DOI: 10.2507/29th.daaam.proceedings.083

DIGITAL TWIN APPLICATIONS: DIAGNOSTICS, OPTIMISATION AND PREDICTION

Dmitry Kostenko, Nikita Kudryashov, Michael Maystrishin,
Vadim Onufriev, Vyacheslav Potekhin, Alexey Vasiliev

This Publication has to be referred as: Kostenko, D[mitry]; Kudryashov, N[ikita]; Maystrishin, M[ihail]; Onufriev, V[adim]; Potekhin, V[yacheslav] & Vasiliev, A[lexey] (2018). Digital Twin Applications: Diagnostics, Optimisation and Prediction, Proceedings of the 29th DAAAM International Symposium, pp.0574-0581, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-20-4, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/29th.daaam.proceedings.083

Abstract

The paper shows a relevance of digital twin's usage and also their main functions are listed. The authors show requirements for digital twins' infrastructure needed to supply its functionality. Then paper describes digital twins' applications in static and dynamic diagnostics tasks' solution. A knowledge base management system usage is shown for static diagnostics task solution. An execution environment usage is shown as a mean for dynamic diagnostics task solution. Also authors show digital twin's application for the technical process optimisation task solution, in this case Pareto efficiency can be used as a criterion. Then the digital twin's complete functional scheme is described. In conclusion the perspective of multi-agent approach usage is shown.

Keywords: Industry 4.0; digital twin; intelligent control system; automation; global digitalisation

1. Introduction

Industry 4.0 is a worldwide trend and a strategy of industrial production management based on global digitalization, data transfer, the prevalence of cyber-physical systems, Internet of things, cloud computing and artificial intelligence [1]. The core part of the Industry 4.0 strategy is the way the object is being represented as an informational model, also called a Digital model. So, the digital twin is an object-oriented digital model of a physical asset that comprises all the data about the asset, including system, structural, engineering and technological models.

Digital twin conception is rather new so different researches understand it in different ways. One question is if the control system is part of Digital twin or it is not included into? Answering this question influences its functions and thus its structure. So, the authors of [2] consider digital twin as rather visualization system than control system. We suggest including control system into Digital twin. This mean it will also include some statistical data, a knowledge base and a set of control agents. Then the most important scientific problem is connected with difficulties of effective control over large-scale distributed systems (manufactures) under conditions of uncertainty. The continued sophistication of systems and structures leads to uncertainty and as a result, poor formalizability of an object and a problem of control.

The impossibility of an accurate prediction of structurally complex network-based structures (systems) behaviour raises the question about the development of a new paradigm of a complex systems control; even more so for systems operated within environments of extensive uncertainties and unpredictable behaviour. The problem has inspired the scientific society to look for new effective approaches, methods, and models to control exceptionally complex network structures, such as manufacturing systems.

It includes：
- heterogenic sources and means of information consumption (means of measurement, gathering and processing data, as well as executive elements and decision-making structures);
- structurally complex hierarchy of master, interworking and slave control organs, standalone functionaries, etc.;
- various types, forms and kinds of information circulating in the system;
- various methodologies, models, and means of information processing.

A comparative analysis of goals and objectives of a Digital twin has been summed up in [3]. The digital twin must support a variety of different functions：

1. Asset analysis and status prediction:
   1.1. System anomalies monitoring,
   1.2. Monitoring of different deformations of asset material,
   1.3. Asset reliability estimation.
2. Digital reflection of an asset life cycle:
   2.1. Long term research of a system behaviour and performance prediction under the influence of the outside environment,
   2.2. Asset life cycle management. Maintaining an uninterrupted data flow on various stages of an asset life cycle,
   2.3. Virtual launch of an asset.
3. Asset management:
   3.1. Optimal asset management,
   3.2. Cooperation with other assets for an optimal factory management.
4. Help with decision-making based on engineering and statistical analysis:
   4.1. System optimisation during the engineering stage,
   4.2. Life cycle optimisation based on past and future asset condition, and prediction of a future condition.

With a wide range of different equipment and processes the goal of creating a digital twin is undeniably complex and requires a systematic approach based on modern standards of business control system creation.The goal of this paper is to suggest the digital twin's structure, providing a solution of the following tasks: static diagnostics, dynamic diagnostics, identification and optimisation. We will form the Digital twin's structure step by step considering list of complex tasks, where suggested elements will be described.

Finally, digital twin should provide:
- An ability to expand processes hierarchy.
- An ability to create different (both universal and specialized) interfaces for different levels of hierarchy.
- A scalability — an ability to increase a model's complexity by both hierarchy levels addition, and adding more details to every stage of a process.
- A flexibility — an ability to represent different types of industrial processes using the same display format;
- An ability to view different scales of visualization with different amounts of hierarchy levels and a varied amount of details on every stage of any process.
- A semantical model that structures all asset-related information and clearly describes the control logic for various types of a manufacturing processes;
- A framework that connects different modelling methods and cross-disciplinary digital models of an asset.
- A communication layer to maintain an uninterrupted data exchange between a physical object and a digital twin.
  In this article we will consider tasks, which can be solved using suggested structure of an industrial digital twin.

2. Diagnostics task solution using known models

Consider the task of production state diagnostics. This task means checking if control plant's performance indicators meet all the requirements. So we can differ static diagnostics, when all the key performance indicators are given and are stored as constants during some time, and dynamic diagnostics, when the indicators are unknown and we need to compute them before control.

2.1. Static diagnostics task

Predetermined presets based control is implemented via synergetic system, which includes a control application, a knowledge base and an OPC-server.

The first step is input data reading. Input data consists of parameters that characterise controlled system's behaviour, such as pressure, temperature or fuel consumption. Input data from sensors, IIoT (Industry Internet of Things) devices [4], etc. is being sent to the OPC-server. The second step is presets reading. Presets are the arrays of master values, marking the borderline parameters of the system. Just like the input data, presets can be read from a remote node or stored on an OPC-server, thus allowing the operator to edit them from the operator's terminal.

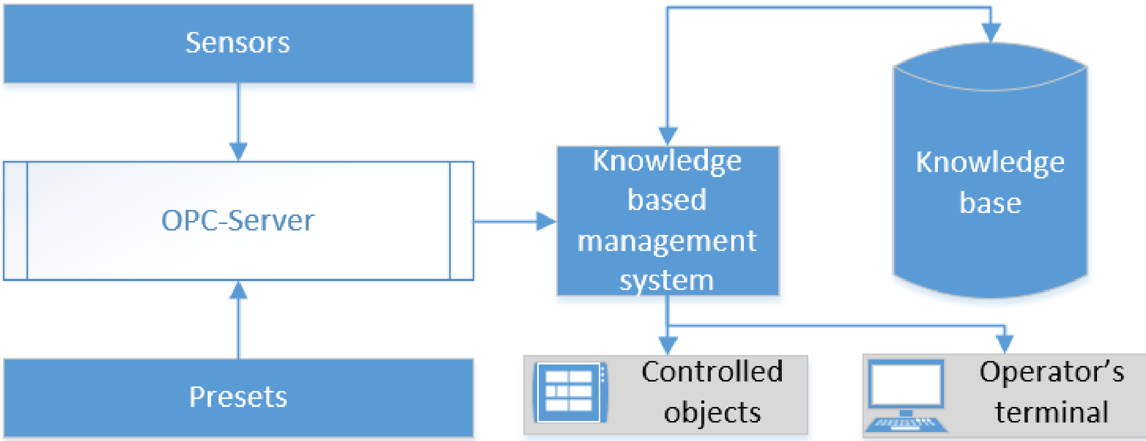

Fig. 1. The scheme of static diagnostics task solution

The third step is accessing the knowledge base. The knowledge base contains a set of rules, based on mathematical models and empirical knowledge. Connections between the nodes of the knowledge base allow the knowledge based management system to produce controlling signals. An excerpt from the knowledge base source code written using Resource description framework (RDF) [5] is listed below.

```
@prefix : <http://this.storage/elements/> .

:cleanPump    a    :Solution .
:checkMeasurementCircuitFqip1001    a    :Solution .
:malfunctionMeasurementCircuitFqip1001    a    :Malfunction .
:malfunctionPipeLine   a    :Malfunction .
:malfunctionMeasurementCircuitFqip1001    :uses_freq    6 .
:malfunctionPipeLine   :uses_freq  9 .
```

Listing 1. Part of Knowledge base source code

For instance, knowledge base entries for an oil processing plant raw oil consumption can be presented as a set of IF-THEN statements:

- if $Q\_current < Q\_set\_sis\_low$, then emergency alarm;
- if $Q\_set\_sis\_low \le Q\_current \le Q\_set\_normal\_min$, then yellow alarm code;
- if $Q\_set\_normal\_min < Q\_current < Q\_set\_normal\_max$, then green alarm code;
- if $Q\_set\_normal\_max \le Q\_current \le Q\_set\_alarm\_high$, then yellow alarm code;
- if $Q\_current > Q\_set\_alarm\_high$, then red alarm code

Q_current stands for current raw oil consumption value from a sensor, and *Q_set_normal_min*, *Q_set_normal_max* and *Q_set_alarm_high* are the presets.

Replacing variables with values gives us a set of rules which describe the system's various gradations of reaction:

- if oil consumption $<$ 366.3 m3/h, then emergency alarm;
- if 366.3 m3/h $\le$ oil consumption $\le$ 373.7 m3/h, then yellow alarm code;
- if 373.7 m3/h $<$ oil consumption $<$ 673.2 m3/h, then green alarm code;
- if 673.2 m3/h $\le$ oil consumption $\le$ 686.8 m3/h, then yellow alarm code;
- if oil consumption $>$ 686.8 m3/h, then red alarm code

The last stage of the scenario is sending a query to the knowledge base in order to localize possible system malfunctions, which could cause these problems. The knowledge base not only stores borderline values, it also keeps track of empirical data on causes and consequences of various system faults. The knowledge base saves and marks each malfunction, preserving the most frequent fault descriptions. The knowledge base can thus be categorized as an instance of a self-learning system designed to work with incomplete and inaccurate data [6]. A partial graphical representation of the scheme made in Cmap Tools [7], used to build the knowledge base can be seen below:

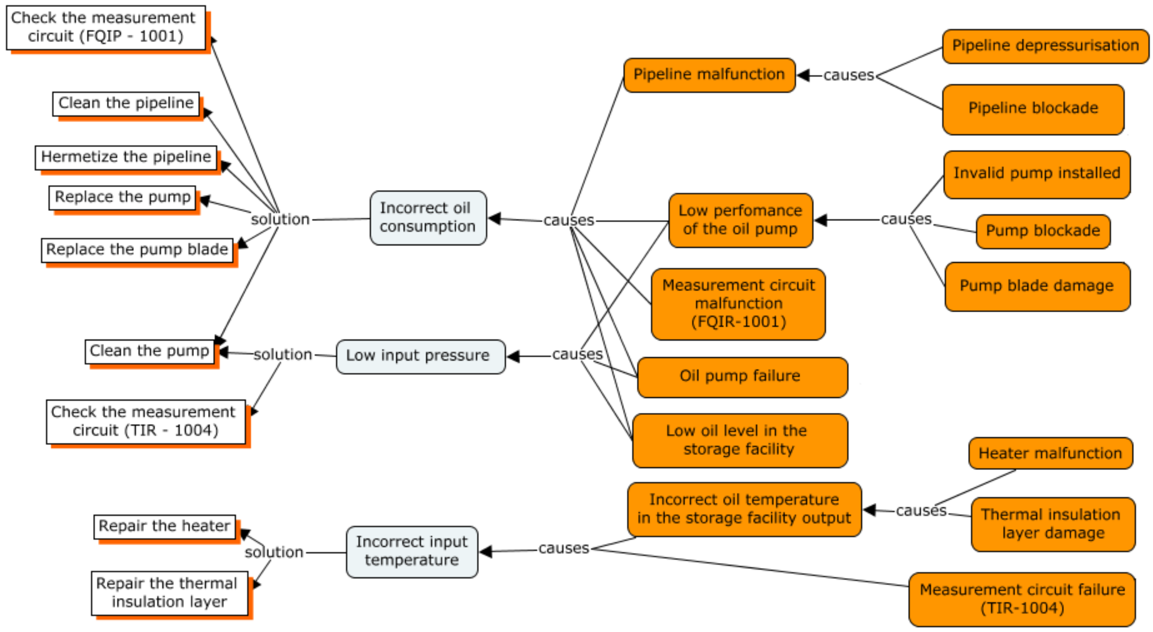

Fig. 2. Concept map of the knowledge base

With this information the system is not limited to make emergency-case decisions on object control: it can give an operator a list of the most probable malfunctions and recommend how to fix them. Furthermore, the knowledge base management system can collect statistical data about actual reasons of malfunctions and can thus be trained.

2.2. Dynamic diagnostics

Dynamic diagnostics mode is run in situation of production task changing or equipment modernization of source materials changing, when it happens during a control time. In all these cases we cannot a priori possess knowledge about desired performance indicators, because we do not know how the task, the equipment and the source material will change. That is why corresponding key performance indicators should be computed. In order to do it, we offer to use cooperative work of following elements, which we then need to include into digital twins: digital model, execution environment, Database management system and control system block.

In order to get data from control plant we can use distributed control system (DCS) or improved control system (ICS). They are directly connected to the control plant, so they can send control signals to it. We also presume that a protective safety system (PSS) is also placed there. In order to calculate key performance indicators (KPI) in situation of task, source material or equipment changing we propose to use compositional program-analytic models (CPAM), which are models of different level KPIs connection. They are actually expressions, explaining how on KPI depends on another one or on control parameters. So, data measured by sensors is sent to the database management system (DBMS). Then we should use Execution environment to run digital models (particularly CPAM) in order to compute KPI corresponding to relevant situation. Thus execution environment also needs relevant data from control plant.

Then the execution environment computes the relevant KPI, basing on current data and CPAM. So, reference data is computed and then it should be sent back to DBMS in order to match and to check if current data meets KPI's limitations (fig. 3). It is quite similar with the previous task. After the evaluation and comparison, set points go back to DCS to provide control signals, based on modelling values. At the same time the data from execution environment is displayed with human-machine interface (HMI) to show the difference between set-point and current values.

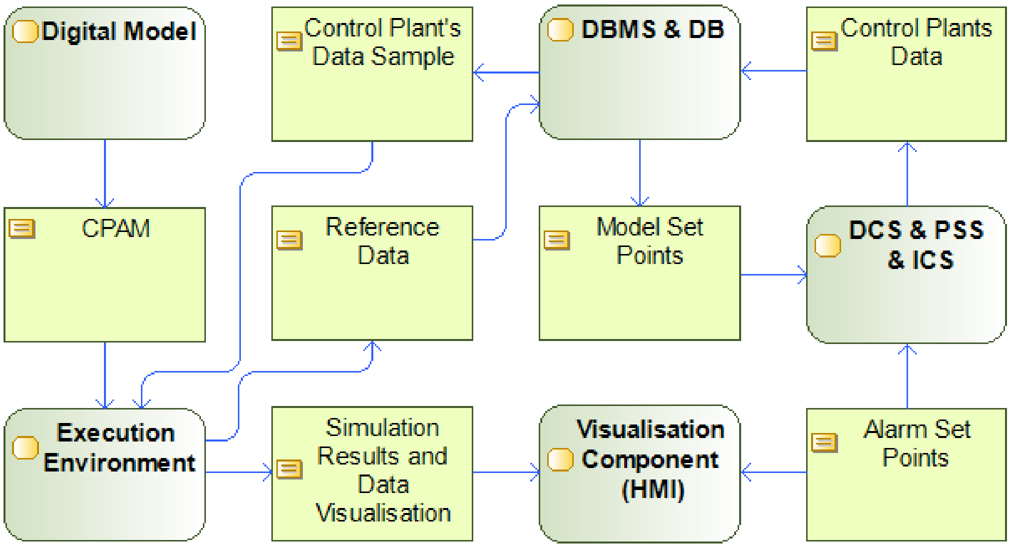

Fig. 3. Functional scheme of system dynamic diagnostics

CPAM can be represented in different ways: formula-based models, neural-network model, analytical models. On the other hand, they can be represented as files of modelling systems such as ChemCAD, MATLAB, MathCAD etc. In this case, execution environment should be able to run these kinds of files.

3. Optimisation and prediction tasks

Sometimes the digital twin does not have the CPAM needed to calculate necessary KPIs. In this case it should be able to compute CPAM basing on statistical data it stores in the database (DB).

Thus, optimisation task solution using the digital twin should be decomposed to two tasks. The first one is the task of getting dependence of functions, being optimised, on control parameters -- getting the CPAM. The second task is optimisation task, which bases on computed CPAM.

The main difficulty of first task solution is caused by hierarchical structure of production processes. So, control parameters and being optimised parameters can belong to different level. Parameters being optimised are often placed in the highest level of hierarchy -- business processes level (for ex. amount of goods delivered) and business indicators (for ex. annual income). Control parameters can be placed both in the highest level (for ex. invested amount) and in lower levels (for ex. controlled temperature during different production processes).

For example, we have the production process, transforming a source material in *State 0* to the *State 2* (final product). The vector u_1, temperature and pressure of steam, is not directly connected with vector KPI_2 (company's income and amount of goods produced). So, the model of their connection cannot be estimated directly. Furthermore, vector u_1 can also be not connected directly with KPI_1 -- liquid viscosity (fig. 4).

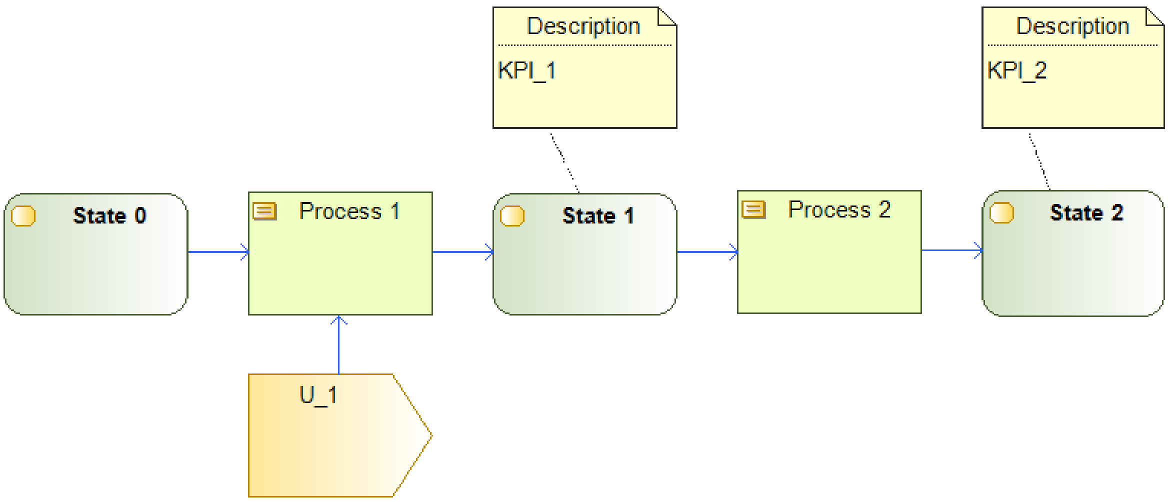

Fig. 4. System model of production

Thus, the task of finding relation between u_1 and KPI_2 is complex cause of subprocesses mixing. So we suggest decomposition of technological processes in order to solve this task. We need to get a hierarchical description of processes tree, so we decompose the Process 1 (fig. 5).

Consider the task of KPI_2 vector parameters optimisation, using u_1 as the vector of control parameters. But this vectors directly connected only with Process1, but is not connected with *Process 2*. KPI_2 is the vector, describing the *State2*, which is result of *Process 2*. Thus, the task of estimating relation KPI_2 = f(u_1) can be complex and mathematically incorrect.

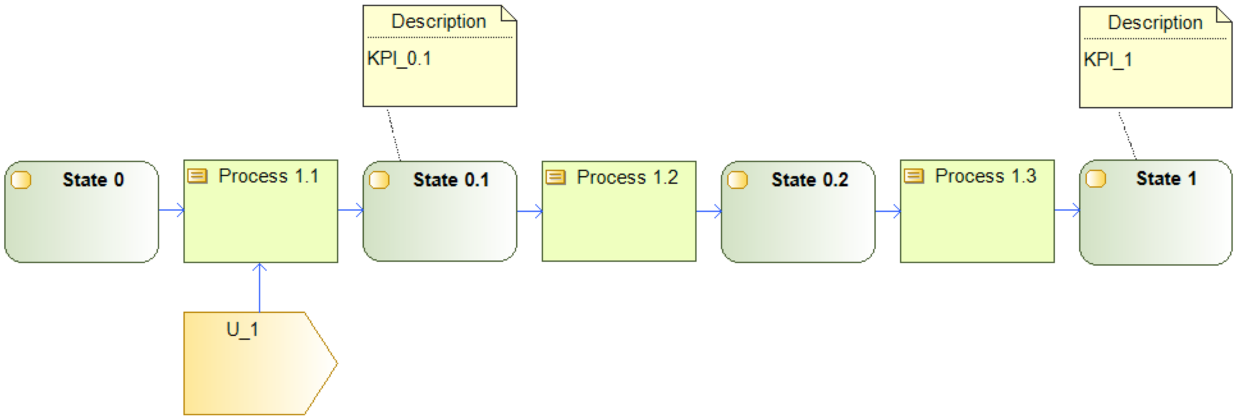

Fig. 5. Decomposed system model of production

The Fig. 5 shows the *Process 1*, decomposed to *Process 1.1*, *Process 1.2* and *Process 1.3* and to two intermediate states: *State 0.1* and *State 0.2*. We also get intermediate KPIs. So, we can see that vector u_1 is directly connected with $KPI_{0.1}$, which is directly connected with KPI_1. Using the same approach, we can connect KPI_1 and KPI_2. So, firstly we can estimate connection $KPI_{0.1}$ = f(u_1), then KPI_1 = f($KPI_{0.1}$), then KPI_2 =f(KPI_1), that can help us to get KPI_2 = f(u_1).

In all these cases the identification process can be lead using, for example, radial bases functions neural networks [8]. To implement this identification we need the historical data, stored in DB. After CPAM is computed it should be saved to digital models storage. After analytical relation of vector KPI_2 and vector u_1 is computed, we can apply any of multiobjective optimisation algorithms, for example, Fonseca and Fleming's Multiobjective Genetic Algorithm, described in [9].

The multiobjective optimisation can be implemented basing on Pareto optimality [10]. The prediction task can be solved using the CPAM. Then the time should be also considered as one of parameters.

4. Functional scheme of the digital twin

So, the digital twin should be able to process current data in order to implement static and dynamic diagnostics, optimisation and prediction tasks. Also it should use digital models of control plants and identify models if they are not given. Thus, the set of model is sent to Execution environment. After the execution reference data is sent to the database, controlled by DBMS. The interaction between main digital tween components (fig. 6) provides such actions.

After the simulation and visualization (if needed in the scenario) the result of such processes visualizes on HMI.

After the execution and storing the reference data into DB, digital tween needs to create the set points, based on modelling data. Such set points, combined with the data in knowledge base, are sent in form of warning / alarm set points and deviation signals to control system in order to provide alarm indications and control. This is also needed to visualize the deviation between current and planned values.

The Integrating IIoT Environment aggregates all the data about the control object. At the same time, it provides the analytics, based on KPI, deviations and prognosis. This data is needed to provide the distributed control.

In the end control systems only need to apply the set point into the current object. Data about it comes from the Information-measuring system. DCS / PSS / ICS provides the real Control signals which comes into control object.

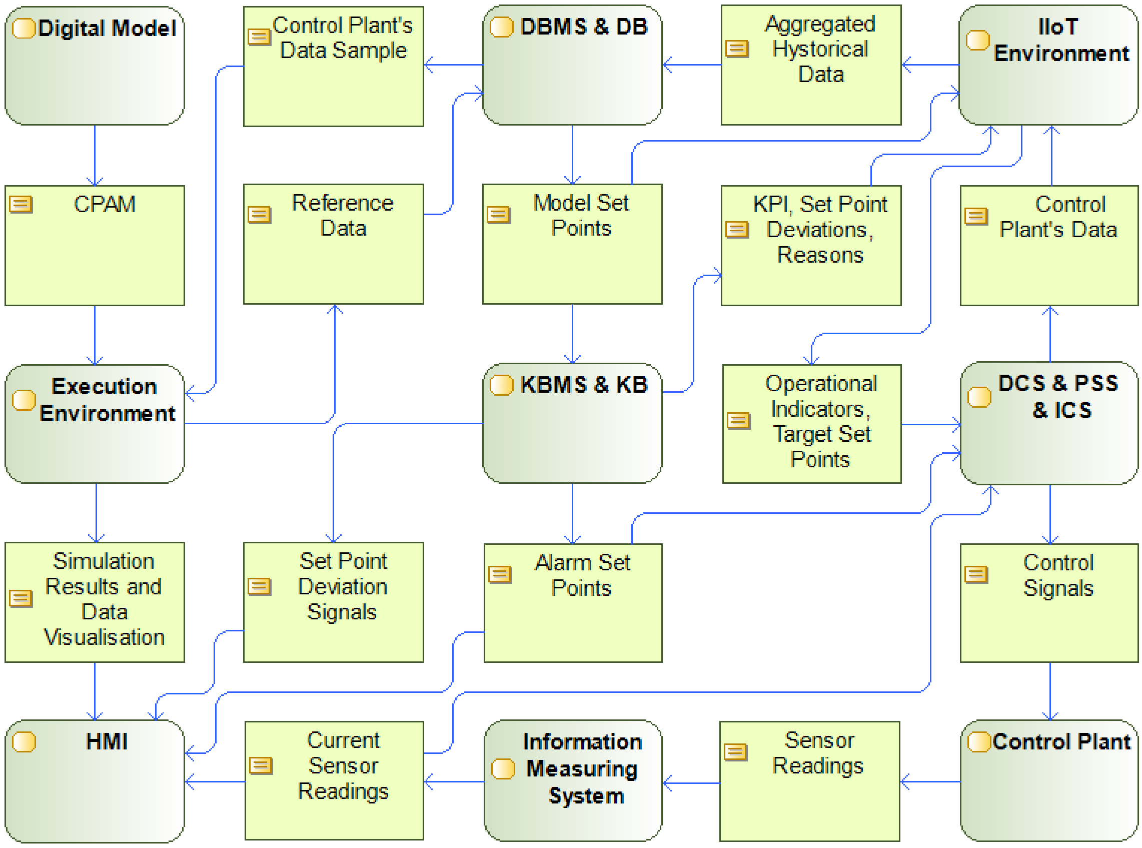

Fig. 6. Functional scheme of a Digital twin

5. Conclusion

So, the proposed structure and functional scheme of the digital twin can provide solution of diagnostics tasks (static and dynamic), optimisation and prediction tasks. Before start operating the digital twin needs:

- the initial digital model, describing the processes and relations between some of control parameters and being optimised parameters;
- the knowledge base, filled with initial information about desired KPI;
- the database, able to store current data from control object;
- the ability to connect to DCS / PSS / ICS to send them a control signals;
- the execution environment, which can run digital models;
- the chosen algorithms of system identification and optimisation.

6. Acknowledgments

The article is published with supporting of the project Erasmus+ 573545-EPP-1-2016-1-DE-EPPKA2-CBHE-JP Applied curricula in space exploration and intelligent robotic systems (APPLE) and describes the part of the project conducted by SPbPU.

7. References

[1] Ponomarev, K.; Kudryashov, N. & Popelnukha, N. (2017). Main Principals and Issues of Digital Twin Development for Complex Technological Processes, Proceedings of the 28th DAAAM International Symposium, pp.0523-0528, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978- 3-902734-11-2, ISSN 1726-9679, Vienna, Austria.

[2] Deac, C. N.; Popa, C. L.; Ghinea, M. & Cotet, C. E. (2017). Machine Vision in Manufacturing Processes and the Digital Twin of Manufacturing Architectures, Proceedings of the 28th DAAAM International Symposium, pp.0733-0736, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria DOI: 10.2507/28th.daaam.proceedings.103.

[3] Negri, E.; Fumagalli, L. & Macchi, M. A review of the roles of Digital Twin in CPS-based production systems, Procedia Manufacturing, Volume 11, 2017, Pages 939-948.
[4] Nikolic, B.; Ignjatic, J.; Suzic, N.; Stevanov, B. & Rikalovic, A. (2017). Predictive Manufacturing Systems in Industry 4.0: Trends, Benefits and Challenges, Proceedings of the 28th DAAAM International Symposium, pp.0796-0802, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria DOI: 10.2507/28th.daaam.proceedings.112.
[5] https://www.w3.org/TR/2014/REC-n-triples-20140225, (2014). RDF 1.1 N-Triples. A line-based syntax for an RDF graph, The World Wide Web Consortium (W3C), Accessed on: 2018-07-14.
[6] Dutta, S.; Bandopadhyay, S.; Ganguli, R. & Misra, D. Machine Learning Algorithms and Their Application to Ore Reserve Estimation of Sparse and Imprecise Data, J. Intelligent Learning Systems & Applications, 2010, Volume 2, Pages 86-96.
[7] https://cmap.ihmc.us/cmaptools/, (2018), CmapTools: Construct, Navigate, Share and Criticize, The Institute for Human & Machine Cognition (IHMC), Accessed on: 2018-07-17.
[8] Pislaru, C. & Shebani, A. Identification of Nonlinear Systems Using Radial Basis Function Neural Network. World Academy of Science, Engineering and Technology International Journal of Mechanical and Mechatronics Engineering, 2014, Volume 8, No. 9, Pages 1678-1683.
[9] Fonseca, C. M. & Fleming, P. J. (1993). Genetic algorithm for multiobjective optimization, formulation, discussion and generalization. Genetic Algorithms: Proceeding of the Fifth International Conference. CA, pp. 416-423.
[10] Bessonov, A. A. (2012), Mnohokryteryalnaia neiroevoliutsyonnaia optimizatsiya nelyneinykh funktsii [Multiobjective Neuroevolutionary Optimisation of Nonlinear Functions], Information Processing Systems, Volume 107, No. 9, Pages 5-10 (in Russian).