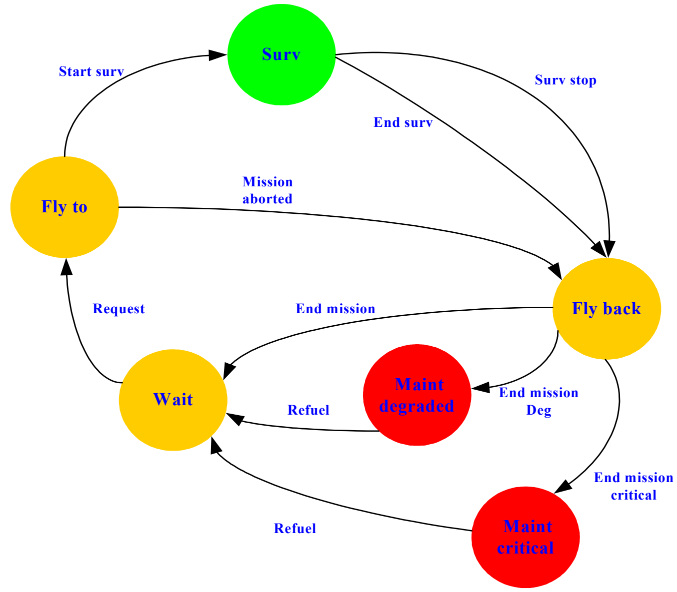

Figure 3. logic of the mission state of a helicopter

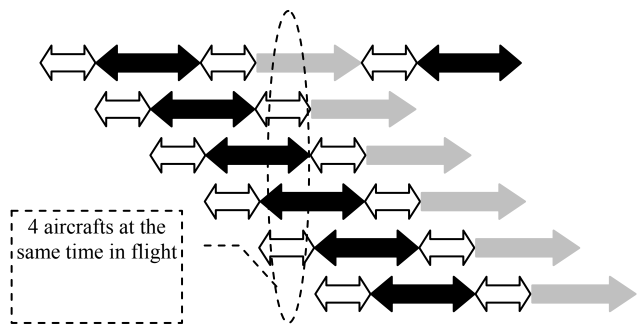

Figure 4. Flight strategy to have always two helicopters in surveillance

	Mean	Standard deviation
Penalty interruption mission	945,000	1,887
Cost maintenance preventive & corrective.	214,013	14,055
Availability	79.71%	10.65%

Table 1 – Simulation results

4.2 Modeling with a spreadsheet

To be able to model the mission on the spreadsheet, a number of assumptions must be made to reduce the number of dimensions of the problem.

Firstly, time dependent phenomenon are impossible to model. Therefore, it is simply impossible to assess the second conditions of the SLA and only the availability can be computed.

Secondly, reconfiguration is also excluded. Previously, it was possible to model a virtual component, which manages the demand for a helicopter. In that case it is simply impossible

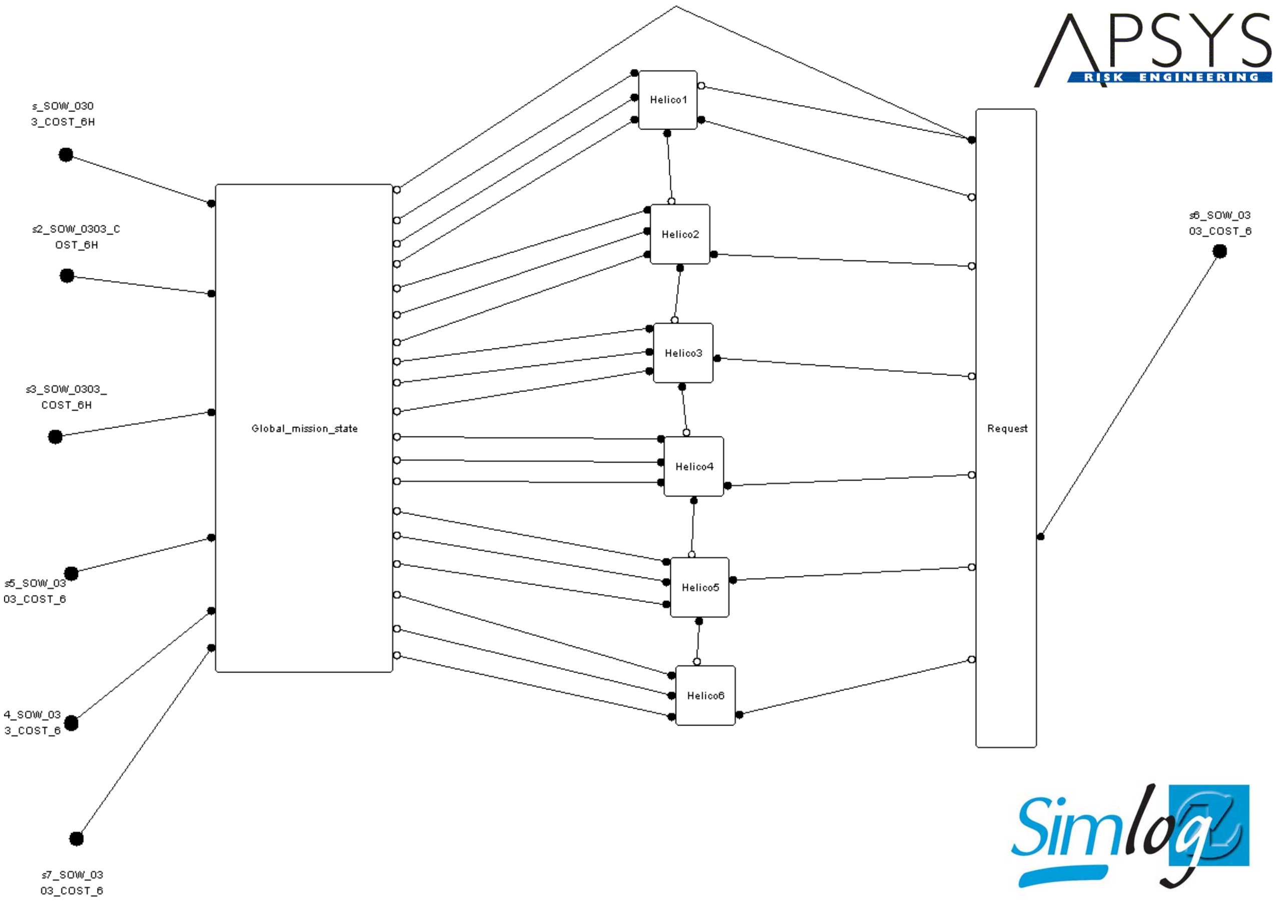

Figure 1. High-level model representation

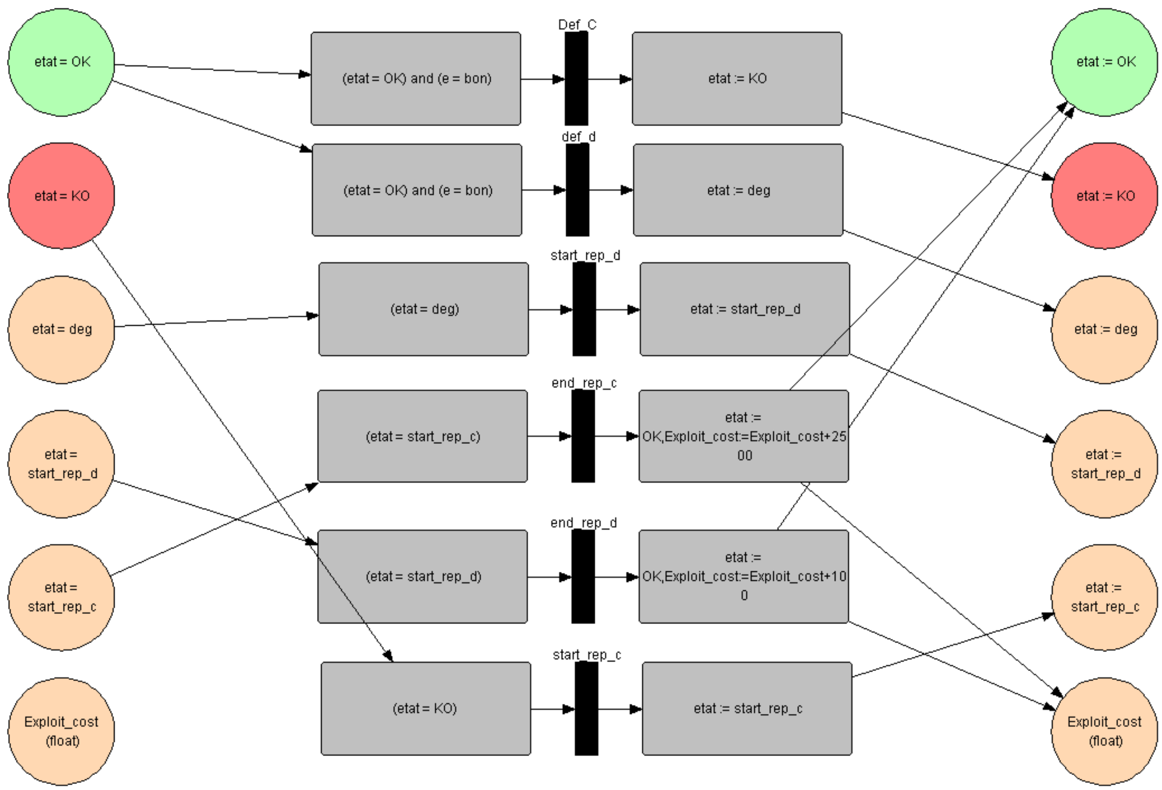

Figure 2. Automata logic of a helicopter

Thirdly, sharing resources and queuing phenomenon are also difficult to model. Therefore, we don't integrate them in the model.

Fourthly, of course it is not possible to realize a Monte Carlo simulation.

With these assumptions, we fill a very simple spreadsheet and calculate the availability of the mission.

The availability computed is 99%.

5 OBSERVATIONS

Due to the number of limitations created by the spreadsheet, the result found is to compare to a result with infinite spares, infinite resources and no time consumed by reconfiguration. Of course this result is not realistic and both calculations clearly shows the interest to use more complex tools to model multi variables, and time dependent phenomenon.

Using Altarica model presents five advantages:

1. The Monte Carlo simulation allows to express the statistical distribution of your performance. In our case, the results (see table 2) shows that if you want to increase your confidence level on your performance from a 50°% confidence level to a 95% confidence level, you need to increase your budget by $2.28 million over the lifetime of the mission (20 years).

Confidence level	50%	95%
Cost of the project over 20 years	$1,235,468	$1,349,264

Table 2 – annual cost at 50% and 95% confidence level

2. Because it is possible to model time dependent phenomenon. It is possible to model conditions such as condition 2 of the SLA.
3. The utilization of virtual components in the models allows to reproduce the decision making process of the operators in selecting the different options to reconfigure a system.
4. The ability to model shared resources and queuing processes for maintenance and flying crews allows to have a more realistic model. The spreadsheet is limited to the modeling of the inherent performances of the system. The Altarica model simulates the performance of the system of systems
5. To link to the previous point, the importance factor on the performance of the mission can be computed. In our case, we have seen that the number of flying crews is a limiting factor. However no constraint has been defined for the maintenance team and facilities but they are probably as important. In the last 20 years, a substantial amount of work has been done to improve the resilience and availability of systems. This has been often achieved by tracking down common modes of failure. Therefore the inherent availability of the systems improved. But in the mean time, a centralization (rationalization) of the maintenance capabilities happened, and if you move your attention from this system to the mission level, you soon realized that the common mode of failure of your mission is not technical but organizational: lack of operators, poor training, lack of resources / spares. Moreover, knowing the limiting factors of your performance could save you from having to spend more money on actions that don't improve it, for example buying an additional aircraft.

However, an Altarica model presents two main drawbacks. Firstly, it is a complex model. It takes time to build the model, to verify and validate it. Therefore, it is an expensive process. To have real added value, the model must be done before the signature of the contract, e.g. during the bidding phase. Therefore, even if the modeling activity allows to better negotiate the SLA, it is a sunk cost that you may not recover in the future.

Secondly, you can't easily reuse the model for another project. Some parts could be reused but the most difficult part to model, which is the translation, the CONOPS of the mission is specific to each project.

6 SYNTHESIS

One very important learning point is that the number of crews are as important as the number of available helicopters to achieve the required SLA. That might seem obvious that a helicopter needs a crew to fly, but this study demonstrates that you have a number of breaking points where the solution to improve performance is no longer to increase the number of helicopters but the number of crews. Therefore your maximum performance is not determined by the number of helicopters but by the number of operators, which depends most of time on the customer. The number of spares, maintenance teams, and maintenance equipment can also become critical. In conclusion, even if your system is highly reliable and resilient, it is impossible to achieve a high level of availability without the right concept of operations and concept of maintenance in place and these aspects must be thought from the initial stages of the project in a systems of systems approach.

From a reliability techniques point of view, this paper shows that simulation packages based on Altarica language allow to model complex phenomenon. Software packages based on spreadsheet calculations don't allow to do that as the following limitations have been identified:

- Difficult to model spares limitations.
- No shared resources, which are a problem as the share resources, can drive the performance.
- No conditional and multiple dependencies relation.
- No time dependent conditions.

REFERENCES

1. A. Griffault, S. Lajeunesse, G. Point, A. Rauzy., J.P. Signoret., and P. Thomas, "The AltaRica language". *Proceedings of the International Conference on Safety and Reliability*, *ESREL'98,* (Jun.) 1998, pp 20-24.

2. A. Griffault, S. Lajeunesse, G. Point, A. Rauzy., J.P. Signoret., and P. Thomas, “Le langage AltaRica In Hermès”, , *Actes du 11ème congrès lambda mu,* (Oct.) 1999.
3. Y. Dutuit., A. Rauzy, and J.P. Signoret, “Réséda: a fiability network analyser.” *Proceedings of European Safety and Reliability Association Conference, ESREL'96,* 1996.
4. Y. Dutuit., A. Rauzy, and J.P. Signoret, “A new notion of prime implicants and its implementation by means of binary decision diagrams” *1st International Conference on Mathematical Methods in Reliability, MMR'97*, 1997, pp 111-119.
5. Y. Dutuit., P. Thomas, A. Rauzy, and J.P. Signoret, “Analyse qualitative et quantitative de la fiabilité d'un système dynamique” *2ème Congrès Pluridisciplinaire Qualité et Sûreté de Fonctionnement*, (Mar.) 1997, pp 243-250.

BIOGRAPHIES

Xavier Quayzin
APSYS
Quadrant House
Celtic Springs, Cleppa Park, Coerdkernew
Newport NP10 8FZ, Unites Kingdom

e-mail: xavier.quayzin@apsys.eads.net

Xavier Quayzin is a principal reliability and safety engineer as well as UK business development manager at APSYS. He is based in Newport (UK) and look after the development of APSYS activity in the Rail, Oil & Gas, Energy, Defense and security domains. Prior to moving in the UK, Xavier worked 6 years for APSYS in Paris where he developed his expertise in safety and reliability in the defense and security industry. Xavier received his Master of Engineering from Grenoble Institute of Technology and he is currently following a MBA course from Manchester Business School. He is a member of IET.

Emmanuel Arbaretier
APSYS
22 Quai Gallieni
92150 Suresnes, France

e-mail: emmanuel.arbaretier@apsys.eads.net

Emmanuel ARBARETIER began his career in THALES (formerly THOMSON) where he was in charge of the adaptation of US Military Standard to the French company ; then he participated in the creation of SOFRETEN where he developed two Software Workshops in the field of Dependability and Logistic Support Analysis named SIMFIA and SIMLOG ; he has been responsible for Research and Development projects for 5 years and finally began COO of SOFRETEN. Since the merger of APSYS and SOFRETEN, he is the Head of the Software Department of APSYS.