Generic prognosis model for proactive maintenance decision support: application to pre-industrial e-maintenance test bed

Alexandre Voisin, Eric Levrat, Pierre Cocheteux, Benoît Iung

GENERIC PROGNOSIS MODEL FOR PROACTIVE MAINTENANCE DECISION SUPPORT:

APPLICATION TO PRE-INDUSTRIAL E-MAINTENANCE TEST BED

Voisin A., Levrat E., Cocheteux P., Iung B.
CRAN, Nancy-Université, CNRS
Boulevard des Aiguillettes B.P. 239
F-54506 Vandœuvre lès Nancy
{alexandre.voisin|eric.levrat|pierre.cocheteux|benoit.iung@cran.uhp-nancy.fr}

Abstract Proactivity in maintenance, which is mainly materialized by degradation-based anticipation, becomes essential to avoid failure situation with negative impact on product and/or system conditions. It leads to make emerging the E-maintenance philosophy to move from "fail and fix" maintenance practices to "predict and prevent" strategies. Within these new strategies, the anticipation action is fully supported by prognosis business process. Indeed it analyses the degradation impact on the component itself but also on the global performances of the production system in order to predict future failures of the system and investigate (future maintenance) actions. However, only few research works focuses on generic and scalable prognostic approach. Existing methods are generally restricted on component view and for solving the failure prediction issue. Consequently, the contribution presented in this paper aims at developing a global formalization of the generic prognosis business process. This generic process can be used after, from an instantiation procedure, to develop specific prognosis processes related to particular application such as shown in this paper with the case of E–maintenance platform developed within DYNAMITE Project.

Keywords Maintenance; e-maintenance; prognosis, e–maintenance platform, MIMOSA.

1. Introduction

Even if maintenance is a necessity, maintenance has a negative image and suffers from a deficiency of understanding and respect. It is usually recognised as a cost, a necessary evil, not as a contributor. Most people think that the role of maintenance is "to fix things when they break" but when things break down maintenance has failed (Blann, 2003). Moreover traditionally the scope of maintenance activities has been limited to the production vs. operation phase. But as the paradigm of manufacturing shift towards realizing a sustainable society, the role of maintenance has to change to take into account a life-cycle management oriented approach (Takata, *et al.*, 2004) for enhancing the eco-efficiency of the product life (DeSimone and Popoff, 1997). In that way, maintenance has to be considered not only in production vs. operation phase but also in product design, product disassembly, and product recycling … (Van Houten, *et al.,* 1998). Thus the product can now play a major role in maintenance mainly when the product is "active" (i.e. Intelligent Product; Holon …) meaning able to support a part of its knowledge. The concept of "life cycle maintenance" (Takata, *et al.*, 2004) emerged to stress this new role leading to develop a general maintenance value chain for which the engineering way is consistent with the system engineering one (INCOSE[1] initiative). The value chain must be supported at each phase of the product life cycle, leading for each phase, to assign a (sub) objective to be fulfilled. Each objective and sub-objective is supported by Business Processes which transform input flows into output flows materialised by resources, information, energies… If all the sub-objectives (local objectives) are carried out, the global value chain is running well. This system view highlights that a not controlled deterioration of one of the product characteristics in a life cycle phase P can have important impact on the use of the product or on its expected service in the life cycle phase P+1 (after P). Consequently, the conventional strategies (Wang, 2002) such as scheduled preventive maintenance strategies carried out some time too late in relation to the current status of the potential failure are not easily compatible with this maintenance vision. A failure can lead to **a bad product or a bad service delivered by the product**. Therefore degradation-based anticipation (the pro-activity in maintenance) becomes essential, at this stage, to avoid failing situation with negative impact on product condition (zero breakdown maintenance).

Pro-activity in maintenance makes emerging the E–maintenance philosophy (Iung and Crespo-Marquez, 2006) to support the moving from "fail and fix" maintenance practices to "**predict and prevent**" strategies (Lee, *et al.*, 2006)… while keeping Maintenance as an Enterprise process (holistic approach) - **Integration concept** (i.e. IEC/ISO 62264) for optimising performances.

E–maintenance (Muller *et al.*, 2008b) is integrating the principles already implemented by Tele-maintenance which are added to the web-services and collaboration vs. synchronisation principles (Iung, 2003) to support pro-activity. Collaboration vs. synchronisation allows not only to share and exchange data and information but also knowledge and (e)-intelligence and this between all the actors (human, units, department…) all along the product life cycle. It leads to develop intelligent and cooperative maintenance architecture implementing all the Business Processes required within the new maintenance value chain.

The Business Processes the most important to develop anticipation action are monitoring, diagnosis, prognosis and decision-making processes. Among these modules, the prognosis process is often considered as the Achilles heel (Wang and Vachtsevanos, 1999) while its goal is fundamental for implementing anticipation capabilities. « We simply do not know how to measure the performance degradation of components and machines; we lack the validated predictive models and tools that tell us what would happen when the process parameters decrease from values» (Lee, 1998).

Indeed, prognosis has to analyse the impact of **degradation** on the component itself and on the other items of the production system to predict future system failures and investigate (future maintenance) actions (Jardine *et al.*, 2006). Nevertheless most of the existing prognosis methods are component-oriented without really taking into account the system performances (Provan, 2003). In addition, (Lee *et al.*, 2006) underlines that:

"In spite of the progress made, many fundamental Issues still remain:

- *Most of the developed prognostics approaches are application or equipment specific. A generic and scalable prognostic methodology or toolbox does not exist.*
- *Currently, methods are generally focused on solving the failure prediction problem. Tools for system performance assessment and degradation prediction have not been well addressed."*

Indeed only few research work cover both these two issues. For example, (Muller, *et al.*, 2008a) proposes a first step to move from prediction vision to prognosis one but without really generalising the process. Thus, the work presented in this paper concerns more a global formalization of the generic prognosis business process (prognosis of the failure) which can be used after, from an "instantiation procedure", to develop specific prognosis processes related to particular components or systems. Then the specific prognosis processes should be implemented on intelligent vs. collaborative maintenance architecture to be integrated with the other maintenance business processes required to develop pro-activity.
In that way, the rest of the paper is organised, consistently with the e–maintenance framework (based on Zachman[2] framework) proposed by (Levrat and Iung, 2007), as follows:

[1] http://www.incose.org

[2] http://www.zifa.com

- section 2 and 3 gives a short overview on the "prognosis" business process definition and the related work already performed.
- section 4 (business process section) presents the formalization of the generic "prognosis" business process. This formalization is based both on the process approach (processing view) and the objects vs. flows that are processed (data view).
- section 5 (organisation section) describes the use of the generic prognosis process for developing the prognosis dedicated to the DYNAMITE[3] E–maintenance platform. For example, a part of the prognosis in DYNAMITE has to be supported by Web Services.
- Section 6 (IT infrastructure section) gives a first description of the e–maintenance infrastructure for running well the specific prognosis process (activities and procedures) previously described (lab experimentation based on DYNAMITE context).

Finally, conclusions and prospective are developed in section 7.

2. Prognosis definition

The word 'prognosis' comes from the Greek 'progignôskein' which means "to know beforehand". This process associated with proactive maintenance has to predict the future state of a system or a component. Several definitions have been proposed to specify its aims.

For (Byington *et al.*, 2002), "Prognosis is the ability to predict the future condition of a machine based on the current diagnostic state of the machinery and its available operating and failure history data." This definition specifies the needs for the prognosis to integrate its informational environment in order to obtain necessary data (current or past data). For (Lebold and Thurston, 2001) and (Farrar and Lieven, 2007), the prognosis has to project the current health of equipment into the future taking into account estimates of future usage. The future usage define a scenario, composed of the future evolution of data which influence the degradation/failure, maintenance actions…

To anticipate 'at best' and to keep all benefits of the proactive maintenance, the prognosis of the future health of the system has to integrate the maximum of available information to be as exact as possible with respect to reality. That allows integrating future real events and not only statistical data. Thus one of the principal aspects of prognosis is the consideration at the prognosis time of all future possible scenarios about system load and interaction system/environment. For an industrial system, scenarios are principally composed of:

- the manufacturing schedule to obtain the future evolution of system loads (functioning mode),
- the environmental conditions and the information about the future evolution of the factors from the environment which influence the degradation/failure,
- the maintenance schedule because maintenance actions influence the health of the system.

Since 2004, the standard ISO 13381–1 gives a "textual" framework of the prognosis. Prognosis is defined as: "estimation of time to failure and risk for one or more existing and future failure modes." This definition doesn't consider the needs to include a previous scenario, but it adds two important complementary aspects. First the standard specifies that the output of the prognosis process is composed of several remaining useful lives (RUL). Second, the ISO definition uses the notion of current failure mode and potential failure mode (its occurrence is in the future).
These definitions are for the authors incomplete. Therefore the prognosis process will be considered in this article as the process which has to:

- predict the future health of the system,
- generate the different RULs (of the system, part of the system or component) for each detected (current) or potential degradation/failure mode,

by taking into account the knowledge of the system (functional and dysfunctional), past information (background), current information (current state) and future information (scenario with manufacturing and maintenance data).

3. Related work

Prognosis is a process issued from aims and principle of proactivity firstly in the Condition Based Maintenance (CBM) and now in E–maintenance. In such visions, prognosis is in interaction-collaboration with some others business processes (condition monitoring, diagnosis, decision support…) by exchanging-sharing information-knowledge.
In OSA-CBM project, (Open Systems Architecture for Condition-Based Maintenance) (Lebold and Thurston, 2001), an architecture was defined (see Fig. 1). The structure is composed of seven layers which constitute a linear succession of sub-processes, ranging from the data acquisition to the Decision Support. The prognostics layer is located between the health assessment layer and the decision support layer. The health assessment layer includes the process which allows to define the current degradation/failure level (system or component), and the diagnosis process which gives the current degradation/failure mode(s). The decision support layer assists the decision making in order to choose a relevant maintenance action to restore the system in a previous state. The principal data exchanged between the different layers are identified and modeled within MIMOSA framework.

This OSA-CBM model identifies five informational flow classes like inputs of the prognosis layer (see Fig.2):

[3] European Integrated Project DYNAMITE 017498 (Dynamic Decisions in Maintenance)

- knowledge about the past functioning (operational usage/load, history and projection, historical failure rates, maintenance history, CM (Condition Monitor), HA (Health Assessment) and PR (Prognostics) history),
- system information (application knowledge),
- future operational conditions (configuration set-up),
- current health (HA,CM, SP (Signal Processing) and SM (Sensor Module) input),
- prognosis control (PL control vector).

This model also defines the output classes of the prognosis layer:

- the prognosis result could take different forms: RUL, RUL distribution, future performance of the system (PLData),
- the prognosis result tags allow to tag a prognosis result with its control data (PL control vector) and with the used algorithm and prognosis interval definition (PL Configuration),
- explanation on the prognosis result (PL Explanation),
- update of the history (PR history).

In a more formalized way between the business processes (Léger and Morel, 2001) proposed an architecture for an integrated system of proactive maintenance (SIMP). This one is composed of three processes: the condition monitoring and diagnosis process, the prognosis process and the Decision Support process (DS). One of the particularities of a SIMP is to propose a loop between prognosis and DS. In this context the prognosis allows evaluating different maintenance or manufacturing options and allows DS to make the 'best' choice to prevent a failure.

(Muller *et al.*, 2008a) globally define the nature of exchanged data between the SIMP and the different processes of the environment (Fig. 3).
In summary, the papers about the prognosis process modeling are very general and describe the links between input and output flows but don't propose an efficient formalization. Moreover the generic sub processes needed for the prognosis have not been fully identified. To face with this issue, our work aim at proposing a formalization of a generic prognosis process in the framework of an integrated system of proactive maintenance. This formalization based on a process approach allows making the exchanged data between prognosis and its environment explicit. It also allows to define generic sub processes and to define the data exchange between those sub processes. Thus the originality of our contribution is to lead to a generic model of prognosis (generic component) which can be instantiated to a lot of applications for developing specific and consistent prognosis (particular component based on generic one).

4. Formalization of the generic prognosis business process

The formalization of the prognosis business process is based on the concepts developed (a) in the process approach (process identification, process decomposition, identification of the flows produced/consumed by the process…) and (b) in the modeling of the objects/data linked to these flows. The process approach is supported by the MEGA Suite[4] and the Object modeling is supported by UML classes and sequence diagrams.

The formalization is developed on five steps as described in this section:

- formalization of the prognosis process environment, which allows identifying the external partnerships of the prognosis and the external flows exchanged. The main output flow of the prognosis is the Remaining Useful Life (RUL),
- formalization of the final purpose of the prognosis process which consists in defining the rules to calculate the RUL in a general way,
- formalization of the functional decomposition of the prognosis process (sub processes identification),
- formalization of the coordination of the sub processes needed to fulfill the prognosis mission (sequence view),
- formalization of all the objects/data materializing the flows (internal and external to the process).

4.1. Formalization of prognosis process environment

The identification of the prognosis environment is based on an improvement of the results of (Muller *et al.*, 2008a). Thus Fig. 4 is an extension of the SIMP concept Fig. 3, where is added several interactions with others enterprise processes. This extension leads to the modeling of the following business processes: "To acquire and to process signal", "To manage operations", "To manage maintenance", "To manage company". These business processes are representative of the operational, tactical and strategical levels. The interactions between the external business processes and the prognosis process could be summarized as follows:

Main output flow of the prognosis process:

- *RUL + confidence level + future performance system.*

Main input flows of the prognosis process:

- *Fault localisation + Degradation/Failure mode (O1P2)*: (to be referred to the Fig. 6, 7, 8, 9, 10, 11). This input informs about the current degradation/failure mode for the both level (component and functional). This information allows choosing the more pertinent degradation/failure model. Moreover, when a new degradation/failure mode occurs, a new prognosis is performed (the reference instant t=0 change).
- *Data about system and environment (O1P1-3).* These data represent the current health (or degradation/failure level) of the system and of its components. This data are segmented and filtered by "To acquire and to process signal" in order to

[4] http://www.mega.com

contain only the representative parts (e.g. data from a break are eliminated).

- *Company strategies (priority, costs, security, prognosis horizon, etc) (O1P7)*. This input specifies the prognosis horizon of the temporal window $[t_0,T]$. The horizon depends of the company's strategic choices. If a stock of spare parts exists or if a part of maintenance is contracting out, the horizon for the decision making and the prognosis is decreasing.
- *Historical data (O1DB)*. This input contains the historical knowledge of the system and allows obtaining models of degradation/failure mode.
- *Prognosis knowledge (I1P3)*. It is used during the engineering and development phase. This knowledge allows obtaining the degradation/failure mode from the historical data by identifying the parameters of the component degradation/failure mode and by modelling the impact of the component degradation/failure modes on the system degradation/failure mode, the impact of the maintenance actions, the impact of the operational conditions…

Some of other input flows are used for developing the scenario of operation condition and maintenance which has to be known on the prognosis horizon $[t_0,T]$. These additional inputs are:

- *Manufacturing schedule + environment on* $[t_0,T]$*(O1P5)*. This input is composed of the production loads and their durations. In addition the environmental conditions during $[t_0,T]$ are able to influence the degradation/failure (e.g. the outside temperature, hygrometry, systems which are located near the system and which create vibrations or electromagnetic vibes, etc).
- *Maintenance schedule on* $[t_0,T]$*(O1P6)*. This input is composed of the list of the maintenance actions which are initially planned (systematic maintenance). The interest of these data comes from the impact of these maintenance actions on the health.
- *Prognosis request (O1P4-3a) + option of scenario (O1P4-3b)*. This input performs the loop between the prognosis process and the DS process. It allows DS process to evaluate several options of maintenance actions or of manufacturing scheduling. So to prevent the failure, the scenario used by "To prognosticate" is composed from an original scenario with the initial manufacturing and maintenance schedules and from one option of scenario. Then the DS process proposes several options of scenario that the prognosis tests successively. The prognosis generates for each option a new list of RUL and then DS process can compare these options.

4.2. Formalization of the prognosis final purpose

As previously defined and according to the standard ISO 13381−1 the prognosis purpose is to generate the remaining useful life (RUL) of the system and of each component. The getting of RUL is carried out upon a temporal window $[t_0,T]$ (t_0 is the reference instant of prognosis and T is the final time of the prognosis which corresponds to $T=t_c+\Delta T$ with t_c the current time and ΔT the prognosis horizon). A RUL is defined thanks to an instant in the interval $[t_0,T]$ for which the projected value reaches a threshold. Two classes of RUL have to be defined: a 'component-RUL' that corresponds to a failure of the component, and the 'functional-RUL' that corresponds to a loss of a function of the system (final purpose). Therefore there are two kinds of thresholds: the component threshold and the functional threshold.
For a component, the values projected by the prognosis process are the degradation/failure mode levels and the performances. The component threshold corresponds to the shut-down of the component (ISO 13381−1). On the other hand, the functional threshold corresponds to the performance limit for which the final purpose of the function is not guaranteed (critical threshold). Usually, this threshold is defined by the system's requirements and functional specifications. Fig. 5 summarizes both notions.

For a component, several degradation/failure modes can simultaneously exist. Therefore it is necessary to generate a RUL for each degradation/failure mode. Moreover there are two kinds of degradation/failure mode: (a) the proved mode, i.e. the mode which exists at prognosis time and which was detected by the diagnostics, and (b) the potential mode, whose possible occurrence appears after prognosis time. The proved modes are located upon the components that are already degraded, and the potential modes are located either upon already degraded component, or upon components which are not degraded at the prognosis time yet.
Thus we can define for each degradation mode (proved or potential) of each component a RUL_{CMD} which represent the remaining time before the shut-down of the component C due to the degradation mode MD. The set which contained all RUL_{CMD} is able to take the form of the union of three sets of RUL:

$$\{RUL_{Ci\,MDk}\}\cup\{RUL_{Ci\,MDl}\}\cup\{RUL_{Cj\,MDm}\} \qquad (1)$$

with:

- RUL_{CMD} is the RUL of the component *C* with the degradation/failure mode *MD*,
- $C_i\in\check{A}$, $\check{A}$ is the set of already degraded components which are detected by the diagnosis process,
- $C_j\in\check{O}$, $\check{O}$ is the set of components, which are not degraded yet and therefore not detected by the diagnosis process, but for which prognostic predicts one or more apparition(s) of potential mode(s),
- $k\in\{1,\ldots,K_i\}$, K_i is the number of degradation/failure modes which are proved upon the component *i*,
- $l\in\{1,\ldots,L_i\}$, L_i is the number of degradation/failure modes which are susceptible to appear on the component *i*,

- $m \in \{1,\ldots,M_j\}$, M_j is the number of degradation/failure modes which are susceptible to appear on the component *j*.

Moreover two kinds of components are distinguished: the components which can't be repaired, i.e. those components are replaced after failure, and the components which are repairable. In the former case, only one component-RUL is required and is given by the RULs of each degradation/failure mode of the component:

$$RUL_{Ci} = \begin{cases} g\left(\left\{RUL_{Ci\,MDk}\right\} \cup \left\{RUL_{C\,j\,MDm}\right\}\right) & if\ C_i \in \breve{A} \\ g\left(\left\{RUL_{Ci\,MDl}\right\}\right) & if\ C_i \in \breve{O} \end{cases} \quad (2)$$

where *g* is a function to aggregate the RULs. In the case of the form of RULs is a date or a number the function g is the function minimum because when the component has failed all degradation/failure modes stop their progression.
In the latter case, the prognosis process has to give a list of RULs, where each RUL is a component-RUL for one degradation/failure mode (proved or potential) for one component. The list is composed of every RUL for each proved or potential degradation/failure mode. The knowledge of the RUL of each degradation/failure mode allows the DS process to propose maintenance actions to group together potential interventions in the global framework of opportunistic maintenance (Thomas *et al.*, 2006).
For a given function of the system, the final purpose is characterised by demands on the product and its flows as well as demands on the manufacturing system. The demands on the product and its flows concern parameters of the production like the product quality, rating, etc. The demands on the manufacturing system concern the efficiency of means with respect to the result. For example, a system with a 'feedback loop' is susceptible to derive without visible effects upon its output. The final purpose of a function are affected either by an external cause (deviation of a input flow) or by an internal cause (degradation of a sub-function). Thus the degradation modes impact directly on the decrease of the performance of the functions and a RUL of the function *Fi* with the degradation/failure mode *MD* can be defined by:

$$RUL_{Fi\,MDj} = f(CV, Perf_{Fi\,MDj}, Perf_{Fik\,MDj}) \quad (3)$$

with:

- RUL_{FMD} is the RUL of the function *F* with the degradation/failure mode *MD*,
- *CV are the covariate which come from input flows*,
- $Perf_{Fi\,MD}j$ is the performance ok the function F_i with the degradation/failure mode MD_j,
- $Perf_{Fik\,MD}j$ is the performance of the sub-function F_{ik} with the degradation/failure mode MD_j.

From these RUL a functional-RUL, affected by *n* degradation modes, can be obtained with a aggregation function:

$$RUL_{Fi} = f(RUL_{Fi\,MDj}) \quad j \in [1,n]. \quad (4)$$

where *f* is a function to aggregate the RULs. In the case of the form of RULs is a date or a number the function *f* is the function minimum.

Moreover for each RUL (component- or functional-RUL) a confidence level has to be associated (ISO 13381−1) and is defined as follow: "*A confidence level is a figure of merit (percentage) that indicates the degree of certainty that the diagnosis/prognosis is correct. This value is essentially a figure representing the cumulative effect of error sources on the final certainty or confidence in the accuracy of the outcome. Such a figure can be determined algorithmically or via a weighted assessment system.*" This definition looks restrictive because the confidence level is characterized by a confidence interval of level α or by a density function of the variable RUL.

In the specific output flow of "To prognosticate" which is named "*RUL + Confidence level+Futures performances system (O1P3)*", we need to complete the standard proposal by adding the system performances (object) to this result flow. The knowledge only composed of RULs do not allow the DS process to use a specific criterion upon the performance in addition to criteria related to costs, security... This criterion has to be assessed for all the options of the future scenario in order to classify them.

4.3. Formalization of the functional decomposition of the prognosis

For achieving the RUL objective in relation to its input flows the prognosis needs four sub processes (Fig. 6). The sub process "To pilot prognosis" coordinates the three others, ("To initialize state and performances", "To project" and "To compute RUL") which constitute the sequential stages to obtain a prognosis result.

4.3.1. "To pilot prognosis"

In addition to its coordination activity, the sub process provides the appropriated models in relation to the current situation, i.e. the proven degradation/failure modes. This process has **two** types of output flows:

- Request for action to be sent to the three others sub processes (initialisation (O2P1-2a), projection (O2P1-3a) and calculation requests (O2P1-4a)),
- Appropriated models to be provided to the three others sub processes (functional and dysfunctional (O2P1-2b), projection (O2P1-3b), threshold models (O2P1-4b)).

… and several input flows :

- *Fault localization + Degradation/Failure mode (O1P2)*,
- *Prognosis request (O1P4-3a)*. This input allows to this sub process to trigger only the two other sub processes ("To project" and "To compute RUL") when the DS process send a request (of prognosis) for a new option of scenario,

- *Prognosis knowledge (I1P3)*.

4.3.2. "To initialize state and performances"

The purpose of this sub process is to define an initial situation for the projection. It updates from the last real data of the physical system, the image of the health of the system and the degradation/failure level. This image represents a synchronic view of both the system and its components. This process has one output flow composed of:

- the current health of components which is represented by the level of each proven degradation/failure mode,
- the components or system performances given by indicators, which allow quantifying the activity of the system or the component in relation to it final purpose.

… and three input flows:

- *Data about system and environment (O1P1-3)*,
- *Initialisation request (O2P1-2a)*: it is generated by "To pilot prognosis",
- *Functional and dysfunctional models of the system (O2P1-2b)*: these models are generated from a functional analysis based on a process approach and a dysfunctional analysis: FMECA associated with HAZOP (Muller *et al.*, 2008a). The behaviour of a process is described by causal relations who joined inputs, physical support and outputs.

4.3.3. "To project"

This sub process determines the future evolution of the degradation process and the component or system performances (projection in the time, diachronic view of the system). The projection integrates the influence of the different data (operational condition, maintenance actions…) from the scenario. This process has one output flow composed of:

- Future level of degradation/failure
- Uncertainty level (confidence degree) on the result

… and the following inputs:

- *Projection request(O2P1-3a)*: comes from "To pilot prognosis",
- *Current level of degradation/failure + Performances (O2P2)* ,
- *Projection models (O2P1-3b)*. The projection models have to respect three key points of the projection sub process. The first one concerns the modelling of the impact of the scenario. The second one is the projection of the future health and performances with the impact of the scenario from the first point. The last key point is the evaluation of the capacity of the potential degradation/failure mode to occur.
- *Manufacturing schedule + environment on $[t_0,T]$ (O1P5)*,
- *Maintenance schedule on $[t_0,T]$ (O1P6)*,
- *Option of scenario (O1P4-3b)*.
- *Company strategies*

4.3.4. "To compute RUL"

This last sub process computes the RULs which are the final purpose of the prognosis. For providing the RUL it requires three input flows:

- *Calculation request (O2P1-4a)*,
- *Future level of degradation/failure + uncertainty (O2P3)*: this input is the result of the projection sub process ,
- *Threshold models (O2P1-4b)*. The thresholds are models because they integrate the uncertainty linked, for example, the variability between inter-components. They can be represented under the form of distribution.

From the flows previously identified and mainly the event flow (requests) which allow to trigger each of the sub processes, the four processes have to be executed in a sequential and consistent way in order to achieve the prognosis objectives (RUL providing). This sequential way can be described in form of a sequence view and then modelled by a sequence diagram.

4.4. Sequence view of the Prognosis

The sequence view proposed here-after describes the three different stages needed for the prognosis but only in a textual way (a particularization of the formalized sequence diagram can be found Fig. 12). The prognosis is activated by means of the request corresponding to (a) an external demand or (b) an internal demand (SIMP) coming from the DS. (see Fig. 4).
According to the prognosis approach (Byington *et al.*, 2002), t_0 is defined, e.g. in experience-based approach, t_0 is the last As Good As New maintenance action. It usually depends on maintenance action performed on the asset. Hence the first step is to request the "To manage maintenance" process for knowing the past maintenance actions performed on the asset and determine t_0. Then "To manage company" process is requested to assess the prognosis horizon ΔT which depends on company, operational and maintenance strategies. Knowing the degraded asset and the degradation mode, the "To manage maintenance" process is requested to provide the associated degradation model composed of equations, influence variables, degradation modes induced with their causality and occurrence probability levels. Nevertheless the degradation model is slightly different depending on the selected prognosis approach (data, reliability model…).

From the degradation model it is necessary now to define the prognosis scenario to be used for projection. The scenario is mandatory to obtain a realistic prognosis. In that way "To manage production" and "To manage

maintenance" processes are requested for knowing the operational and environmental conditions but also the maintenance actions during the period $[t_0,T]$. All these data are then processed to build the prognosis scenario in a continuous time (for example, data associated to stopping times and maintenance times are deleted).

From the degradation model and the scenario, the projection step is launched. It consists in computing for each time from t_0 to T, a degradation (health) function, according to the impact of the operational, environmental and maintenance conditions on the degradation model. Uncertainty related both to modeling and forecasting are integrated in the degradation function (Fig. 5).
From the degradation function, and taking into account the limit threshold provided by "Threshold models" (O2P1-4b Fig. 6), the elementary RULs associated to each degradation mode are computed. These RULs are then aggregated to form the global RUL and to compute the uncertainty.
The well execution of the stages previously described is based on a well exchange (of flows) between the different processes: interoperability (Panetto, 2005) between processes i.e. maintenance ontology. It implies to formalize, not only on the syntaxic point of view but also on the semantic one, the objects/data materializing the flows.

4.5. Formalization of the Prognosis objects and Data

This formalization is based on MIMOSA–OSA/CBM Standards (Machinery Information Management Open System Alliance) (Lebold and Thurston, 2001) for which we propose some extensions in order to be consistent with our generic prognosis model. The formalization uses UML classes diagram representation, and leads to a unique model which is presented below in five parts to be more readable.
The first part deals with the objects of the system architecture on which the prognosis has to be deployed. The occurrences of the object "segment" materialize the functions (and sub functions) fulfilled by the system. The occurrences of the object "asset" materialize the sub systems and components which support the functions. The objects "segment" and "asset" are associated to the "*Fault localization +degradation/failure mode"* flow (O1P2 in Fig. 4) for identifying the function or the component degraded.

The second part defines the diagnostic knowledge which encompasses information of degradation mode. It links the current degradation modes detected by the diagnostic process to its localization in a segment or an asset. Thus the object "DegradationMode" belongs also to the "*Fault localization +degradation/failure mode"* flow.

The third part concerns Maintenance and Production knowledge and Decision Support scenarios. It is related to the "*Manufacturing scheduling+environmental"* flow and "M*aintenance scheduled"* flow. The objects of these flows are the bases for building the scenario object (this object has been added to MIMOSA proposal) for which is also considered the objects coming from the decision making feedback.

A scenario is divided in several work orders. A work order corresponds to an atomic time sequence where a single functioning mode operates, i.e. production stop, maintenance stop, functioning … A work order pertains to assets and segments. According to the present time, past work order and future scheduled work order can be separated. When an alternative scenario has to be evaluated for the Decision Support, it contains the same past works order and alternative future works order. A work order is composed of several variables, related to segment or asset, and linked to their data according to their type. A special type is condition monitoring data if such prognosis approach is implemented.

The fourth part (prognosis knowledge) underlines two main objects for computing the RUL. The "Degradation level" object is associated to the "C*urrent level of degradation/failure"* flow (O2P2 Fig.6) and "*Future level of degradation/failure"* flow (O2P3 Fig. 6). The "Model" object stores the models information needed for the projection, i.e. input, output, parameters and equation. The ModelInput class is linked to the Variables class of the work order or to the "Degradationlevel" class.

The models are specific to a unique segment or asset and a unique degradation/failure mode. A 3-ary association links asset/segment, degradation mode and model since prognosis deals only with couple of degradation mode-asset/segment for which a model is known.

The last part identifies the prognosis output model which is directly issued from the OSA-CBM model. It has been completed with association of the "Itemprognosis" class to the asset/segment, degradation mode and scenario objects. Such links are compulsory since it allows Decision Support to handle its several alternative scenarios and opportunistic maintenance (Iung *et al.*, 2007).

This phase of Object formalization ends the proposal of the generic prognosis process modeling (conceptual view). The next step now is to show how this generic model can be used for a specific organization. In a theoretical way this use corresponds to an instantiation procedure of the generic model for developing specific model dedicated to a selected organization (organization view).

This phase is still under development implying that the next section is focusing more on the guidelines needed for starting instantiation.

5. Prognosis model for Dynamite E–maintenance organization

Defining a maintenance organization consists of assigning business processes (and their decompositions into activities) to the actors that will perform these activities on a specific site. An actor represents a person or a machine participating in business processes of the

enterprise. A site could be internal or external with the company and can prejudge a subcontracting of competences. The sequence of operations (procedure) carried out by the actors provides the final purpose expected.
During the development of this sequence if some activities of a same business process are distributed among different actors, located on different sites, it will require to create new operations in order to re-establish the right behavior of each business process. These new operations must carry out communication, storage, collaboration, negotiation…
In that way, for developing a specific prognosis business process on an organization it is required:

- To identify the actors,
- To identify sites on which the actors operate,
- To identify activities (sub processes) performed by each actor,
- To create if necessary new operations dedicated to business processes distribution,
- To model the sequence of operations (organizational workflow between the actors), needed to fulfill the prognosis as a whole.

The organization we selected to show the deployment of our generic prognosis model is those of DYNAMITE E–maintenance organization.

The IP DYNAMITE n° 017498 project aims at promoting a major change in the focus of condition based maintenance, taking full advantage of recent advanced information technologies related to hardware, software and semantic information modeling. Special attention is given to the identification of cost-effectiveness related to the upgraded CBM strategies, as well as to the inclusion of innovative technologies within CBM. The DYNAWeb is the ICT architecture concerning software web services and communication architecture that intends to provide support to the new maintenance concept, related mainly to the operational layer of the maintenance system. The experimentation sites in DYNAMITE are dedicated in automotive (FIAT, VOLVO) and machine tools sectors (GORATU)

In DYNAMITE E–maintenance organization the actors are distributed on three different layers:

- a maintenance decisional layer considered as STRATEGIC level.
- a maintenance management layer considered as TACTICAL level.
- a maintenance operational (local) layer considered as OPERATIONAL level.

For each of the three maintenance layers the following maintenance actors could be identified:

- At the Decisional layer : a remote or local maintenance expert actor
- At the Management layer : a CMMS actor manage all the life cycle of the maintenance work-orders in accordance with selected maintenance strategy (it contains knowledge about maintenance actions, past and planned, degradation models, limit threshold, …)
- At an Operational layer :
  - The Computerized Maintenance Operational System (CMOpS) supports a lot of maintenance dynamic processes for selecting the best maintenance work-order and supporting it, provides condition monitoring data and past prognosis results. It is in charge of monitoring the asset when an alert occurred, and to launch the condition monitoring, diagnosis and prognosis services.
  - MES actor provides information about past and planned work order and operations conditions.
  - SCADA actor provides the environmental conditions.
  - A mobile maintenance actor (equipped with an intelligent PDA), is in charge of monitoring the asset when an alert occurred, and to launch the condition monitoring, diagnosis and prognosis services. PDA assists the maintenance operator in carrying out it everyday tasks.
  - Sensors actors on the plant deliver data and information on machine status and are able to generate an alert.
- Web actors support Web services and can be call from the three layers.

Among all these actors, those concerned by the deployment of the prognosis process are:

- Web actor, outside the company,
- The CMMS actor, located off site in the business area,
- The CMOpS actor, located on site but not closed the machine,
- The MES/SCADA actor, located on site for developing the right production actions (in synchronization with maintenance),
- The PDA actor, located on site and very closed the machines,

The identification of the activities supported by each actor and consequently the sequence required to develop the prognosis are directly generated from the generic (functional and sequence) views. For example, the sequence diagram of an implementation of the prognosis in the case of DYNAMITE is illustrated in Fig. 12 and fully based on the textual description done in section “Sequence view of the Prognosis”. In this diagram, the specific DYNAMITE actors exchange flows (containing objects), which are occurrences of the generic flows and which are composed of occurrences of the generic objects (Fig. 6,7,8,9,10,11). As these actors are not located on the same sites, it is necessary for running well the sequence to add communication operations for supporting interoperation.

Mise en forme : Puces et numéros

6. E–maintenance infrastructure for supporting Prognosis

The prognosis generic model has now to be implemented on a particular infrastructure (intelligent vs. collaborative maintenance architecture). This infrastructure consists of a set of specific components (hardware, software, hybrid ...) supporting each part of the sequence and materialising the information technology (IT) means required for running applications and for enabling communication between these applications according to their distribution on sites.

The IT infrastructure is thus defined by all the resources (applications, services, protocols…) necessary to the execution of all the operations identified at the organizational level. In that way, it is composed of one or several networks with servers, workstations, applications, databases but also (smart) sensors, PDA… It is also characterized by its operating principles (wireless infrastructure, highly fault-tolerant, secured…) and the concrete implementation of a technological interoperability.

Several e–maintenance infrastructures already exist such as ICAS-AME, WSDF, PROTEUS (Bangemann *et al.*, 2006), QUESTRA, ENIGMA… Based on the concepts of our TELMA platform (on which is implemented most of the e–maintenance principles), we developed in our lab a first new IT infrastructure for supporting DYNAMITE prognosis. Its specificity lies in wireless communication and web services technologies (Fig. 13)

.

At the field level, the communication is based on a Zigbee area for the interactions between sensors and PDA (HP iPAQ 4700) or CMOpS (TESSNET software of Tekniker[5]). At the business level, the communication is based on a Wifi/Ethernet area for the interactions between PDA or CMOpS and CMMS, MES, both supported by a MIMOSA[6] database, and web services. A Gateway makes the link between the Wifi area and the Zigbee area.

In accordance with sequence diagram (Fig. 12) each activity supported by an actor led to develop software module. Moreover, for communicating the required objects between the actors, the software module is linked to a communication software (a protocol) interacting with the wireless technology. For example, the HP iPAQ 4700 is equipped with a Wifi card but also a Zigbee CF card (i.e. link with sensors) and a Zigbee CF card (link with Smart Tag).

According to DYNAMITE architecture, the prognosis service must be implemented as a web service that is located outside the enterprise. Such architecture is of interest since it allows to share prognosis knowledge for several equipments from several sites and to use the most up to date available services. But, there are 2 main drawbacks to such view. The first is that the data about the processes are confidential and must not be made available outside the enterprise. So the enterprise information system is protected using Firewall meaning that the web service cannot access directly intra-enterprise data and that specific contextualized data must not be sent outside the enterprise. And secondly, the huge amount of data to be send to the web service for a complete prognosis of a system is not well fitted to web service architecture. Under these hypothesises, the functional view of figure 6 has to be mapped to the DYNAMITE architecture. The prognosis service is split into 2 services: a distant web service and a local agent. The local agent will take in charge part of the functions to pilot prognostic (2P1) and to initialise state and performances (2P2) and the web service the remaining functions. In this sense, the web service serves as a basic prognosis function dedicated to prognosticate a simple asset/function RUL for a single failure/degradation mode. In this implementation, the management for prognosticate sub-function, function or asset from basic asset/component has to be performed by the local agent. Moreover, it will not be possible to consider interaction between concurrent failure/degradation modes.

The agent has to manage all the local “point of view” which defines the context such as asset/segment informations, models, data… maintenance actions, production and environmental conditions. The agent must extract this information from the MIMOSA database and structure it to call the web service. It must also keep a data structure to manage the prognosis of the whole system/sub system under consideration. This is performed in the MIMOSA database since in the DYNAMITE project, a unique scenario, i.e. the production/maintenance plan, is investigated and the prognosis output information can be stored in the MIMOSA database. The information send to the web service do not embed any link to specific process/asset and can be safely send to the web service. Its structure is derived from the generic UML class diagram and is presented figure 14.

The Web Service is invocated by the agent and compulsory asset/function data (Fig. 14) is provided by mean of a XML file since it represent high level information and not only input parameters of the web service. In the same way, the web service returns the result back to the client using a XML file. The web service is then composed of three stages: to parse out XML input data into usable format, to prognosticate and to parse in XML output data.

7. Conclusion

In consistence with the e-maintenance framework proposed by (Levrat and Iung, 2007), this paper develops the formalization of the generic “prognosis” business process. This formalization is based both on the process approach (processing view) and on the objects vs. flows that are processed (data view). In particular, initial OSA-CBM models are completed with specific prognosis classes presented in Fig. 7, 8, 9, 10, 11 and they are used to define the basis of a domain model or ontology of prognosis in a CBM system.

[5] http://www.tekniker.es
[6] http://www.mimosa.org

Then the generic process is used (by a first step of instantiation) for developing a specific prognosis dedicated to the DYNAMITE E−maintenance organisation. It gave a first description of the e−maintenance infrastructure for running well the specific prognosis process (lab experimentation based on DYNAMITE context). By using the required IT components, a first set of experimentation is currently in progress in order to assess the performances of the prognosis implemented trough the infrastructure, but also the flexibility of such IT architecture to support different levels of proactivity.

The main prospective is to end (a) the scientific work on instantiation phase but also (b) the experimentations to show the real added value (and the constraints) of this prognosis approach.

Acknowledgments

The research reported here is partially supported by the European Integrated Project DYNAMITE (Dynamite Decision in Maintenance). The authors are grateful to all the partners who contributed by constructive discussions to the emergence of this work.

References

Bangemann, T., Reboul, D., Scymanski, J., Thomesse, J-P., & Zerhouni, N. (2006). PROTEUS - An Integration Platform for Distributed Maintenance Systems. *Computers in Industry*, Special issue on e-maintenance, 57(6), 539—551.

Blann Dale R., (2003). Reliability as a Strategic Initiative: To Improve Manufacturing Capacity, Throughput and Profitability Asset Management & Maintenance Journal, 16(2). http://www.marshallinstitute.com/inc/eng/justforyou/Body/articles/reliability_str_init.pdf

Byington, C.S., Roemer, M.J.,& Galie, T. (2002). Prognostic Enhancements to Diagnostic Systems for Improved Condition-Based Maintenance. *IEEE Aerospace Conference Proceedings*.

DeSimone, L.D. & Popoff, F., with The WBCSD (1997). Eco-Efficiency. *MIT Press*.

Farrar, C.R., & Lieven, N.A.J. (2007). Damage prognosis: the future of structural health monitoring. *Philosophical transactions of the Royal Society A*, 365, 623—632.

Iung, B., (2003). From remote maintenance to MAS-Based E-maintenance of an industrial process. *International Journal of Intelligent Manufacturing,* 14(1), 59—82.

Iung, B., & Crespo Marquez, A. (2006). Special issue on e-maintenance, *Computers in Industry*, 57(6), 473—606.

Iung, B., Levrat, E., & Thomas, E. (2007). 'Odds Algorithm'-based Opportunistic Maintenance Task Execution for Preserving Product Conditions. *Annals of the CIRP - Manufacturing Technology*, 56 (1), 13—16.

Jardine, A., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20 (7), 1483—1510.

Lebold, M., & Thurston, M. (2001). Open Standards for Condition-Based Maintenance and prognostic systems. *5th Annual Maintenance and Reliability Conference (MARCON 2001)*, Gatlinburg, USA.

Lee, J., (1998). Teleservice engineering in manufacturing: challenges and opportunities. *International Journal of Machine Tools and Manufacture*, 38(8), 901--910.

Lee, J., Ni, J., Djurdjanovic, D., Qiu, H., & Liao, H. (2006). Intelligent prognostics tools and e-maintenance, *Computers in Industry*, Special issue on e-maintenance, 57(6), 476—489.

Léger, J.B., & Morel, G. (2001). Integration of maintenance in the enterprise: towards an enterprise modeling-based framework compliant with proactive maintenance strategy. *Production Planning and Control*, 12(2), 176--187.

Levrat, E., & Iung, B. (2007). E-maintenance: Overview and Conceptual Framework. Submitted to *Production Planning and Control*, special issue on Maintenance and Facility Management organised by Pr. M. Garetti.

Muller, A.., Suhner, M.C., & Iung, B. (2008a). Formalization of a new prognosis model for supporting proactive maintenance implementation on industrial system. *Reliability Engineering and System Safety*, 93(2), 234—253.

Muller, A., Crespo Marquez, A., & Iung, B. (2008b). On the concept of e-maintenance: Review and current research. *Reliability Engineering & System Safety*, in Press.

Panetto, H., (2005). Interoperability of enterprise software and applications: Workshops of the INTEROP−ESA International Conference EI2N, WSI, ISIDI and IEHENA 2005. *Hermes Sciences Publishing.*

Provan, G., (2003). Prognosis and Condition-Based Monitoring : an open systems architecture. *5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 57--62, Washington, USA.

Takata, S., Kimura, F., van Houten, F.J.A.M, Westkämper, E., Shpitalni, M., Ceglarek, D., & Lee, J. (2004). Maintenance: Changing Role in Life Cycle Management, *Annals of the CIRP*, 53(2), 643—656.

Thomas, E., Levrat, E., Iung, B., & Monnin, M. (2006). 'Odds algorithm'-based opportunity-triggered preventive maintenance with production policy. *6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, China, 835--840.

Van Houten, F.J.A.M., Tomiyama, T., & Salomons, O.W. (1998). Product modelling for model-based maintenance. *Annals of the CIRP*, 47(1), 123—129.

Wang, H., (2002). A survey of maintenance policies of deteriorating systems. *European Journal of Operational Research*, 139, 469--489.

Wang, P., & Vachtsevanos, G. (1999). Fault Prognosis using dynamic wavelet neural networks. *AAAI Spring Symposium in Equipment Maintenance Service and Support*, Polo Alto, USA.

FIGURE'S CAPTION

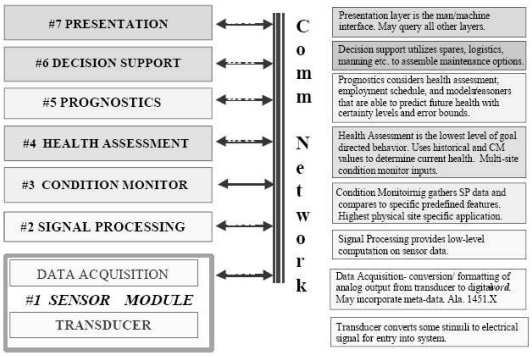

Fig. 1 The architecture OSA-CBM (Lebold *et al.*, 2001)

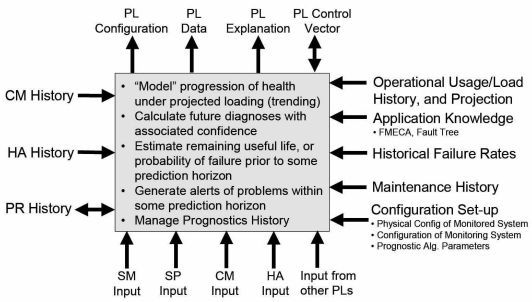

Fig. 2 Input / output of the "Prognosis Layer" from OSA-CBM model.

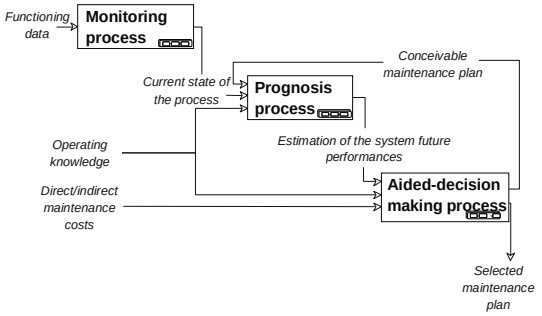

Fig. 3 Integrated system of proactive maintenance (Muller *et al.*, 2008) based on (Léger and Morel, 2001).

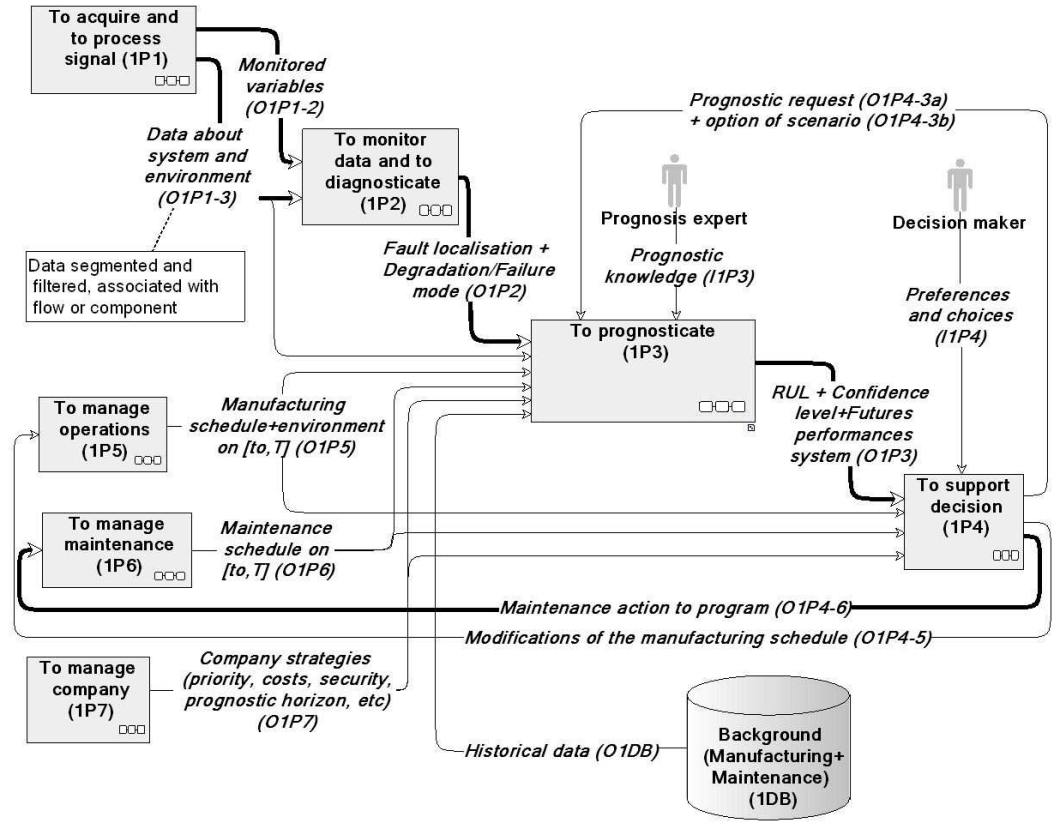

Fig. 4 Environment of the process "To prognosticate"

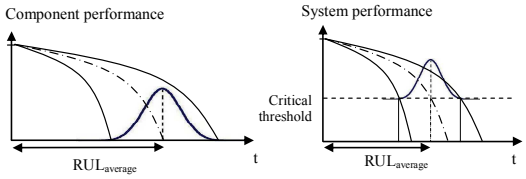

Fig. 5 Component and functional thresholds

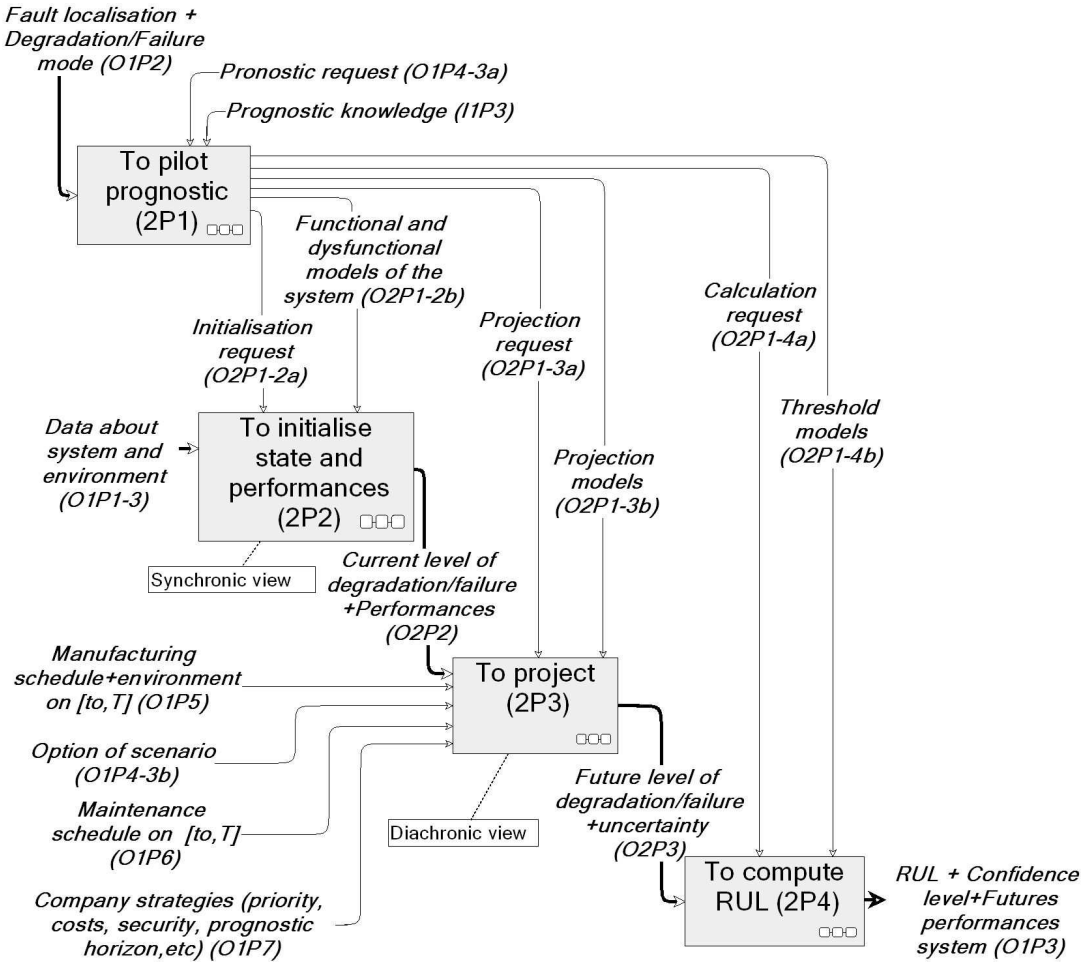

Fig. 6 Sub processes of "To Prognosticate"

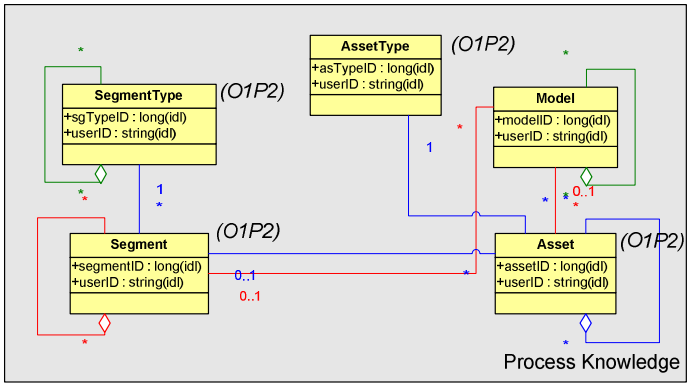

Fig. 7 Process knowledge data

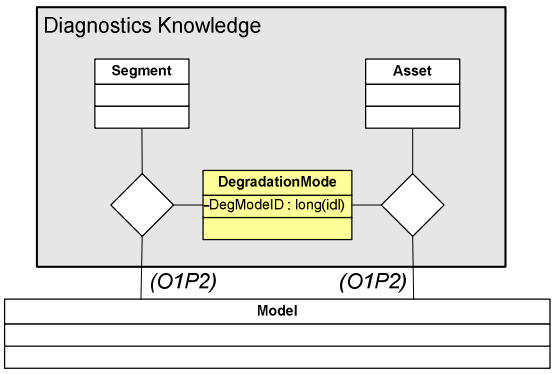

Fig. 8 Diagnostic knowledge

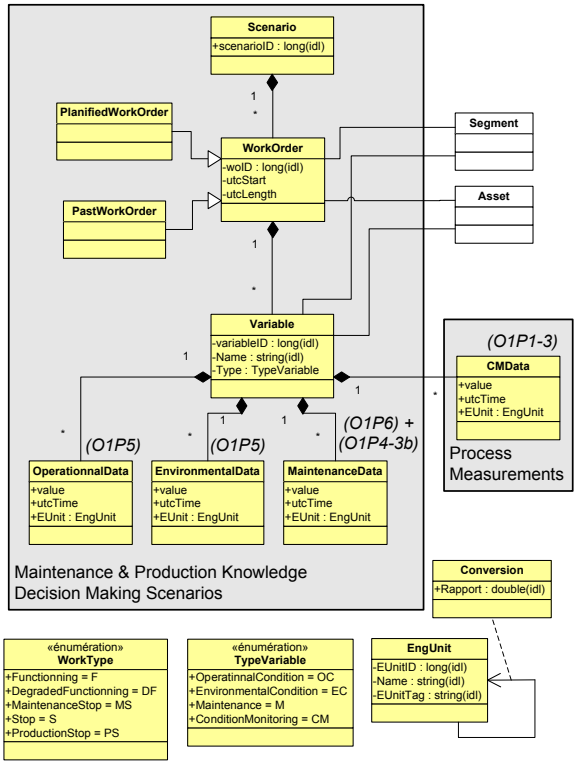

Fig. 9 Maintenance and Production knowledge

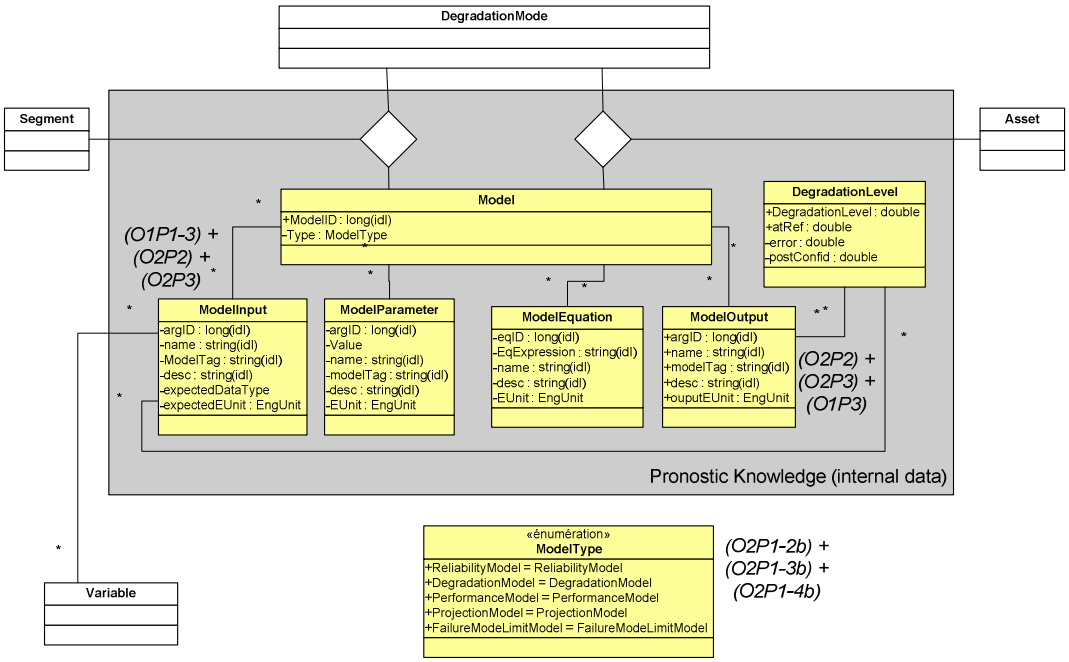

Fig. 10 Prognosis Knowledge

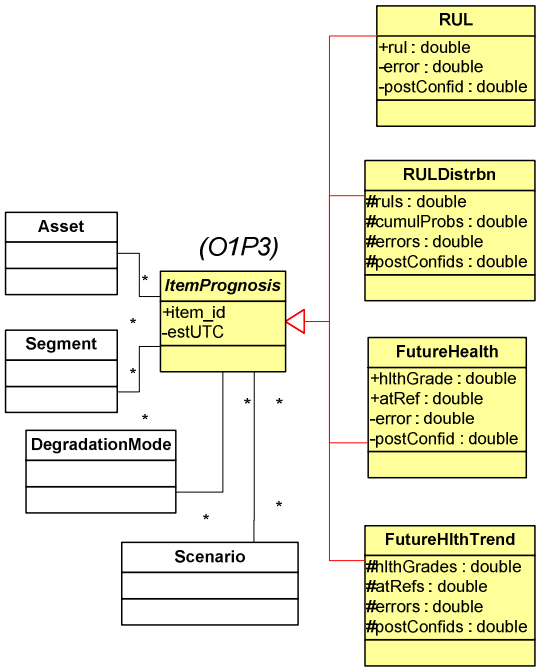

Fig. 11 Item prognosis

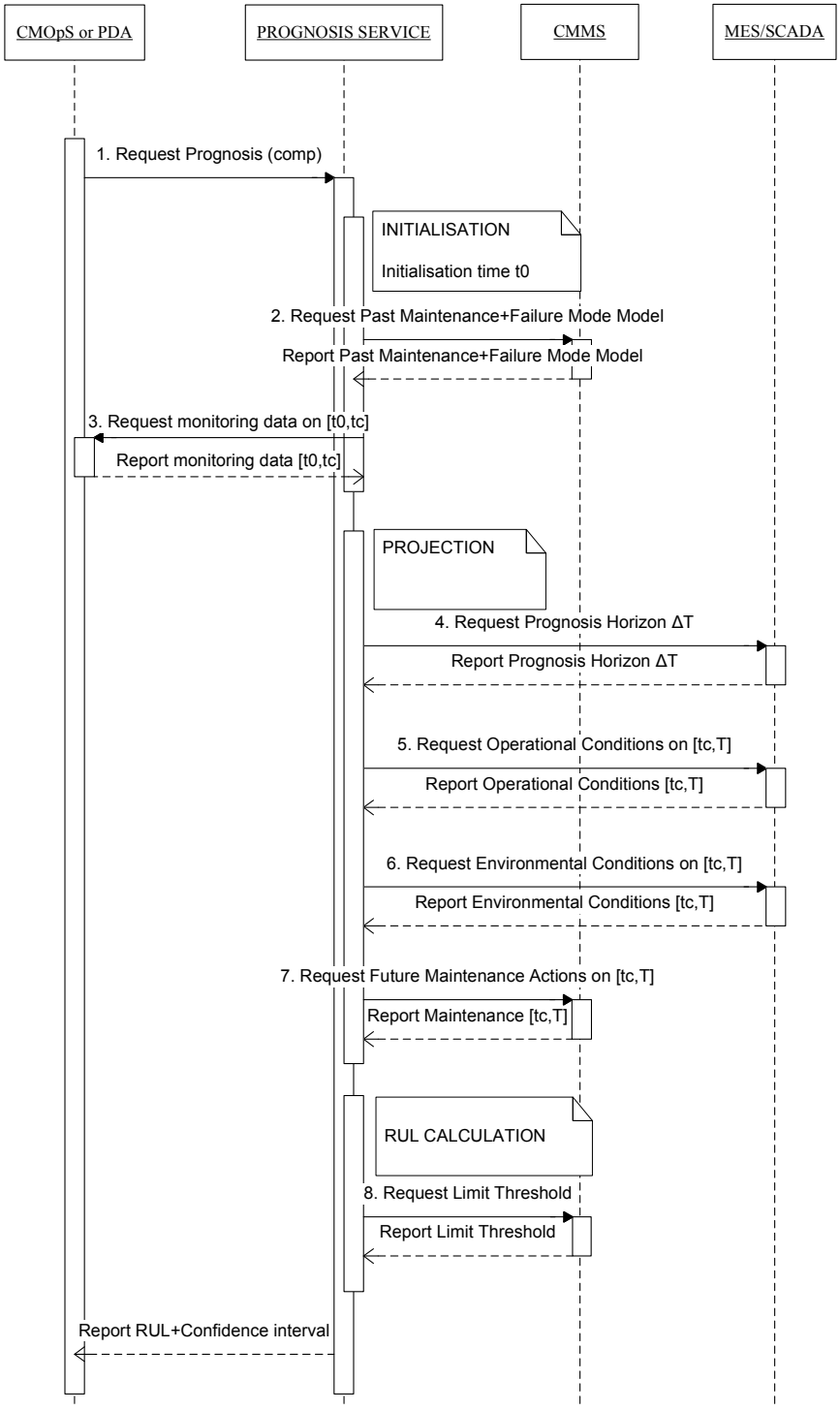

Fig. 12 Sequence Diagram of the prognosis service

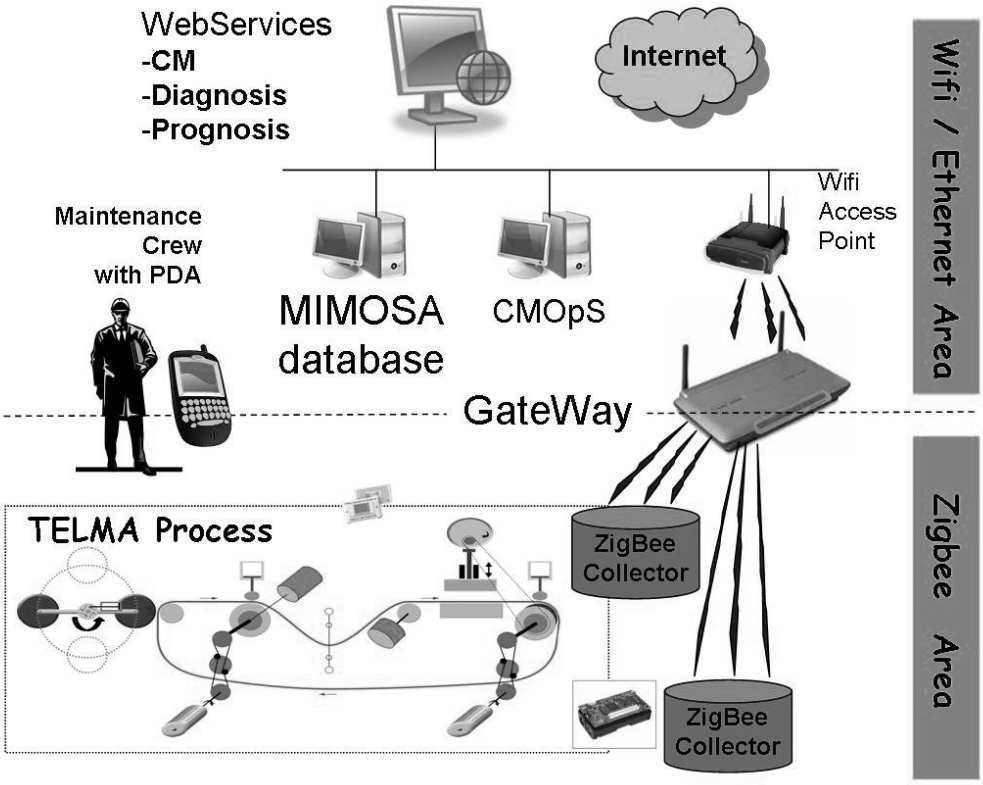

Fig. 13 A first hardware and software lab infrastructure for supporting prognosis.

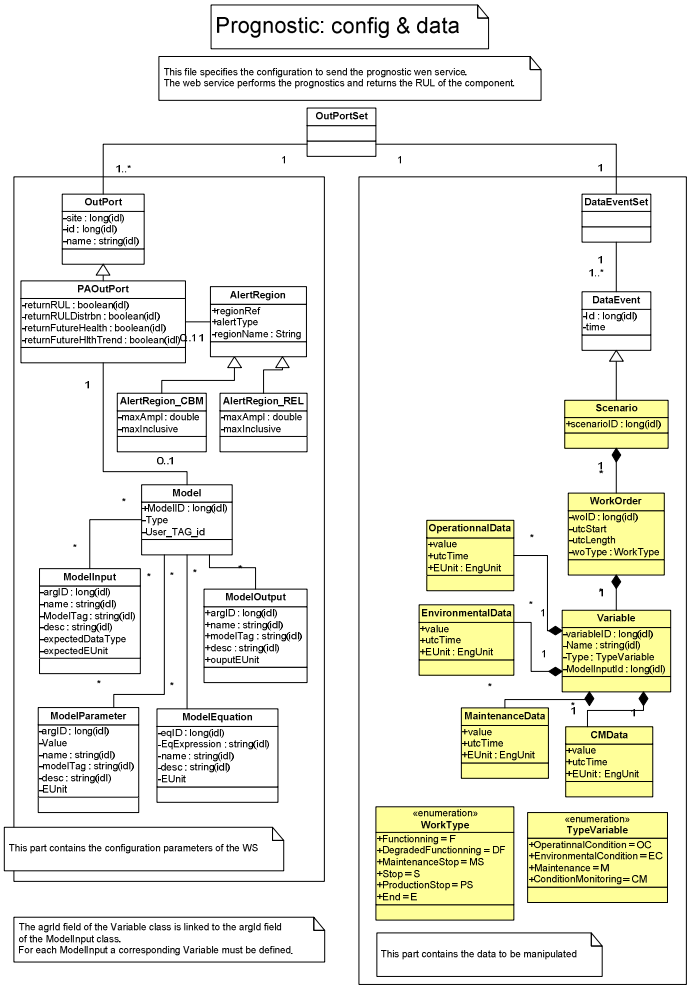

Fig. 14 Information sent to the prognosis Web Service.