IOP Conference Series: Materials Science and Engineering

Collecting diagnostic operational data from CNC machines during operation process

To cite this article: G M Martinov *et al* 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **709** 033051

IOP Conf. Series: Materials Science and Engineering **709** (2020) 033051 doi:10.1088/1757-899X/709/3/033051

Collecting diagnostic operational data from CNC machines during operation process

G M Martinov, R L Pushkov* and S V Evstafieva

MSTU Stankin, department of Computer Control Systems, 127055 Vadkovsky per 3a, Moscow, Russia

* pushkov@ncsystems.ru

Abstract. The article deals with issues related to obtaining operational data from process equipment and machine parameters of the CNC system in order to monitor the correct use of equipment, and to comply with the regulations for its operation [1]. Data is collected using data acquisition modules that can work with equipment using specific protocols. To store and process the received data using its own REST-server. The obtained data at the request of the user can be configured by selecting from a common array of data and designating them.

1. Introduction

One of the most important tasks of MES-systems (manufacturing execution system, production process control system) is the collection of information related to production from the automation systems of the production process, sensors, equipment. For each specialist working in an enterprise, different types of data collected are relevant. For example, for a technologist of interest are the following: monitoring in real time of the parameters of the processing mode, tracking possible critical loads on the tool, tracking the actual time distribution of processing modes and machine states. The following data is necessary for the mechanic: equipment operating time, reports on critical processing parameters, information from sensors, analysis of the temperature of technological components of the equipment, compliance with work regulations with equipment prior to its launch, during operation and at the end of work (for example, heating the spindle before starting work) control of the limiting parameters of the equipment). The use of the collected data will allow: reduce the number of scrap, rework, speed up equipment retooling, optimize the process, improve equipment accuracy, extend equipment life [2].

There are various ways of organizing the data acquisition subsystem from the process equipment. One option is to install smart sensors that, when the values of monitored parameters change, will send data to the shared cloud storage, for example, Microsoft Azure, Amazon Cloud. This approach has several drawbacks: the data is stored on a remote server and data integrity issues are beyond the competence of the enterprise. Also worth mentioning separately is the cost of using cloud storage. In this case, it depends on the required power of resources, i.e. if it is necessary to solve the problems of analysing the collected data, the cost of using the resource can seriously increase. This article assumes the use of your own server with software for collecting, storing and analysing data.

2. Architecture of the subsystem for collecting diagnostic and informational data

On the one hand, the data acquisition subsystem should interact with technological equipment, which can be controlled by different CNC systems, have different protocols for data transmission, etc., and on the other hand, be integrated into the enterprise MES system for data accumulation and processing [3-7].

The architecture of the subsystem for the collection of diagnostic and informational data is presented in Figure 1.

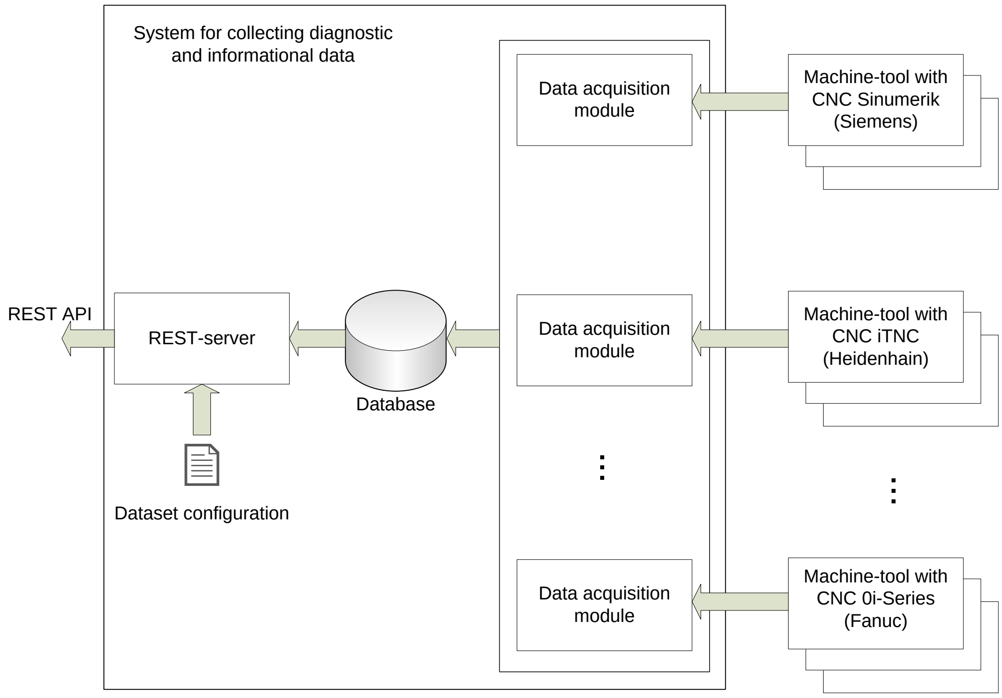

Figure 1. Architecture of the subsystem for collecting diagnostic and informational data.

CNC process equipment is polled by data acquisition modules with a specific period [8-15]. For each type of machine (turning, milling, controlled by different CNC systems), you can define your own data set for collecting from sensors and / or obtaining machine parameters of the system [16,17]. Each of the data acquisition modules is focused on interaction with a specific equipment according to a specific protocol and places the obtained data in a document-oriented database. The REST server (Representational State Transfer - “transfer of the presentation state”) accepts requests from the client, which can be a remote terminal, or a module that uses data for further calculations and provides information about equipment, data sets or the data itself in accordance with the data configuration, described for server [18,19].

3. Data configuration

In the considered architecture provides the ability to configure the data. The data configuration file contains metadata for all sets collected by the server. The JSON file (JavaScript Object Notation is a text-based data exchange format based on JavaScript) contains an array of descriptions of equipment items (for example, machines) and an array of descriptions of data sets.

Consider the system shown in Figure 2. Operational data is collected from two machines - turning and milling. Each machine has a spindle temperature sensor, and information about the spindle speed is also obtained.

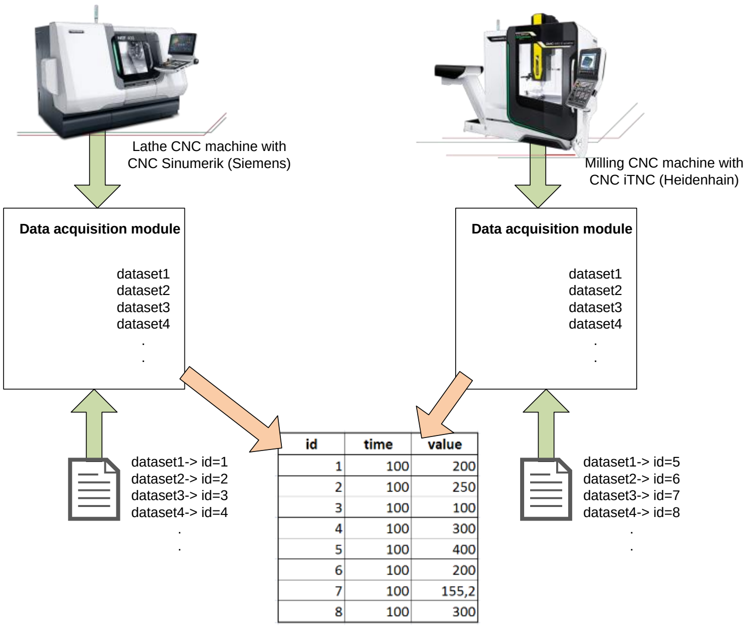

Figure 2. Collecting information about temperature and spindle speed from two machines.

The fragment of the configuration file corresponding to the data to be collected is shown in Listing 1. As the configuration data, types and names of the machines are indicated, the data from which will be collected, the data necessary for obtaining the data associated with the machine are indicated. Similarly, you can configure all the received data.

```
{ “machines”: [
  {    “id”: 1,
   “name”: “Lathe CNC Machine”,
   “description”: “Model TC-200”
  },
  {    “id”: 2,
   “name”: “CNC Milling Machine”,
   “description”: “Model MC-200”
  }
 ],
 “datasets”: [
  {    “id”: 1,
   “machine_id”: 1,
   “unit”: “rpm”,
   “description”: “Spindle speed”
  },
  {    “id”: 2,
   “machine_id”: 1,
   “unit”: “C”,
   “description”: “Spindle temperature”
  },
  ]
}
```

The data from the configuration file is requested by the interface to display the data sets, their descriptions, and accessories to certain equipment.

4. Presentation of data on the server

To store the values of parameters collected by the server, you can use one of the types of databases. Since the values of different parameters can be of different types and it is difficult to create an unambiguous database scheme, the use of relational database management systems is very problematic. Document-oriented database management systems, such as MongoDB, are better suited for storing such data sets. Data schemes and data itself are stored in it in BSON format, which is a binary JSON.

Each data point corresponds to a document whose structure contains:

- _id – ObjectId type parameter, which is a unique identifier (primary key) and is generated by the DBMS automatically when adding a point to the database;
- dataset – identifier of the data set (int, 32 bits), determines the ownership of the data set to a source. Each data source has its own unique identifier within the system.;
- time – parameter of type Date, which contains a time stamp of data in Unix format;
- value – directly the value of any valid BSON type.

5. Requests to the server

Requests to the REST server are made via the REST API. Figure 3 shows a fragment of the REST call set.

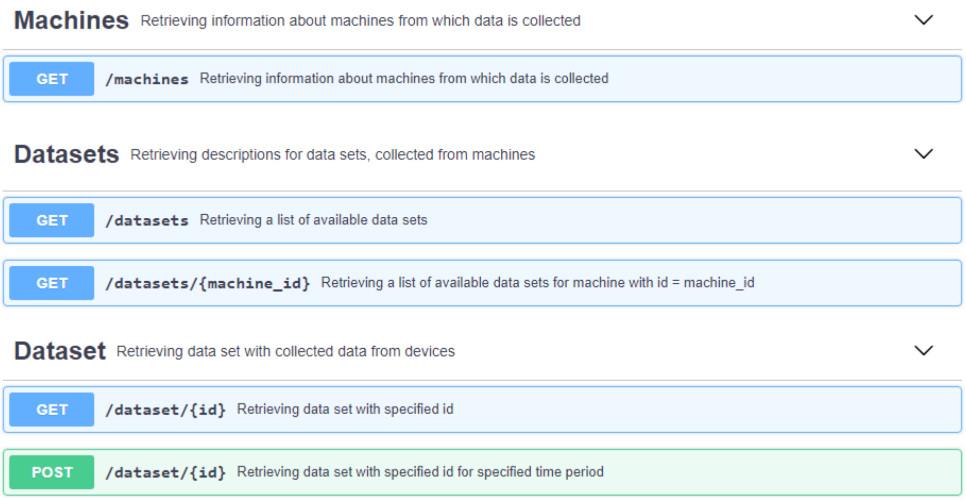

Figure 3. REST API call set example.

Calls are divided into three groups.:

1. Machines - receiving information about the equipment from which data is being received.
   - A call to GET / machines returns a list of equipment for which data is being collected.
2. Datasets - getting information about datasets from hardware.
   - A GET / datasets call returns a list of data sets that are collected by the server.
   - A call to GET / datasets / {machine_id} returns a list of data sets that are collected for equipment with a machine_id.
3. Dataset - getting data from a specific data set.
   - The GET / dataset / {id} call returns all the data set with id.
   - The POST / dataset / {id} call returns the data set with the identifier id for a certain time. The time range is transmitted as additional call parameters (Figure 4).

Name	Description
id * required `string` *(path)* *x-oad-type: parameter*	
begintime * required `integer($int64)` *(query)* *x-oad-type: parameter*	Begin time (Unix-format) for data retrieving period.
endtime * required `integer($int64)` *(query)* *x-oad-type: parameter*	End time (Unix-format) for data retrieving period.

Figure 4. Passing a time range as additional call parameters.

The types and formats of responses for the data set are shown in Figure 5:

Code	Description
200	*OK* Example Value | Model `[ { "time": 0, "Value": 0 } ]`
400	*Period is invalid*
404	*Data set with specified id not exist*
500	*Server error*

Figure 5. Types and formats of responses for the data set.

6. Practical implementation

Many instructions on the use of machines, especially with high-speed spindles, require the heating of the spindle assembly after a long idle. This is due to two factors:

1. it is necessary to warm up the bearings slowly and disperse the lubricant using the mechanisms of the spindle assembly before the spindle assembly begins to work under load;
2. if the spindle assembly has not reached the nominal temperature, then changes in its geometrical parameters are possible, which will lead to a decrease in the accuracy of the parts produced.

For example, the documentation for the spindle assembly indicates that when the machine is idle for more than four hours but less than eight days, it is necessary to start the warm-up cycle according to the following scheme:

1. start the spindle with a speed of 25% of the maximum, work for two minutes;
2. set the speed of 50% of the maximum, work for two minutes;
3. set the speed of 75% of the maximum, work for two minutes;
4. the spindle is ready for operation.

Information from the spindle node temperature sensors and the spindle speed is fed to the collection system server. In the future, it can be obtained, analyzed and used for any other purposes. For example, it is possible to check whether the operator fulfills the prescribed instructions for heating the spindle assembly, or when justifying the refusal of warranty repair for non-compliance with the operating regulations. The graphs in Figure 6 show the temperature of the spindle assembly and the speed of the spindle after turning on the machine. In this case, the heating of the spindle assembly is performed in accordance with the instructions [20-22].

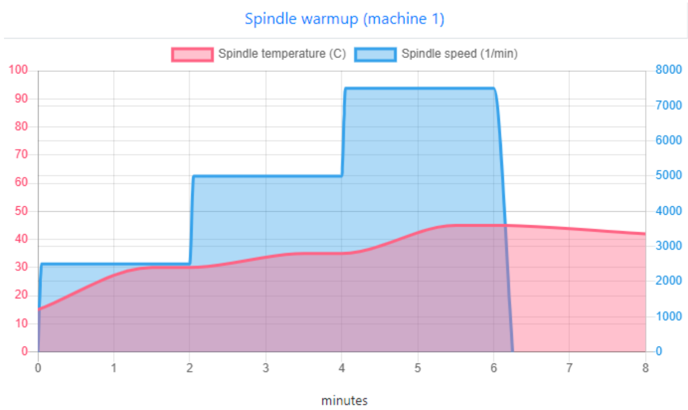

Figure 6. Temperature and speed of the spindle after turning on the machine.

One of the advantages of using this approach and technology is the ability to present data on the state of technological equipment in the web interface [9,10].

7. Conclusion

The proposed approach to organizing data collection software allows data collection from various technological equipment. For work with various types of equipment, including for different data transfer protocols, different data acquisition modules are used. Information from all modules comes in a single database, where it is accumulated and can be further processed and used. In most cases, the user does not need all the information stored in the database on the state of all elements of the technological equipment of the enterprise, but only some part of it, for example, information about the temperature of the spindle assembly on several machines, or about the load on the equipment of a particular section, or information from sensors of one particular machine. Selecting data and presenting them in a user-friendly form allows a REST server with a data configuration function.

Acknowledgments

This work was supported by the Ministry of Education and Science of the Russian Federation as part of the fulfillment of the state assignment (No. 2.1237.2017 / 4.6) and was carried out using equipment provided by the collective center of Moscow State University "STANKIN"

References

[1] Martinova L, Stas' A, Grigor'yev A and Babin M 2017 Avtomatizatsiya operatsionnogo kontrolya na frezernykh stankakh s CHPU *Avtomatizatsiya v promyshlennosti* **5** 33-6.
[2] Tretyakov I 2017 Monitoring oborudovaniya s CHPU: sbor i obrabotka mashinnyh dannyh *Promyshlennye stranicy Sibiri* **12** 28-31.
[3] Martinov G and Kozak N 2016 Specialized numerical control system for five-axis planing and milling center *Russian Engineering Research* **36**(3) 218-22.
[4] Martinov G and Kozak N 2015 Numerical control of large precision machining centers by the AxiOMA control system *Russian Engineering Research* **35**(7) 534-8.
[5] Martinov G, Kozak N, Nezhmetdinov R, Grigoriev A, Obukhov A and Martinova L 2017

IOP Conf. Series: Materials Science and Engineering **709** (2020) 033051 doi:10.1088/1757-899X/709/3/033051

Method of decomposition and synthesis of the custom CNC systems *Automation and Remote Control* **78**(3) 525-36.
[6] Martinov G, Obukhov A and Kozak N 2018 The Usage of Error Compensation Tools of CNC for Vertical Milling Machines *Russian Engineering Research* **38**(2) 119-22.
[7] Martinova L, Kozak N, Nezhmetdinov R, Pushkov R and Obukhov A 2015 The Russian multi-functional CNC system AxiOMA control: Practical aspects of application *Automation and Remote Control* **76**(1) 179-86.
[8] Pushkov R, Salamatin E and Evstafieva S 2018 *International Conf. on Modern Trends in Manufacturing Technologies and Equipment* (MATEC Web Conf) 224 pp 1-7.
[9] Martinova L and Martinov G 2018 *3rd Russian-Pacific Conf. on Computer Technology and Applications* (Vladivostok) pp 1 - 4.
[10] Martinova L and Fokin N 2018 *International Conf. on Modern Trends in Manufacturing Technologies and Equipment* (MATEC Web Conf) pp1-5.
[11] Kovalev I, Nikishechkin P and Grigoriev A 2017 *International Conference on Industrial Engineering, Applications and Manufacturing* (ICIEAM) pp.1-4. ()
[12] Martinova L, Grigoryev A and Sokolov S 2012 Diagnostics and forecasting of cutting tool wear at CNC machines *Automation and Remote Control* **73**(4) 742-9.
[13] Nezhmetdinov R, Sokolov S, Obukhov A and Grigor'ev A 2014 Extending the functional capabilities of NC systems for control over mechano-laser processing *Automation and Remote Control* **75**(5) 945-52.
[14] Nezhmetdinov R, Nikishechkin P and Nikich A 2018 *International Russian Automation Conf (RusAutoCon)* (Sochi: IEEE)
[15] Nezhmetdinov R, Nikishechkin P, Kovalev I and Chervonnova N 2017 Podkhod k postroyeniyu sistem logicheskogo upravleniya tekhnologicheskim oborudovaniyem dlya realizatsii kontseptsii «Industriya 4.0» *Avtomatizatsiya v promyshlennosti* **5** 5-9.
[16] Nikishechkin P,. Kovalev I and Nikich A 2017 *International Conf. on Modern Trends in Manufacturing Technologies and Equipment* (MATEC Web Conf) 129 p.03012.
[17] Martinova L, Sokolov S and Nikishechkin P 2015 *6th International Conference CCI Advances in Swarm and Computational Intelligence* (Proc. Part II) pp.200-207.
[18] Kozak N, Nezhmetdinov R and Martinova L 2018 Integratsiya dannykh sistem logicheskogo upravleniya v «umnoye» proizvodstvo na osnove kontseptsii Industry 4.0 *Avtomatizatsiya v promyshlennosti* **5** 11-5.
[19] Nikishechkin P, Kovalev I, Grigor'yev A and Nikich A 2017 Krossplatformennaya sistema sbora i obrabotki diagnosticheskoy informatsii o rabote tekhnologicheskogo oborudovaniya *Vestnik MGTU Stankin* **1** (40) 94-8.
[20] Nikishechkin P, Chervonnova N and Nikich A 2018 *International Conf. on Modern Trends in Manufacturing Technologies and Equipment* (MATEC Web Conf) 224 pp 1-9.
[21] Chekryzhov V, Kovalev I and Grigoriev A 2018 *International Conf. on Modern Trends in Manufacturing Technologies and Equipment* (MATEC Web Conf) 224 pp.1-7.
[22] Nikishechkin P, Chervonnova N and Nikich A 2018 Podhod k postroyeniyu spetsializirovannyh portativnyh terminalov dlya kontrolya i upravleniya tekhnologicheskim oborudovaniyem *Avtomatizatsiya v promyshlennosti* **6** 63-7.