On-Board Data Management Structure For Advanced Construction Machine Support

by

Jarosław Jurasz[1], Agata Ligier[1], Alfons Horn[2], Jochen Wendebaum[1]

ABSTRACT: A flexible on-board data management structure for a construction machine and interfaces between its elements are discussed. The solution, which grants the required flexibility and enables advanced, distributed functions is discussed on the example of the road paving process.

KEYWORDS: CANopen; digital work documentation; computer integrated road construction; road product model; site information systems; OPC; XML

1. INTRODUCTION

Nowadays the construction machines are increasingly often equipped with the on-board computers, which support the operator and take over the control and documentation functions. Their application is especially promising in civil engineering, where the machines perform repeatable and well-defined tasks, e.g. laying, compacting, grading. Thanks to advances in positioning technology, on-board communication and following the need of the users, who wish to actively participate in the quality control (Build-Own-Operate etc.), the on-board IT plays increasing role on the modern construction site.

Major part of the requirements set for the on-board IT is due to the need for applicability to different machines and worksites. This includes varying configuration of positioning equipment, sensors and measurement systems coming from different providers.

With the increasing number and extending functionality of CIRC systems in existence, the problems of interoperability and standardisation start to play a major role. Those topics are addressed in the Osyris project, in the frame of which this work is carried out.

The goal set by the authors is to design a flexible on-board data management structure for a construction machine, equipped with standardised interfaces. The article is structured as follows: after discussing the context and the requirements for the on-board structure the contents of the information is investigated. Then the Osyris layered on-board structure is presented, followed by two examples of advanced, distributed functionality.

2. CONTEXT

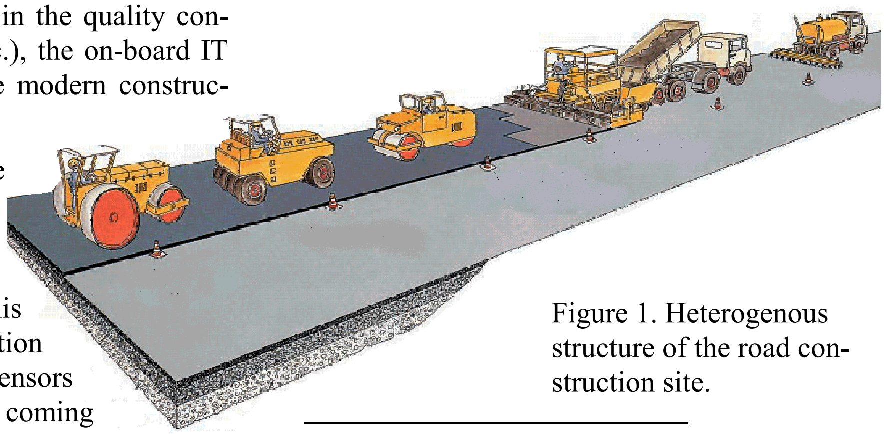

Figure 1. Heterogenous structure of the road construction site.

[1] Institute for Technology and Management in Construction (TMB), University of Karlsruhe, Am Fasanengarten, D-76128 Karlsruhe, Germany, tel. +49 721 608-3884, e-mail: {jurasz | ligier | wendebaum}@tmb.uka.de

[2] Moba Mobile Automation GmbH, Vor den Eichen 4, D-65604 Elz, Germany, tel. +49 6431 9577-132, e-mail: horn@moba.de

The major challenge for the Osyris on-board data management structure is due to the heterogeneous information structure of the worksite (see Fig. 1.), featuring:

- Separate processes (e.g. earthmoving, laying, re-profiling, transport), often performed by separate organisations,
- Few standardised interfaces; GPS NMEA standard[3] is one notable exception,
- Differently equipped machines coming from different manufacturers,
- Varying configuration (ad-hoc task allocation and team definition, add-on sensors, different operating modes).

Clearly a separation is needed between the managed process information and the details of the algorithm used to obtain or process it. To assure flexibility and extensibility of the implementation we propose to fix only data storage structure and its interfaces. In this way the information can be accessed and/or provided from any component without revealing the details of how the information was acquired. As the machines work in team, wireless real-time interfaces to the other machines have to be taken into account.

Nowadays 80..90% of the projects conducted by European road contractors are various maintenance tasks. The typical limited maintenance configuration (small paver and 1-2 small rollers) requires simple and cost-effective solutions.

3. INFORMATION CONTENTS

The contents of the managed information is defined in the Osyris Functional Design. A distinction between volatile real-time and static information can be made here. The most important data categories managed on-board are (see Fig. 2.):

- **Design** – a description of a target road in a suitable digital form (what). Remains static during the project execution.
- **Mission** – a description of the task to be performed by a given machine (how). Remains static for the given mission, changing typically daily.
- **Machine state** consists of position, tool geometry and the process data, which vary in real-time. It can be used to derive the achieved work.
- **Target machine state** is a subset of machine state subject to automatic control, derived from design, mission and current machine state, for example designed elevation at the given point.
- **Achieved work** is a record of work execution. It is gathered in real time and remains static after the work execution.

All the data categories listed above can be expressed as parameter values assigned to a geometry, where design, tool or achieved geometry can be used to carry the parameters. Depending on the geometry they are assigned to, the same parameters can be used to specify the actual or target values. The managed parameters can be outlined as follows:

- Position: Geographic, Curvilinear coordinates, linear and angular speed
- Inherent properties of the material: temperature and contents
- Geometrical properties: thickness, evenness, volume, level deviation
- Process and machine parameters: vibration, tampering, hydraulic pressure.
- Ambient conditions (temperature, wind, sunlight)

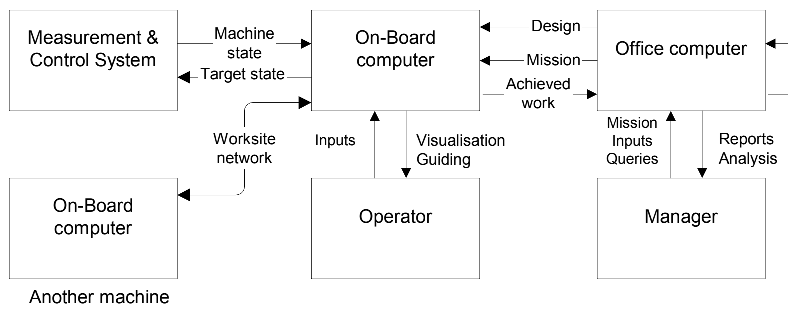

Figure 2. Functional decomposition of an OSYRIS system

Clearly it is not possible to define an exhaustive list of the parameters, so the definition of the additional parameters must be possible at the runtime. The identification of parameters can be performed using standardised names or codes. In any case it is very important to clearly define the meaning and units of the parameter values.

[3] Although many GPS receivers offer best performance using custom interfaces.

The piecewise-linear ribbon structure[4] [1] can be used to efficiently represent all data categories listed above. In the case of achieved work description, the tool geometry with assigned parameters describing state can be used as a ribbon generator. The ribbons introduce the natural ordering of the target parameter values along the road axis, so that extrapolation principle can be used to limit the number of managed parameter values.

It is important to note that multiple values of one parameter may coexist at given time. For example the following values of machine speed need to be managed concurrently on-board:

- measured by absolute positioning device, e.g. GPS or Robotic Total Station
- measured by local positioning device, e.g. encoder
- elaborated by data fusion algorithm, e.g. Kalman filter
- specified as target for the mission.

4. LAYERED ON-BOARD STRUCTURE

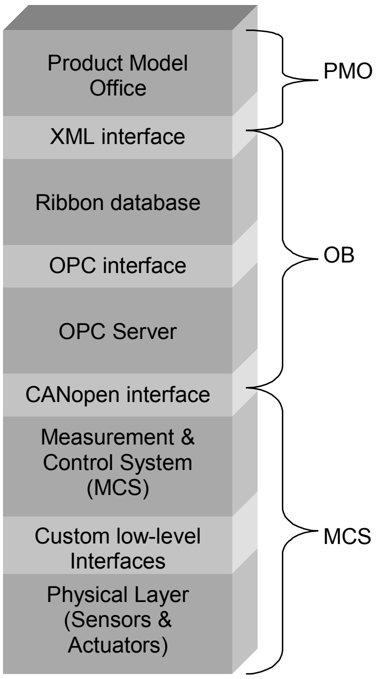

Figure 3. Osyris layered on-board structure

The Osyris system is based on components. In comparison with monolithic systems (e.g. CIRC [3]) this approach enables easy reconfiguration and provides framework for specialised components which provide advanced services (e.g. soft sensors).

The on-board system layers are presented in the Fig. 3. The abstraction level, amount of managed data, and required performance depend on the specialisation level:

PMO – responsible for permanent storage, mission preparation and work analyses [2].

OB – responsible for visualisation and storage functions, best effort (not guaranteed) real time behaviour.

MCS – responsible for guaranteed real time interfacing, filtering and control functions. Multiple MCS devices are possible, although as of today an architecture with single, central MCS is the most economical solution.

4.1. CANopen interface

The Osyris CANopen interface approach is based on concept of a device profile, describing the measurement and control objects provided by the real time measurement and control systems. The fast exchange of measured data between the MCS and the on-board computer is guaranteed by the efficient CANopen communication layer (so called communication profile). The CANopen communication profile is already standardised.

Former approaches to the interfacing to the measurement and control systems on construction machines were based on lower level solutions (e.g. CAN) and, as the interface description was hardcoded, changes in the system structure required a lot of modifications on software and hardware level, which led to additional costs and less reliable systems.

Therefore the Osyris solution promotes a standardised communication for construction machines. As a part of the proposed solution the standard connector is also presented. The advantages of standardised devices are numerous. It encourages the manufacturers to produce standard measurement and control devices with their own technology hidden behind the standard interface.

Furthermore the open solution allows contractors to use construction machines coming from different manufactures and connect them

[4] The ribbon is created by moving a planar figure called generator along a curve called axis [1]

to one on-site management system, without any adaptations or modification in the MCS.

The central part of the device profile is the object dictionary description (an extract is shown in Table 1). The object dictionary is essentially a grouping of objects accessible via CANopen in an ordered, predefined fashion. Each object within the dictionary is addressed using a 16-bit index and 8-bit sub-index. The Object Dictionary contains only few mandatory items. They can be accessed according to the worksite requirements and a current configuration. The MCS not only offers but also requires items. The availability of items is resolved by the on-board computer in runtime.

Table 1. Extract of the paver device profile

Index	Object	Description
6000	**Type of machine**	Kind and type of machine
6010	**MCS functionality**	Indices of items supported by particular implementation
6020	**Event**	General (start, stop etc) and machine-specific (vibration on/off etc) events
6100	**Position**	The geographic position of machine's tool in local coordinate system (E,N,H)
6101	**Angle Position**	The attitude of the tool
6102	**Curvilinear Co-ordinates**	The coordinates of the tool in the local curvilinear system
6103	**Level Deviation**	The levelling error
6200	**Thickness**	The thickness of the laid layer
6300	**Screed width**	The width of the screed and its extensions
6310	**Volume**	The volume of the laid material
6500	**Material core temperature**	The temperature of the laid material
8010	**MCS wish list**	Indices of items required by the MSC

4.2. OPC layer

The standardised interface to the actual machine state parameters at the level of on-board computer is defined with help of the OPC.

OPC (OLE for Process Control) is a standardised set of interfaces, based on OLE/COM and DCOM technology, for open software application interoperability between control applications, field systems and devices, and business/office applications [4]. Osyris uses a subset of OPC called Data Access. The data exchange is based on the server (provider)-client (consumer) principle (see Fig. 4.).

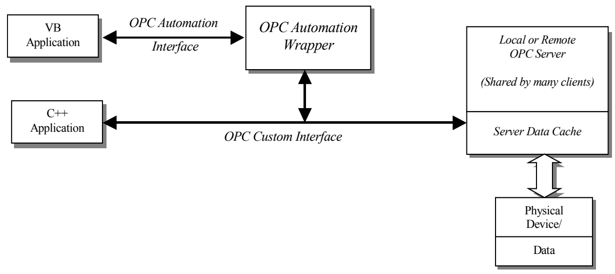

Figure 4. OPC Architecture

OPC provides a snapshot of current machine state as a hierarchy of named items together with the time stamp and quality information. The items have standardised names and are mostly numeric values (vector and string are also possible). The scaling between metric (SI) units and device units is performed by the Osyris OPC Server.

Osyris OPC namespace is divided in the following groups, mostly according to the origin:

- CanOpen – machine state as received from the MCS
- CanOpenOut – volatile target machine state (e.g. level deviation) elaborated by the on-board computer, transmitted to the MCS
- Designed – target state (e.g. thickness) as defined in the mission
- Pos – machine position elaborated by the positioning component
- Manual – parameter values may be overridden manually by the operator
- Default – for many parameters, e.g. Temperature or Speed sensible default values can be defined here, and used in case of lack of sensors
- Process-specific groups, e.g. Material, CES (Compaction Expert System), CM (Cooling Model)

In the Osyris implementation there is only one Osyris OPC server, which allows reading and writing clients (no arbitration for writing is guaranteed at the OPC level). The Osyris implementation uses the OPC concept not only for client-device communication, but additionally (which is an extension to the OPC standard) as a storage for the current machine state. Also clients which perform parameter estimation (soft sensors) can write into current state. Writing to the device (in this case MCS) is allowed only in the CanOpenOut group.

In addition to the standard OPC, a concept of the 'unqualified items' has been introduced. As already mentioned, there can exist more than one value for a parameter. If the client component is interested in the ‘best’ value, it subscribes to the unqualified item (without specifying the group) and the server decides which one it is, based on the priority list and current qualities. (see Fig. 5.)

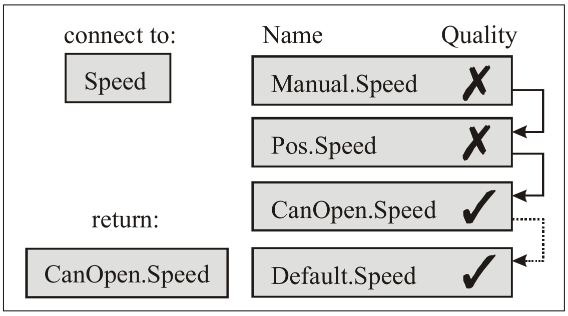

Figure 5. Example of Osyris OPC connection using an unqualified item name.

Further extension of the OPC standard is the meta-information contained in the 'Events' OPC group. This concept is used to notify the client that the best source of data has changed. In this way a fallback mechanism is implemented.

The OPC standard includes also a time-referenced interface to the past data, so called historical OPC. Unfortunately it cannot be employed in the CIRC context, as space reference is missing. OPC can be also accessed remotely via DCOM. It is not used directly by the Osyris framework, but can be advantageous for debugging, scripting or Web clients.

4.3. *Ribbon database*

As described in [1], ribbons (Fig. 6.) can be used as a permanent storage for design, mission and achieved work.

For each machine the tool geometry (defined in mission or measured) is a machine ribbon generator. There exist one dynamic ribbon for each machine, containing data gathered in real time (for more complex tool geometries multiple ribbons are foreseen). The recorded position is represented by a ribbon diagonal, with the parameters assigned to it.

Ribbons provide following services:

- ❖ Map visualisation of the parameters
- ❖ Storage for cooling model and other algorithms
- ❖ Geometrical searching and iteration
- ❖ Parameter interpolation, e.g. elevation, thickness interpolation
- ❖ Curvilinear transformation.

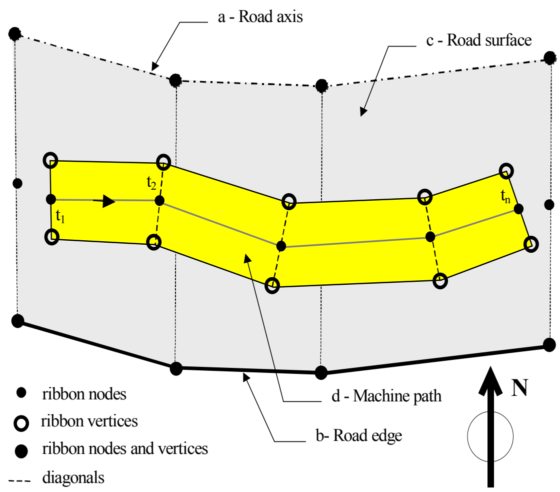

Figure 6. Sample ribbons in geographic coordinates representing: a - road axis, b – road edge, c – road surface and d - machine path

4.4. *XML interfaces*

The interfaces between the on-board and the Office components are based on the XML standard. XML standard is wide spread in Internet applications and many tools are available. A concept central to XML is the separation of the information contents and format. The information contents of the geometry-based Osyris XML files is based on ribbon concept.

At the beginning of work the on-board must be supplied with mission information (worksite geometry, designed values of the work parameters like speed, fleet configuration). At the end of the work the achieved information has to be exported. There is also a request-response schema, allowing the transfer of the current work state at any time. The detailed schema has been defined for each exchange file format.

5. DISTRIBUTED OPERATION

5.1. *Wireless communication*

Wireless communication between machines is based on a reliable multicast implemented upon Internet datagram (UDP) protocol. Typically WaveLANs are used as physical transport. Ribbons with parameters are automatically synchronised among the machines. This property can be used to implement distributed algorithms (see the following subchapters).

5.2. Distributed pass counting

One of the most valuable information concerning the quality of compaction comes from the pass counting. Given the past position of the compactor, the number of the done passes can be calculated for each point of the surface, resulting e.g. in a compaction coverage map. On the most worksites at least two compactors are working together. The coverage map is only then useful if it contains passes from all the machines, and this is guaranteed by the wireless communication algorithm.

5.3. Cooling model

Another important parameter to take into account during the asphalt laying is the temperature of the asphalt layer. Attempts to measure the surface temperature of the asphalt at the compactor using infrared sensors are unreliable due to the high influence of the wind speed on the surface temperature.

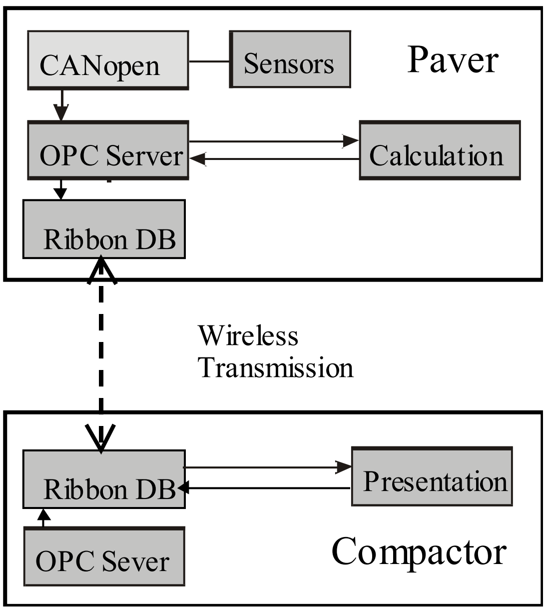

Figure 7. Cooling model as an example of a distributed application in the OSYRIS system

To calculate the evolution of the temperature inside an asphalt layer, a mathematical model has been used. The model parameters like *layer thickness* and *asphalt temperature at laying* can only be measured at the time of paving on the paver, but the results of the core temperature calculation are most interesting for the compactor operator, giving him the very valuable information about the time left to finish the compaction (Fig. 7.).

6. CONCLUSIONS

The presented on-board structure is universal: all the relevant process parameters are represented in uniform and coherent way and can be transparently accessed on all the levels of the system. Pre-existing standards have been used as a base for the Osyris interfaces.

The presented solution grants the required flexibility and enables advanced, distributed functions, for example distributed pass counting and real-time simulation of the asphalt cooling (cooling model).

Validation of the proposed framework on the worksite is planned for Autumn 2002. The specifications of the interfaces will be published at the end of the project.

Acknowledgements

The work described in the paper was carried out in the framework of the European Project OSYRIS: Open System for Road Information Support (Growth No. GRD1-1999-10815), 2000-2003 [5].

7. REFERENCES

[1] J. Jurasz, Universal digital environment model for computer integrated road construction. Proc. 18th ISARC 31-36, Cracow 2001

[2] A. Ligier, J. Fliedner, J. Kajanen, F. Peyret, *Open System For Road Information Support*. Proc. 18th ISARC, Cracow 2001

[3] F. Peyret, J. Jurasz, A. Carrel, E. Zekri, B. Gorham: *The Computer Integrated Road Construction project*, Automation in Construction 9 (5-6) (2000) pp. 447-461.

[4] OPC Foundation at http://www.opcfoundation.org

[5] Osyris Website at http:// www.osyris.org