

La maintenance prédictive d'un compresseur à gaz :  
Utilisation de plateforme IA et d'API web

# SOMMAIRE

## Table des matières

1) INTRODUCTION .....	1
2) LA COMPREHENSION DU BESOIN CLIENT .....	2
3) ETAT DE L'ART .....	3
<input type="checkbox"/> Les machines tournantes .....	3
<input type="checkbox"/> La maintenance prédictive .....	3
o Le format des données en maintenance prédictive .....	4
<input type="checkbox"/> Thingworx de PTC.....	5
4) LA TRADUCTION TECHNIQUE ET CHOIX TECHNIQUES DU PROJET .....	6
<input type="checkbox"/> Les spécifications techniques de l'USE CASE .....	6
<input type="checkbox"/> Les spécifications techniques de la collecte de données .....	6
<input type="checkbox"/> Les spécifications techniques de la sélection et utilisation d'algorithmes de machine et deep learning .....	8
<input type="checkbox"/> Les spécifications techniques du déploiement .....	9
<input type="checkbox"/> Les spécifications techniques de la validation client .....	9
5) LE BILAN DE PROJET .....	10
<input type="checkbox"/> L'atteinte des objectifs.....	10
6) LES AXES D'AMELIORATION .....	11
7) CONCLUSION .....	12
BIBLIOGRAPHIE.....	

# 1) INTRODUCTION

Le contexte relatif à l'élaboration du projet est la convergence du Big Data et de l'Internet des objets. Ma formation développeur DATA/Intelligence Artificielle de 2 ans a été mise en place dans ce contexte. Au cours de celle-ci, une alternance de 6 mois au sein de SPIE ICS à Nîmes a contribué à l'élaboration de ce projet.

Spie ICS, entreprise de services numériques, est une filiale de Spie France. Elle est spécialisée dans le conseil, l'ingénierie, l'intégration, l'infogérance, la maintenance, les services opérés et le Cloud. Elle compte 3000 employés.

Mes missions au sein de l'entreprise étaient d'utiliser l'intelligence artificielle pour de la maintenance prédictive, tester différentes plateformes IA existantes, établir un système de classement de fichiers.

Ce projet était pour moi l'occasion de mettre en pratique des compétences clés.

Afin de présenter l'utilisation de plateforme IA et d'API web pour la maintenance prédictive d'un compresseur à gaz j'ai divisé mon rapport en 4 parties.

Dans un premier temps, je vais aborder la problématique client et la traduction en objectifs.

Dans un deuxième temps, je vais faire un état de l'art sur les compresseurs à gaz, la maintenance prédictive et algorithmes IA associés, exposer un use case.

Dans un troisième temps, j'aborderai la traduction des spécificités fonctionnelles en spécificités techniques

Dans un quatrième temps, j'établirai un bilan et les axes d'améliorations à apporter au projet.

Enfin, j'effectuerai une conclusion sur la cohérence du travail effectué avec la compétence évaluée et l'apport de mon projet dans la problématique client.

## 2) LA COMPREHENSION DU BESOIN CLIENT

Le client voudrait une solution optimale pour effectuer une maintenance prédictive de son compresseur à gaz.

Il se pose une question, comment choisir un service IA performant et présenter des résultats visuels et interprétables liés à la maintenance prédictive d'un compresseur à gaz ?

Les objectifs formulés par le client sont :

- Comparer les résultats obtenus entre le Random Forest développé par Spie ICS et les services IA fournis sur le cloud.
- Présenter les résultats obtenus par des algorithmes IA à l'aide d'une API web

### 3) ETAT DE L'ART

- **Les machines tournantes**

Par abus de langage le terme compresseur à gaz a été utilisé tout au long de l'alternance. Le terme adéquat est turbine à gaz (ou encore turbine à gaz de combustion). Elle représente une machine tournante thermodynamique appartenant à la famille des moteurs à combustion interne.

Une turbine à gaz utilise le dioxygène de l'air ambiant comme carburant et le comprime. Par combustion du gaz il se produit une augmentation brutale de la pression et du volume de celui-ci mettant en rotation rapide la turbine (1).

Les éléments constituant une turbine à gaz sont : l'entrée, le compresseur, la chambre de combustion, la turbine, la tuyère.

- **La maintenance prédictive**

La maintenance prédictive utilise des données historiques et effectue une surveillance en temps réel du comportement d'un produit.

La maintenance prédictive repose sur 4 technologies clés (figure1).

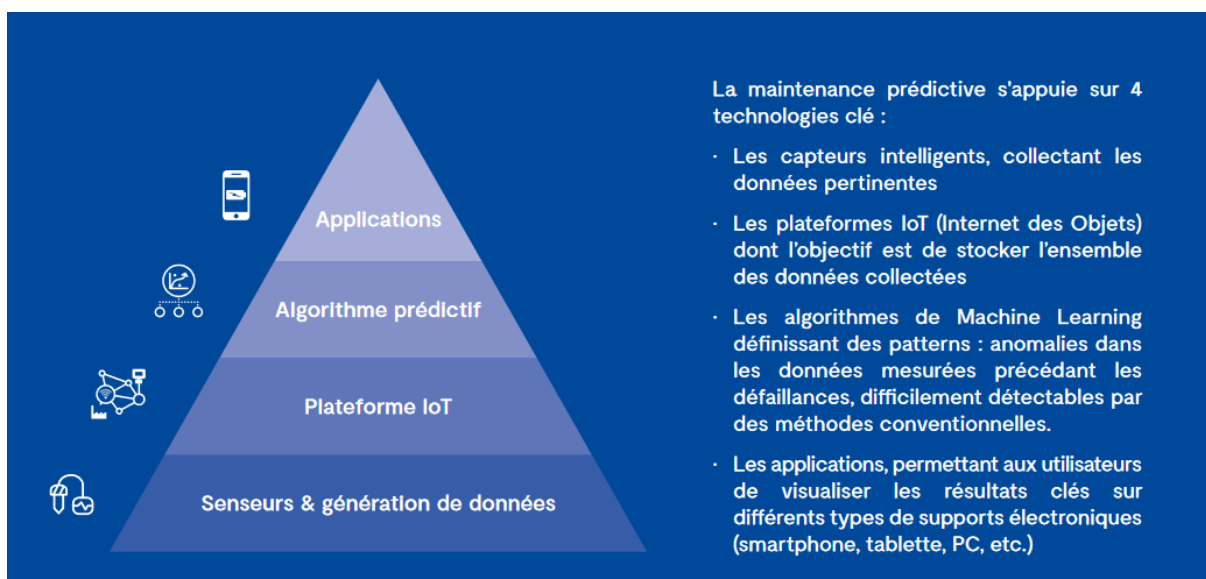


Figure 1 : Les 4 technologies clés de la maintenance prédictive

La maintenance prédictive peut utiliser des algorithmes supervisés ou non supervisés (figure 2). La principale différence est le besoin d'informations sur les pannes précédentes pour effectuer des prédictions en supervisé contrairement au non supervisé.

	Supervised Learning	Unsupervised Learning
Nécessite des données d'incidents passés	Oui	Non
Capacité à prédire de nouveaux types d'incidents	Non	Oui
Difficulté de développement	Modéré	Elevé
Optimisation et maintenance des algorithmes	Elevé	Bas

Figure 2 : Comparaison des principales caractéristiques des algorithmes de machine learning supervisés et non supervisés (2)

### ○ Le format des données en maintenance prédictive

La labélisation des données d'entraînement est effectuée selon le protocole suivant (figure 3) (3) :

- Pour un modèle de régression la cible (target en anglais) est le nombre de cycles restant avant la panne.
- Pour un modèle de classification binaire, une labélisation 0 ou 1 sur un critère défini à l'avance : une plage temporelle avant panne choisie selon les besoins du client afin de répondre à la question : la machine va-t-elle tomber en panne dans X temps ?
- Pour un modèle de classification multiclassées, une labélisation 0 à n sur des critères définis à l'avance : des plages temporelles avant panne choisies selon les besoins du client.

id	cycle	...	RUL	label1	label2
1	1	1	191	0	0
1	2		190	0	0
1	3		189	0	0
1	4		188	0	0
...	...	...	...	...	...
1	160		32	0	0
1	161		31	0	0
1	162		30	1	1
1	163		29	1	1
1	164		28	1	1
1	165		27	1	1
1	166		26	1	1
1	167		25	1	1
1	168		24	1	1
1	169		23	1	1
1	170		22	1	1
1	171		21	1	1
1	172		20	1	1
1	173		19	1	1
1	174		18	1	1
1	175		17	1	1
1	176		16	1	1
1	177		15	1	2
1	178		14	1	2
1	179		13	1	2
1	180		12	1	2
1	181		11	1	2
1	182		10	1	2
1	183		9	1	2
1	184		8	1	2
1	185		7	1	2
1	186		6	1	2
1	187		5	1	2
1	188		4	1	2
1	189		3	1	2
1	190		2	1	2
1	191		1	1	2
1	192		0	1	2

**w0 = 15**  
**w1 = 30**

**w1**

**w0**

- Rul : Remaining useful life (temps restant avant la panne).

- Cycle : pas de temps (heure, minute, seconde ...).

- Label 1 : labélisation en classification binaire.

- Rul : Remaining useful life (temps restant avant la panne).

- Cycle : pas de temps (heure, minute, seconde ...).

- Label 1 : labélisation en classification binaire.

- Label 2 : Labélisation en classification multiclassées.

- W0 et W1 : plages temporelles.

Figure 3 : labélisation des données en maintenance prédictive pour des algorithmes de classification et régression

- **Thingworx de PTC**

C'est une Plateforme as a Service (Paas) offrant de nombreuses fonctionnalités IOT (connexion, développement, analyse, gestion et expérience). Elle est conçue pour les applications industrielles et propose de nombreux outils (Vuforia Studio, ThingWorx Manufacturing Apps, ThingWorx Navigate, Thingworx analytics).

Thingworx analytics est un outil d'analyse de données IOT. Il gère une masse importante de données. Ses fonctionnalités principales sont (4) :

- L'évaluation des anomalies liées aux flux de données
- L'identification des facteurs contribuant à un résultat
- Le maintien automatique de l'intelligence prédictive

Thingworx Analytics est disponible en version évaluation pendant 30 jours.

## 4) LA TRADUCTION TECHNIQUE ET CHOIX TECHNIQUES DU PROJET

A partir des objectifs client et personnels j'ai établi un workflow en 5 étapes (figure 4).

Pour chaque étape j'ai effectué une traduction des spécifications fonctionnelles en spécifications techniques (démarches et procédés incluant les technologies).

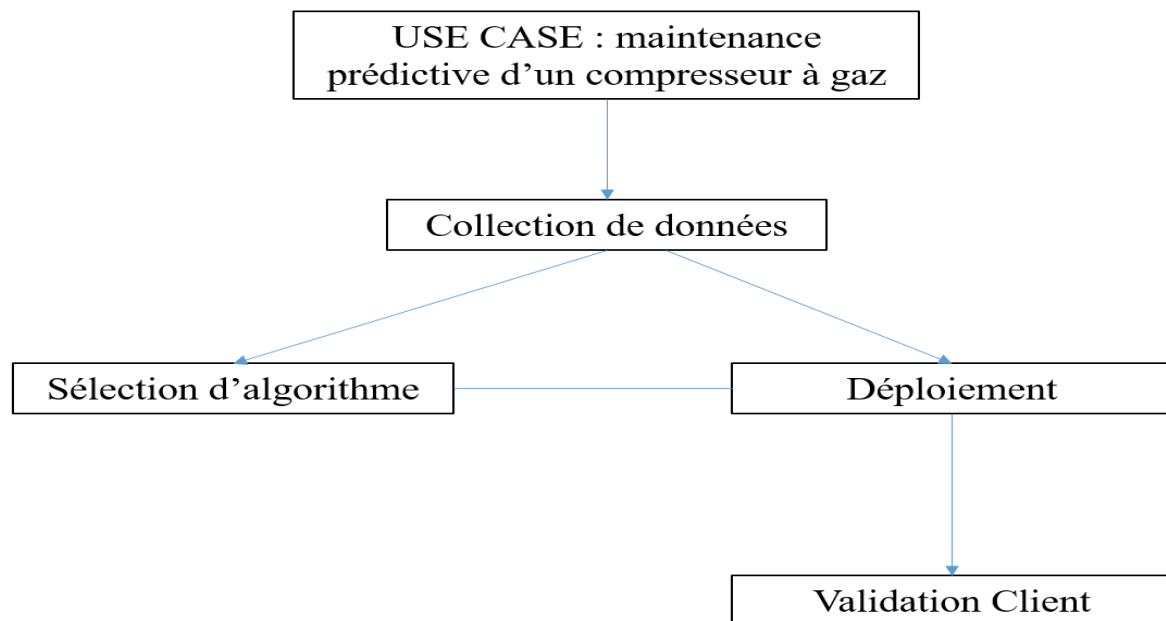


Figure 4 : Workflow du projet

- **Les spécifications techniques de l'USE CASE**

La première étape consiste à définir le cadre dans lequel s'inscrit la problématique client.

- **Les spécifications techniques de la collecte de données**

La deuxième étape consiste à collecter les données afin de pouvoir les explorer et les manipuler pour fournir des résultats chiffrés permettant de répondre aux attentes du client.

Les données sont enregistrées par des capteurs machine et fournies par le client sous forme de fichiers csv (données historiques). J'ai donc décidé de créer une base de données (BDD) en local puis de la stocker sur un bucket dans l'objectif de présentation des données sous forme de dashboard dynamique. La BDD nouvellement créée a été visualisée avec MySQL Workbench.

Par ailleurs, les collections peuvent être utiles dans l'optique d'améliorer le projet. En effet, elles sont très utilisées dans le « Big Data » et l'IOT (Internet of Things), pour la collecte de données transportées. De plus, avec l'augmentation de la taille des données récoltées par capteurs, le Cloud ou une



plateforme personnelle peuvent rapidement devenir une solution de stockage. Enfin, la base de données permettrait de traiter le temps réel.

L'importation des données sur Thingworx Analytics a été réalisée à partir du jeu d'entraînement du Random Forest en régression préparé dans le projet chef d'œuvre.

Un CSV a été créé à partir d'un dataframe python avec la fonction `to_csv()` puis importé sur thingworx grâce à l'utilisation de l'onglet « Data » dans le module « Analytics Builder ».

- **Les spécifications techniques de la sélection et utilisation d'algorithmes de machine et deep learning**

La troisième étape consiste à utiliser l'algorithme de prédiction de Thingworx. Celui-ci a été choisi puisque Spie est client de PTC (distributeur de ThingWorx) et surtout parce qu'il est adapté aux problèmes de maintenance prédictive (voir l'état de l'art).

Pour utiliser l'algorithme il faut aller dans l'onglet « Models » dans le module « Analytics Builder » de Thingworx.

Dans l'onglet advanced model configuration, configurez de la manière suivante afin de coller le plus possible au modèle ayant permis de faire les prédictions du compresseur à gaz du projet chef d'œuvre (figure 5). D'après l'image, la proportion de données allant dans le dataframe de validation est de 20%, les méthodes d'IA ensemblistes comme le Random Forest font des prédictions à partir de la moyenne des estimations obtenues, LE RMSE est utilisé comme métrique d'évaluation, plusieurs techniques et algorithmes IA sont utilisés pour créer un modèle : les arbres de décision, le boosting, la régression linéaire et les réseaux de neurones.

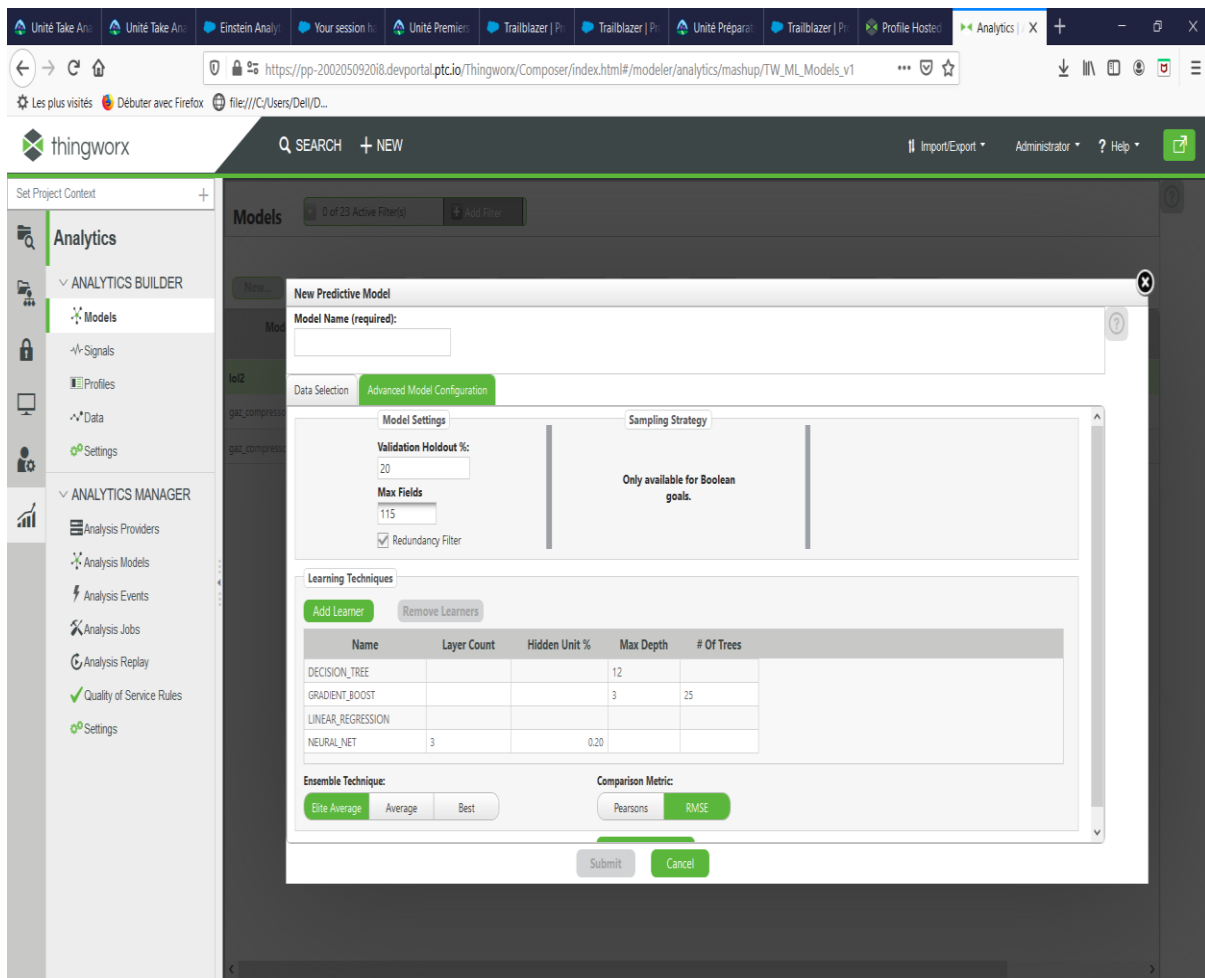


Figure 5 : Configuration du module IA de thingworx analytics

Dans l'onglet « Predictive scoring » de l'onglet « view » il est possible d'ajouter de nouvelles données au modèle IA entraîné. Il est possible, par exemple d'ajouter les derniers enregistrements d'un compresseur dont on cherche le temps de survenue de la panne.

Ainsi un csv des résultats du modèle IA avec les enregistrements d'intérêt sont obtenus.

Il est possible de faire des prédictions sur des séries temporelles.

## • Les spécifications techniques du déploiement

La 4<sup>ème</sup> étape consiste au déploiement de bases de données et/ou d'algorithmes de machine learning. Je ne travaille pas sur un projet de BDD, la seule chose que j'ai déployée c'est l'algorithme de machine learning afin de passer du développement à la production et fournir un webservice au client. Pour cela j'ai utilisé streamlit, un framework de machine learning permettant de créer une application web. J'ai ainsi pu présenter les différents résultats permettant de répondre aux objectifs client.

L'API Streamlit utilise plusieurs fonctions afin de pouvoir créer différents widgets (recherche, affichage de données, de graphiques ...).

Ensuite, j'ai utilisé docker sous linux pour créer un container pour une application scalable et portable (libérée des contraintes de systèmes d'exploitation).

Enfin, j'ai utilisé Heroku pour déployer mon application et permettre au client de l'utiliser en toute autonomie.

## • Les spécifications techniques de la validation client

La dernière étape consiste à faire valider le travail par le client.

Cela a été fait par comparaison des métriques des modèles IA utilisés dans le projet chef d'œuvre et des métriques de résultat d'utilisation du service web.

Les métriques établissent la fiabilité des prédictions dans un contexte donné.

La présentation des principaux résultats obtenus par IA avec l'utilisation d'un API web a aussi été réalisée auprès du client.

## 5) LE BILAN DE PROJET

### • L'atteinte des objectifs

Les objectifs clients ont tous été atteints. En effet :

- Les résultats obtenus entre l'algorithmes déjà utilisé et les services IA fournis sur le cloud ont été comparés. Les prédictions de RUL ont été effectuées. Les deltas entre les prédictions et vrais valeurs sont similaires pour le service IA thingworx et le Random Forest de Spie ICS (figure 6).

Date	rul (Original Value)	rul	rul_mo	Delta(Thingworx)	Delta(Random Forest)
08/01/2019 00:12	765	703.6829268292682	0.1455393850732716	62 min	40 min
10/01/2019 11:57	567	781.2173913043479	0.16157546872892406	3h34 min	3 h
12/01/2019 12:25	413	544.952380952381	0.11270990298911707	2h11 min	2h30 min
19/01/2019 18:07	1177	1123.5	0.23236814891416752	54 min	10h
25/01/2019 13:17	760	984.1923076923077	0.20355580303873996	3h44 min	3h10 min
06/02/2019 17:08	498	777.8692307692307	0.16088298464720388	4h 39 min	4h30 min
27/02/2019 19:06	551	781.2173913043479	0.16157546872892406	3h50 min	3h10 min

Figure 6: comparaison des deltas entre prédictions et vraies valeurs obtenues avec le service IA Thingworx et le Random Forest de Spie ICS

- Les résultats obtenus par le Random Forest en maintenance prédictive élaboré par Spie ICS ont été présentés au client à l'aide d'une API web (figure 7). On peut y distinguer plusieurs widgets. Un widget de recherche de fichier csv en local (stockage données remontées par capteurs). Plusieurs widgets d'affichage pour la visualisation des RULs, des dates d'intérêt à partir desquelles ont été effectués les prédictions, du RMSE.

Enter a file path:

File not found.

☐ Show dataframe

temps restant avant panne (en min) à partir des dates intéressées:

	0
0	784.5089
1	2,841.2967
2	1,991.0481
3	790.8087

RMSE: 466.0896846761825

interested\_dates:

	Date
174	Nov 19, 2018 9:06 AM
174	Dec 1, 2018 7:10 AM
174	Dec 5, 2018 11:43 AM
174	Dec 15, 2018 10:03 AM

Figure 7 : Présentation web des résultats obtenus avec le Random Forest de Spie ICS (paramétrage non optimisé)

## 6) LES AXES D'AMELIORATION

Thingworxs propose aussi un service de présentation de données (« mashups »). Ce dernier peut utiliser les données de Thingworx pour les présenter sur le web grâce à un API REST ([14](#)).

Cet API permet d'interroger un serveur à partir d'un URL. Il utilise plusieurs méthodes : GET, POST, PULL et DELETE.

Pour créer un « mashup », il faut d'abord créer un projet avec Thingworx. Cette réalisation suit une charte de construction. Elle doit contenir des classes hiérarchisées avec propriétés (« Thing templates ») contenant en bout de chaîne les objets (« Things ») avec leur valeur.

« Things template » est un modèle et « thing » est une instance d'objet.

La documentation sur la création d'un « mashup » au sein de Spie ICS est manquante mais en cours de construction à l'heure actuelle.

## 7) CONCLUSION

A partir de objectifs client j'ai comparé plusieurs modèles IA dont l'un était fourni par une plateforme PaaS proposant un service IA. J'ai aussi utilisé un API afin de présenter des résultats obtenus par intelligence artificielle.

Les résultats obtenus ont été satisfaisants pour le client.

Le client voudrait maintenant pouvoir visualiser des résultats d'IA obtenus par la création de Mashups Thingworx.

Le travail effectué répond à la problématique client de déterminer le meilleur modèle IA à sa disposition et de présenter les résultats obtenus sur le web. Ce service web, lui permet en effet d'obtenir des informations visuelles, interprétables.

# BIBLIOGRAPHIE

- (1) [https://fr.wikipedia.org/wiki/Turbine\\_%C3%A0\\_gaz](https://fr.wikipedia.org/wiki/Turbine_%C3%A0_gaz)
- (2) <https://www.mobility-work.com/fr/blog/maintenance-predictive-vs-maintenance-preventive-strategie-entreprise>
- (3) <https://gallery.azure.ai/Experiment/a677f8eececf40eaa158699a2b27e3c8>
- (4) <https://developer.thingworx.com/resources/guides/thingworx-rest-api-quickstart>
- (5) <https://www.4cad.fr/iot/ptc-thingworx-analytics>