

**VDMA 40077**

ICS 25.120.20; 83.200

Replaces
VDMA 40077:2019-11**OPC UA interfaces for plastics and rubber machinery –
Data exchange between injection moulding machines and MES**OPC UA Schnittstellen für Kunststoff- und Gummimaschinen –
Datenaustausch zwischen Spritzgießmaschinen und MES**VDMA 40077:2020-06 is identical with OPC 40077 (Release 1.01) and
EUROMAP 77 (Release 1.01)**

Document comprises 21 pages

VDMA

Contents

	Page
Foreword.....	7
1 Scope	8
2 Normative references	8
3 Terms, definitions and conventions	9
3.1 Overview	9
3.2 Conventions used in this document.....	9
3.3 Abbreviations	9
4 General information to OPC UA interfaces for plastics and rubber machinery and OPC UA	9
5 Use cases	9
6 Basic security requirements.....	9
6.1 Application Security	9
6.2 User security/Access control	10
7 IMM_MES_InterfaceType.....	10
8 MachineInformation.....	13
8.1 DeviceClass.....	13
8.2 LogbookEvents	13
9 MachineConfiguration	13
10 MachineMESConfiguration	13
11 MachineStatus	13
12 MachineMESStatus.....	13
13 PowerUnits	14
14 Moulds	14
15 InjectionUnitsType.....	14
16 InjectionUnitType ObjectType Definition	15
16.1 Overview	15
16.2 Index.....	15
16.3 BarrellId	15
16.4 IsPresent.....	15
16.5 InProduction.....	15
16.6 ScrewId	15
16.7 ScrewDiameter	15
16.8 ScrewVolume	16
16.9 MaxScrewStroke	16
16.10 TemperatureZones	16
17 Jobs.....	16
17.1 Types, methods and events for cyclic jobs	16

17.2	Optional properties in CyclicJobInformationType.....	16
17.3	InjectionUnitCycleParametersType	16
18	ProductionDatasetManagement	18
19	Profiles and Conformance Units.....	18
20	Namespaces.....	19
20.1	Namespace Metadata	19
20.2	Handling of OPC UA Namespaces.....	20
Annex A	(normative) OPC 40077 Namespace and mappings.....	21

Figures

Figure 1 – IMM_MES_InterfaceType Overview.....	11
------------------------------------------------	----

Tables

Table 1 – IMM_MES_InterfaceType Definiton.....	11
Table 2 – IMMMessageClassificationEnumeration Definition	14
Table 3 – InjectionUnitsType Definition	15
Table 4 – InjectionUnitType Definiton.....	15
Table 5 – Types, methods and events for cyclic jobs.....	16
Table 6 – Example of an event type derived from CycleParametersEventType with two moulds and two injection units	16
Table 7 – InjectionUnitCycleParametersType Definition	17
Table 8 – Description of cycle parameters	18
Table 9 – Profile URIs for OPC 40077	19
Table 10 – OPC 40077 Basic Server Profile Definition	19
Table 11 – OPC 40077 Jobs Server Facet Definition.....	19
Table 12 – OPC 40077 ProductionDatasetManagement Server Facet Definition	19
Table 13 – NamespaceMetadata Object for this Specification.....	20
Table 14 – Namespaces used in an OPC 40077 Server	20
Table 15 – Namespaces used in this specification.....	20

OPC Foundation / EUROMAP

AGREEMENT OF USE

COPYRIGHT RESTRICTIONS

- This document is provided "as is" by the OPC Foundation and EUROMAP.
- Right of use for this specification is restricted to this specification and does not grant rights of use for referred documents.
- Right of use for this specification will be granted without cost.
- This document may be distributed through computer systems, printed or copied as long as the content remains unchanged and the document is not modified.
- OPC Foundation and EUROMAP do not guarantee usability for any purpose and shall not be made liable for any case using the content of this document.
- The user of the document agrees to indemnify OPC Foundation and EUROMAP and their officers, directors and agents harmless from all demands, claims, actions, losses, damages (including damages from personal injuries), costs and expenses (including attorneys' fees) which are in any way related to activities associated with its use of content from this specification.
- The document shall not be used in conjunction with company advertising, shall not be sold or licensed to any party.
- The intellectual property and copyright is solely owned by the OPC Foundation and EUROMAP.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OPC or EUROMAP specifications may require use of an invention covered by patent rights. OPC Foundation or EUROMAP shall not be responsible for identifying patents for which a license may be required by any OPC or EUROMAP specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OPC or EUROMAP specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

WARRANTY AND LIABILITY DISCLAIMERS

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OPC FOUNDATION NOR EUROMAP MAKES NO WARRANTY OF ANY KIND, EXPRESSED OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OPC FOUNDATION NOR EUROMAP BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you.

RESTRICTED RIGHTS LEGEND

This Specification is provided with Restricted Rights. Use, duplication or disclosure by the U.S. government is subject to restrictions as set forth in (a) this Agreement pursuant to DFARs 227.7202-3(a); (b) subparagraph (c)(1)(i) of the Rights in Technical Data and Computer Software clause at DFARs 252.227-7013; or (c) the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 subdivision (c)(1) and (2), as applicable. Contractor / manufacturer are the OPC Foundation, 16101 N. 82nd Street, Suite 3B, Scottsdale, AZ, 85260-1830

COMPLIANCE

The combination of EUROMAP and OPC Foundation shall at all times be the sole entities that may authorize developers, suppliers and sellers of hardware and software to use certification marks, trademarks or other special designations to indicate compliance with these materials as specified within this document. Products developed using this specification may claim compliance or conformance with this specification if and only if the software satisfactorily meets the certification requirements set by EUROMAP or the OPC Foundation. Products that do not meet these requirements may claim only that the product was based on this specification and must not claim compliance or conformance with this specification.

TRADEMARKS

Most computer and software brand names have trademarks or registered trademarks. The individual trademarks have not been listed here.

GENERAL PROVISIONS

Should any provision of this Agreement be held to be void, invalid, unenforceable or illegal by a court, the validity and enforceability of the other provisions shall not be affected thereby.

This Agreement shall be governed by and construed under the laws of Germany.

This Agreement embodies the entire understanding between the parties with respect to, and supersedes any prior understanding or agreement (oral or written) relating to, this specification.

Foreword

This specification was created by a joint working group of the OPC Foundation and EUROMAP. It is adopted identically as VDMA Specification.

Amendments

The previous edition of this document is EUROMAP 77:2018 (Release 1.00, Editorial update 1.00a) which has been adopted as VDMA Specification VDMA:40077:2019-11 without any modification. It has not been released under the OPC Foundation, because the joint working group was established afterwards.

This document differs from the previous version as follows:

- a) The namespace has been changed to <http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/>.
- b) URIs for Profiles have been added.
- c) It has been adopted to the template for Companion Specifications of the OPC Foundation.
- d) Errors in the NodeSet-file have been corrected.

Previous editions

EUROMAP 77, version 1.00a = VDMA 40077:2019-11

EUROMAP

EUROMAP is the European umbrella association of the plastics and rubber machinery industry which accounts for annual sales of around 13.5 billion euro and a 40 per cent share of worldwide production. Almost 75 per cent of its European output is shipped to worldwide destinations. With global exports of 10.0 billion euro, EUROMAP's around 1,000 machinery manufacturers are market leaders with nearly half of all machines sold being supplied by EUROMAP members.

EUROMAP provides technical recommendations for plastics and rubber machines. In addition to standards for machine descriptions, dimensions and energy measurement, interfaces between machines feature prominently. The provision of manufacturer independent interfaces ensures high levels of machine compatibility.

OPC Foundation

OPC is the interoperability standard for the secure and reliable exchange of data and information in the industrial automation space and in other industries. It is platform independent and ensures the seamless flow of information among devices from multiple vendors. The OPC Foundation is responsible for the development and maintenance of this standard.

OPC UA is a platform independent service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications into one extensible framework. This multi-layered approach accomplishes the original design specification goals of:

- Platform independence: from an embedded microcontroller to cloud-based infrastructure
- Secure: encryption, authentication, authorization and auditing
- Extensible: ability to add new features including transports without affecting existing applications
- Comprehensive information modelling capabilities: for defining any model from simple to complex

1 Scope

OPC 40077 describes the interface between injection moulding machines (IMM) and manufacturing execution systems (MES) for data exchange. MES are used for collecting the information generated by IMM at a central point for easier quality assurance and job and dataset management. The target of OPC 40077 is to provide a unique interface for IMM and MES from different manufacturers to ensure compatibility.

The following functionalities are covered:

- General information about the IMM (manufacturer, model, serial number...), current configuration and status of the IMM including moulds, injection units and power units, and logbook of relevant changes on the machine.
- Job management: Information on the jobs running on the machine and the parameters of the production cycles and methods to send jobs from the MES to the IMM and to release the production.
- Dataset management: IMM store their configurations in so-called datasets. These include information on nominal process parameters (times, temperatures, pressures ...) related to the IMM but also to installed handling systems. OPC 40077 allows transferring datasets between IMM and MES for building a central repository of datasets.

Following functions are not included:

- Safety related signals like emergency stop
- Direct control of machine movements by the MES

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies

OPC 10000-1, *OPC Unified Architecture - Part 1: Overview and Concepts*

<http://www.opcfoundation.org/UA/Part1/>

OPC 10000-2, *OPC Unified Architecture - Part 2: Security Model*

<http://www.opcfoundation.org/UA/Part2/>

OPC 10000-3, *OPC Unified Architecture - Part 3: Address Space Model*

<http://www.opcfoundation.org/UA/Part3/>

OPC 10000-4, *OPC Unified Architecture - Part 4: Services*

<http://www.opcfoundation.org/UA/Part4/>

OPC 10000-5, *OPC Unified Architecture - Part 5: Information Model*

<http://www.opcfoundation.org/UA/Part5/>

OPC 10000-6, *OPC Unified Architecture - Part 6: Mappings*

<http://www.opcfoundation.org/UA/Part6/>

OPC 10000-7, *OPC Unified Architecture - Part 7: Profiles*

<http://www.opcfoundation.org/UA/Part7/>

OPC 10000-8, *OPC Unified Architecture - Part 8: Data Access*

<http://www.opcfoundation.org/UA/Part8/>

OPC 10000-9, *OPC Unified Architecture - Part 9: Alarms and Conditions*

<http://www.opcfoundation.org/UA/Part9/>

OPC 10000-11, *OPC Unified Architecture - Part 11: Historical Access*

<http://www.opcfoundation.org/UA/Part11/>

OPC 10001-1, *OPC Unified Architecture V1.04 - Amendment 1: AnalogItem Types*

<http://www.opcfoundation.org/UA/Amendment1/>

OPC 10001-3, *OPC Unified Architecture V1.04 - Amendment 3: Method Metadata*

<http://www.opcfoundation.org/UA/Amendment3/>

OPC 10000-100, *OPC Unified Architecture - Part 100: Devices*

<http://www.opcfoundation.org/UA/Part100/>

OPC 40083: OPC UA interfaces for plastics and rubber machinery – General Type definitions (version 1.02)

<http://www.opcfoundation.org/UA/PlasticsRubber/GeneralTypes>

3 Terms, definitions and conventions

3.1 Overview

It is assumed that basic concepts of OPC UA information modelling are understood in this specification. This specification will use these concepts to describe the OPC 40077 Information Model. For the purposes of this document, the terms and definitions given in the documents referenced in Clause 2 apply.

Note that OPC UA terms and terms defined in this specification are *italicized* in the specification.

3.2 Conventions used in this document

The conventions described in OPC 40083 apply.

3.3 Abbreviations

IMM Injection Moulding Machine

MES Manufacturing Execution System

4 General information to OPC UA interfaces for plastics and rubber machinery and OPC UA

For general information on OPC UA interfaces for plastics and rubber machinery and OPC UA see OPC 40083.

5 Use cases

OPC 40077 covers the following functionalities:

- General information about the IMM (manufacturer, model, serial number...), current configuration and status of the IMM including moulds, injection units and power units, and logbook of relevant changes on the machine.
- Job management: Information on the jobs running on the machine and the parameters of the production cycles and methods to send jobs from the MES to the IMM and to release the production.
- Dataset management: IMM store their configurations in so-called datasets. These include information on nominal process parameters (times, temperatures, pressures ...) related to the IMM but also to installed handling systems.

6 Basic security requirements

6.1 Application Security

For the communication between IMM and MES the OPC UA application authentication via X509 certificates shall be used. OPC UA provides functionalities for using self-signed certificates that have to be manually added to a “trust list” as well as for certificates issued by a certificate authority (CA).

The minimum requirements of the protocol level for a OPC 40077 compliant connection are:

- Use of (self-signed) certificates for OPC UA application authentication

- Security Policy: Basic256
- Message Security Mode: sign

NOTE: It is not fixed by this specification if the certificate includes a fixed IP address and/or the host name. However, if the certificate includes a host name, a DNS server is expected to resolve the host name. An OPC UA GDS (Global Discovery Server) can be used to manage the connections and certificates.

6.2 User security/Access control

6.2.1 On IMM

On the IMM authentication via user name and password is commonly used.

6.2.2 On MES

For the users and roles of the connection the following applies:

- User names can be manufacturer dependent.
- Standard roles are
 - “OPC40077”: read and write access for selected parameters
 - “OPC40077_read_only”: no writing permissions
- Manufacturers can add additional roles. They may not start with “OPC40077”. For these roles, more parameters can be writeable than for the OPC40077 role.
- The standard user “OPC40077” has the role “OPC40077” (and no other additional role), “OPC40077_read_only” has the role “OPC40077_read_only” (and no other additional role); the passwords for the standard users are defined by the manufacturers (they may be empty).

NOTE: OPC UA also allow an anonymous-token (e.g. for testing)

7 IMM_MES_InterfaceType

This OPC UA *ObjectType* is used for the root *Object* representing an injection moulding machine with all its subcomponents. It is formally defined in Table 1.

The instance(s) of *IMM_MES_InterfaceType* shall be located under the *DeviceSet Object* of the Server (see OPC UA Part 100).

NOTE: In most cases the OPC UA server will be implemented in the control of the IMM so only one instance of *IMM_MES_InterfaceType* will be created. But it is also possible that one OPC UA server is connected to several machine controls as one interface to the MES. In this case several instances of *IMM_MES_InterfaceType* will be created.

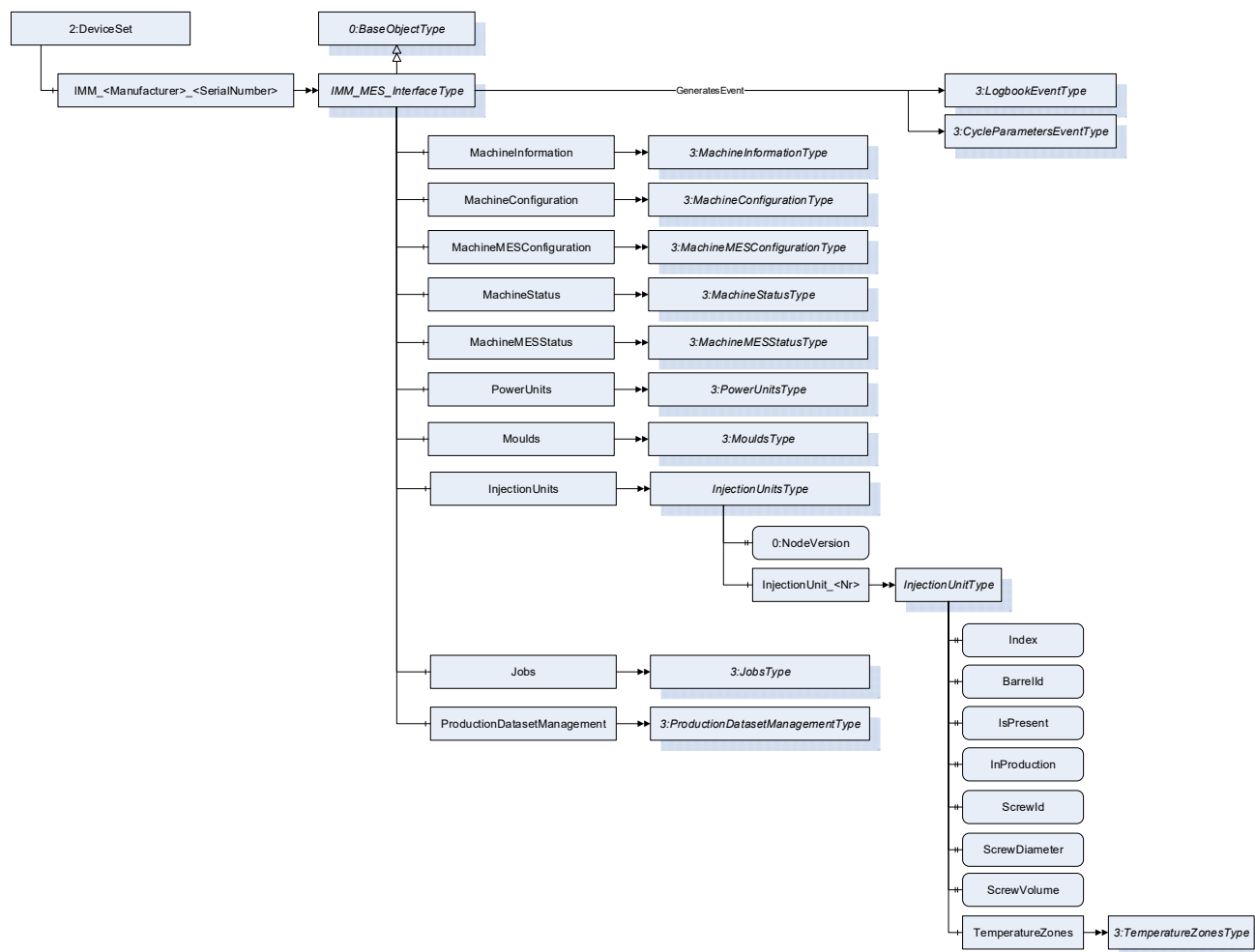


Figure 1 – IMM_MES_InterfaceType Overview

Table 1 – IMM_MES_InterfaceType Definition

Attribute	Value				
BrowseName	IMM_MES_InterfaceType				
IsAbstract	False				
References	Node Class	BrowseName	Data Type	Type Definition	Other
Subtype of 0:BaseObjectType defined in OPC UA Part 5					
0:HasComponent	Object	MachineInformation		3:MachineInformationType	M
0:HasComponent	Object	MachineConfiguration		3:MachineConfigurationType	M
0:HasComponent	Object	MachineMESConfiguration		3:MachineMESConfigurationType	M
0:HasComponent	Object	MachineStatus		3:MachineStatusType	M
0:HasComponent	Object	MachineMESStatus		3:MachineMESStatusType	M
0:HasComponent	Object	PowerUnits		3:PowerUnitsType	M
0:HasComponent	Object	Moulds		3:MouldsType	M
0:HasComponent	Object	InjectionUnits		InjectionUnitsType	M
0:HasComponent	Object	Jobs		3:JobsType	O
0:HasComponent	Object	ProductionDatasetManagement		3:ProductionDatasetManagementType	O
0:GeneratesEvent	ObjectType	3:LogbookEventType			
0:GeneratesEvent	ObjectType	3:CycleParametersEventType			

The *BrowseName* of the object instance shall be "IMM_<Manufacturer>_<SerialNumber>"

Example: "IMM_ARBURG_0123456"

NOTE: The namespace of this *BrowseName* is the local server URI with namespace index 1 or a vendor specific namespace with server specific namespace index (see Table 14). The *BrowseNames* of the nodes below are in the namespace of the specification where used Type is defined.

Examples:

BrowseName	Namespace	Namespace index	Remarks
IMM_ARBURG_0123456	Local Server URI or vendor specific namespace	1 or server specific	OPC 40077 only defines the <i>IMM_MES_InterfaceType</i> . The instance is generated in the local server

↓

MachineStatus	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/	server specific	The object <i>MachineStatus</i> is a child <i>IMM_MES_InterfaceType</i> which is defined in OPC 40077
---------------	-----------------------------------------------------	-----------------	-------------------------------------------------------------------------------------------------------

↓

MachineMode	http://opcfoundation.org/UA/PlasticsRubber/GenrealTypes/	server specific	The variable <i>MachineMode</i> is a child of <i>MachineStatusType</i> which is defined in OPC 40083.
-------------	----------------------------------------------------------	-----------------	-------------------------------------------------------------------------------------------------------

BrowseName	Namespace	Namespace index	Remarks
IMM_ARBURG_0123456	Local Server URI or vendor specific namespace	1 or server specific	OPC 40077 only defines the <i>IMM_MES_InterfaceType</i> . The instance is generated in the local server

↓

MachineInformation	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/	server specific	The object <i>MachineInformation</i> is a child of <i>IMM_MES_InterfaceType</i> which is defined in OPC 4007
--------------------	-----------------------------------------------------	-----------------	--------------------------------------------------------------------------------------------------------------

↓

Manufacturer	http://opcfoundation.org/UA/DI/	server specific	The variable <i>Manufacturer</i> is a child of the <i>DeviceType</i> (supertype of <i>MachineInformationType</i>) which is defined in OPC UA DI.
--------------	---------------------------------	-----------------	---------------------------------------------------------------------------------------------------------------------------------------------------

BrowseName	Namespace	Namespace index	Remarks
IMM_ARBURG_0123456	Local Server URI or vendor specific namespace	1 or server specific	OPC 4007 only defines the <i>IMM_MES_InterfaceType</i> . The instance is generated in the local server

↓

ProductionDataset Management	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/	server specific	The object <i>ProductionDataset Management</i> is a child of <i>IMM_MES_InterfaceType</i> which is defined in OPC 4007
------------------------------	-----------------------------------------------------	-----------------	------------------------------------------------------------------------------------------------------------------------

↓

ProductionDataset Transfer	http://opcfoundation.org/UA/PlasticsRubber/GenrealTypes/	server specific	The object <i>ProductionDataset Transfer</i> is a child of <i>ProductionDataset ManagementType</i> which is defined in OPC 4083
----------------------------	----------------------------------------------------------	-----------------	---------------------------------------------------------------------------------------------------------------------------------

↓

CloseAndCommit	http://opcfoundation.org/UA/	0	The method <i>CloseAndCommit</i> is a child of <i>ProductionDatasetTransfer</i> which has the <i>TemporaryFile TransferType</i> as type definition which is defined in OPC UA Part 5.
----------------	------------------------------	---	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

BrowseName	Namespace	Namespace index	Remarks
IMM_ARBURG_0123456	Local Server URI or vendor specific namespace	1 or server specific	OPC 40077 only defines the <i>IMM_MES_InterfaceType</i> . The instance is generated in the local server
↓			
Moulds	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/	server specific	The object <i>Moulds</i> is a child of <i>IMM_MES_InterfaceType</i> which is defined in OPC 40077
↓			
Mould_1	Local Server URI or vendor specific namespace	1 or server specific	The objects for the moulds are modelled as <i>OptionalPlaceholder</i> . The instances are server specific
↓			
Id	http://opcfoundation.org/UA/PlasticsRubber/GenrealTypes/	server specific	The property <i>Id</i> is a child of <i>MouldType</i> which is defined in OPC 40083.

8 MachineInformation

The *MachineInformation Object* provides general information on the injection moulding machine. The *MachineInformationType* is defined in OPC 40083.

8.1 DeviceClass

The *DeviceClass Property* in the *MachineInformation Object* shall have the value "Injection Moulding Machine".

8.2 LogbookEvents

The logbook events defined in OPC 40083 can be used. The supported logbook event types shall be listed in *MachineInformation.SupportedLogbookEvents*.

9 MachineConfiguration

The *MachineConfiguration Object* represents the current configuration of the injection moulding machine. The *MachineConfigurationType* is defined in OPC 40083.

10 MachineMESConfiguration

The *MachineMESConfiguration Object* represents the current configuration of the injection moulding machine related to the MES. The *MachineMESConfigurationType* is defined in OPC 40083.

11 MachineStatus

The *MachineStatus Object* represents the current status of the injection moulding machine. The *MachineStatusType* is defined in OPC 40083.

12 MachineMESStatus

The *MachineMESStatus Object* represents the current status of the injection moulding machine related to the MES. The *MachineMESStatusType* is defined in OPC 40083.

The *Object* can generate *Events* of *MessageConditionType* which includes a *Classification Property*. For injection moulding machines, the *IMMMessageClassificationEnumeration* defined in Table 2 shall be used (also in the related logbook event).

Table 2 – IMMMessageClassificationEnumeration Definition

Name	Value	Description	
		Indication which machine part has caused the message	
		machine	part
OTHER	0	This state is used if none of the other entries below apply.	
IMM_INJECTION_UNIT	100	Injection moulding machine	injection unit
IMM_CLAMPING_UNIT	101	Injection moulding machine	clamping unit
IMM_HARDWARE	102	Injection moulding machine	hardware
IMM_COMPRESSED_AIR_CONTROL	103	Injection moulding machine	compressed air control
IMM_MACHINE_MONITORING	104	Injection moulding machine	machine monitoring
IMM_MOULD	105	Injection moulding machine	mould
IMM_EJECTOR	106	Injection moulding machine	ejector
IMM_CORE_PULL	107	Injection moulding machine	core pull
IMM_TABLE	108	Injection moulding machine	table
IMM_INJECTION_PROGRAM	109	Injection moulding machine	injection program
IMM_HYDRAULIC_TEMPERATURE_CONTROL	110	Injection moulding machine	temperature control hydraulic
IMM_CYLINDER_TEMPERATURE_CONTROL	111	Injection moulding machine	cylinder temperature control
IMM_MOULD_TEMPERATURE_CONTROL	112	Injection moulding machine	mould temperature control
IMM_HOT_RUNNER	113	Injection moulding machine	hot runner
IMM_INTERFACES	114	Injection moulding machine	interfaces
IMM_MEASURING_SYSTEM	115	Injection moulding machine	measuring system
IMM_ROBOTIC_SYSTEM_INTERFACE	116	Injection moulding machine	robotic system interface
IMM_SPECIAL_PURPOSE_SIGNALS	117	Injection moulding machine	special purpose signals
IMM_REAL_TIME_ETHERNET_SYSTEM	118	Injection moulding machine	real-time Ethernet system (Varan, Ethercat, ProfiNET, ...)
IMM_MACHINE_CONTROLLER	119	Injection moulding machine	machine controller
IMM_SOFTWARE_MONITORING	120	Injection moulding machine	software monitoring
PERIPHERAL_EXTERNAL_DEVICE_INTERFACE	200	peripheral equipment	external device interface
PERIPHERAL_TEMPERATURE_CONTROL_UNIT	201	peripheral equipment	temperature control unit
PERIPHERAL_ROBOTICS_SYSTEM	202	peripheral equipment	robotics system
PERIPHERAL_LSR	203	peripheral equipment	LSR (Liquid Silicone Rubber)
PERIPHERAL_STRIPPER_UNIT	204	peripheral equipment	stripper unit
PERIPHERAL_DRYER	205	peripheral equipment	dryer
PERIPHERAL_CONVEYOR_BELT	206	peripheral equipment	conveyor belt
PERIPHERAL_SORTER_UNIT	207	peripheral equipment	sorter unit
PERIPHERAL_COLOURING_UNIT	208	peripheral equipment	colouring unit
PERIPHERAL_FEEDING	209	peripheral equipment	feeding unit
PERIPHERAL_EXTERNAL_ALARMS	210	peripheral equipment	external alarms
PERIPHERAL_VACUUM_CONTROL	211	peripheral equipment	vacuum control
PERIPHERAL_PRINTER_INTERFACE	212	peripheral equipment	printer interface
OPERATION_QUALITY_MONITORING	300	operation	quality monitoring
OPERATION_MANUAL_OPERATION	301	operation	manual operation
OPERATION_EMERGENCY_STOP	302	operation	emergency stop
OPERATION_JOB_STATUS	303	operation	production status

13 PowerUnits

The *PowerUnits Object* is a container for the power unit(s) of the injection moulding machines. The *PowerUnitsType* is defined in OPC 40083.

14 Moulds

The *Moulds Object* is a container for the mould(s) of the injection moulding machines. The *MouldsType* is defined in OPC 40083.

15 InjectionUnitsType

This *ObjectType* is a container for the injection unit(s) (see container concept in OPC 40083). It is formally defined in Table 3.

Table 3 – InjectionUnitsType Definition

Attribute	Value				
BrowseName	InjectionUnitsType				
IsAbstract	False				
References	Node Class	BrowseName	DataType	TypeDefinition	Other
Subtype of 0:BaseObjectType defined in OPC UA Part 5					
0:HasProperty	Variable	0:NodeVersion	0:String	0:PropertyType	M, RO
0:HasComponent	Object	InjectionUnit_<Nr>		InjectionUnitType	OP
0:GeneratesEvent	ObjectType	0:GeneralModelChangeEvent			

When instances for injection units are created, the *BrowseNames* shall be “InjectionUnit_<Nr>” (starting with 1).

16 InjectionUnitType ObjectType Definition

16.1 Overview

This *ObjectType* represents the description and status of the injection unit(s). It is formally defined in Table 4.

Table 4 – InjectionUnitType Definition

Attribute	Value				
BrowseName	InjectionUnitType				
IsAbstract	False				
References	Node Class	BrowseName	DataType	TypeDefinition	Other
Subtype of 0:BaseObjectType defined in OPC UA Part 5					
0:HasProperty	Variable	Index	UInt32	0:PropertyType	M, RO
0:HasProperty	Variable	BarrelId	String	0:PropertyType	M, RO
0:HasProperty	Variable	IsPresent	Boolean	0:PropertyType	M, RO
0:HasProperty	Variable	InProduction	Boolean	0:PropertyType	M, RO
0:HasProperty	Variable	ScrewId	String	0:PropertyType	O, RO
0:HasComponent	Variable	ScrewDiameter	Double	0:AnalogItemType	O, RO
0:HasComponent	Variable	ScrewVolume	Double	0:AnalogItemType	O, RO
0:HasComponent	Variable	MaxScrewStroke	Double	0:AnalogItemType	O, RO
0:HasComponent	Object	TemperatureZones		3:TemperatureZonesType	M

16.2 Index

The *Index Property* gives the number of the injection unit.

16.3 BarrelId

The *BarrelId Property* provides the Id (e.g. serial number) of the barrel.

16.4 IsPresent

It is allowed to create instances of all *InjectionUnits* that **can** be connected to the machine (e.g. due to available connectors) to avoid dynamical creation of objects. The *IsPresent Property* provides information if the *InjectionUnit* is physically installed on the injection machines.

16.5 InProduction

The *InProduction Property* provides information if the *InjectionUnit* is used in the current running production.

16.6 ScrewId

The *ScrewId Property* represents the Id of the screw installed in the *InjectionUnit*.

16.7 ScrewDiameter

ScrewDiameter represents the diameter of the screw installed in the *InjectionUnit*.

16.8 ScrewVolume

ScrewVolume represents the volume of the screw installed in the *InjectionUnit*.

16.9 MaxScrewStroke

MaxScrewStroke represents the maximum stroke of the screw installed in the *InjectionUnit*.

16.10 TemperatureZones

This *Object* is a container for the barrel temperature zones of the injection unit. The *TemperatureZonesType* is formally defined in OPC 40083. Inside the container the *BarrelTemperatureZoneType* shall be used.

17 Jobs

The *Jobs Object* is used for managing production jobs on the machine and for information on their status including process parameters (temperatures, pressures...). The *JobsType* is defined in OPC 40083.

17.1 Types, methods and events for cyclic jobs

As IMM have a cyclic production, the Types, Methods and Events defined in OPC 40083 for cyclic jobs shall be used:

Table 5 – Types, methods and events for cyclic jobs

Type/Method/Event	SubType/extended Methods to be used for OPC 40077
JobInformationType	CyclicJobInformationType
SendJobList	SendCyclicJobList
RequestJobListEventType	RequestCyclicJobListEventType
ActiveJobValuesType	ActiveCyclicJobValuesType

17.2 Optional properties in CyclicJobInformationType

In the *CyclicJobInformationType*, as defined in OPC 40083, the Properties *MouldId* and *NumCavities* have the *ModellingRule* optional. When the *CyclicJobInformationType* is used for a OPC 40077 interface, these *Properties* become mandatory and shall be filled by the server.

17.3 InjectionUnitCycleParametersType

The *InjectionUnitCycleParametersType* represents information on the production cycle related to an injection unit. This type is used to extend the *CycleParametersEventType* defined in OPC 40083.

The *BrowseNames* of the additional objects shall be “InjectionUnitCycleParameters_<Nr>” (starting with 1)

Table 6 – Example of an event type derived from CycleParametersEventType with two moulds and two injection units

Attribute	Value				
BrowseName	ExampleCycleParametersEventType				
IsAbstract	false				
References	Node Class	BrowseName	DataType	TypeDefinition	Other
Subtype of 3:CycleParametersEventType defined in OPC 40083					
0:HasComponent	Object	MouldCycleParameters_1		Example1MouldCycleParametersType	M
0:HasComponent	Object	MouldCycleParameters_2		Example2MouldCycleParametersType	M
0:HasComponent	Object	InjectionUnitCycleParameters_1		Example1InjectionUnitCycleParametersType	M
0:HasComponent	Object	InjectionUnitCycleParameters_2		Example2InjectionUnitCycleParametersType	M

The Types *Example1MouldCycleParametersType* and *Example2MouldCycleParametersType* used in the example are subtypes of the *MouldCycleParametersType* defined in OPC 40083. The Types

Example1InjectionUnitCycleParametersType and *Example2InjectionUnitCycleParametersType* are subtypes of the *InjectionUnitCycleParametersType* which is formally defined Table 7.

Table 7 – InjectionUnitCycleParametersType Definition

Attribute	Value				
BrowseName	InjectionUnitCycleParametersType				
IsAbstract	True				
References	Node Class	BrowseName	Data Type	Type Definition	Other
Subtype of 0:BaseObjectType defined in OPC UA Part 5					
0:HasProperty	Variable	Index	0:UInt32	0:PropertyType	M
0:HasComponent	Variable	CushionVolume	0:Double	0:AnalogItemType	M
0:HasComponent	Variable	CushionStroke	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationVolume	0:Double	0:AnalogItemType	M
0:HasComponent	Variable	DecompressionVolumeBeforePlastification	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	DecompressionVolumeAfterPlastification	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	HydraulicPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	SpecificPressureMaximum	0:Double	0:AnalogItemType	M
0:HasComponent	Variable	PlastificationRotationalSpeedMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationRotationalSpeedAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationCircumferentialSpeedMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationCircumferentialSpeedAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	InjectionSpeedMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	InjectionSpeedAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	TransferVolume	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	TransferStroke	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	HoldHydraulicPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	HoldHydraulicPressureAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	HoldSpecificPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	HoldSpecificPressureAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	CavityPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationHydraulicPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationHydraulicPressureAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationSpecificPressureMaximum	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	PlastificationSpecificPressureAverage	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	TransferHydraulicPressure	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	TransferSpecificPressure	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	TransferCavityPressure	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	BackPressure	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	InjectionTime	0:Duration	0:BaseDataVariableType	M
0:HasComponent	Variable	DosingTime	0:Duration	0:BaseDataVariableType	M
0:HasComponent	Variable	FlowIndex	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	InjectionStartPosition	0:Double	0:AnalogItemType	O
0:HasComponent	Variable	VPChangeOverPosition	0:Double	0:AnalogItemType	O

Table 8 contains a description of the parameters listed in Table 7.

Table 8 – Description of cycle parameters

Parameter	Description
Index	Index of the InjectionUnit (see 16.2)
CushionVolume	Material volume remained in front of the screw after injection and holding pressure
CushionStroke	Stroke position at cushion
PlastificationVolume	Volume dosed by the machine for the next injection shot
DecompressionVolumeBeforePlastification	Decompression before plastification is the movement of the screw in the opposite direction to injection
DecompressionVolumeAfterPlastification	Decompression after plastification is the movement of the screw in the opposite direction to injection
HydraulicPressureMaximum	Maximum pressure in the hydraulic cylinder
SpecificPressureMaximum	Pressure in front of the screw tip
PlastificationRotationalSpeedMaximum	Maximum plastification speed of the injection unit (RPM)
PlastificationRotationalSpeedAverage	Average plastification speed of the injection unit (RPM)
PlastificationCircumferentialSpeedMaximum	Maximum screw circumferential speed for plastification (e.g. mm/s)
PlastificationCircumferentialSpeedAverage	Average screw circumferential speed for plastification (e.g. mm/s)
InjectionSpeedMaximum	Maximum injection speed (e.g. mm/s)
InjectionSpeedAverage	Average injection speed (e.g. mm/s)
TransferVolume	Switch-over point to the holding pressure via volume
TransferStroke	Switch-over point to the holding pressure via stroke
HoldHydraulicPressureMaximum	Maximum holding pressure in the hydraulic cylinder
HoldHydraulicPressureAverage	Average holding pressure in the hydraulic cylinder
HoldSpecificPressureMaximum	Maximum holding pressure in front of the screw
HoldSpecificPressureAverage	Average holding pressure in front of the screw
CavityPressureMaximum	Maximum pressure during the injection process in the cavity or mould
PlastificationHydraulicPressureMaximum	Maximum plastification pressure in cylinder
PlastificationHydraulicPressureAverage	Average plastification pressure in cylinder
PlastificationSpecificPressureMaximum	Maximum plastification pressure in front of the screw tip
PlastificationSpecificPressureAverage	Average plastification pressure in front of the screw tip
TransferHydraulicPressure	Hydraulic pressure in the cylinder during switch-over to the holding pressure
TransferSpecificPressure	Pressure in front of the screw tip during switch-over to the holding pressure
TransferCavityPressure	Cavity pressure in the mould during switch-over to the holding pressure
BackPressure	Back pressure is the melt-pressure against the screw movement during dosage
InjectionTime	Time required to fill the cavity or mould
DosingTime	Time to melt-up the plastic granulates and feed the melt for the next injection shot to the front of the screw
FlowIndex	Integral of the injection pressure over the injection time as measure for the injection work
InjectionStartPosition	Start position of the injection
VPChangeOverPosition	Screw position at switching between injection (V) and holding pressure (P)

The *InjectionUnitCycleParametersType* is abstract and the OPC server of the machine shall create a derived type with the additional objects of *TemperatureZoneCycleParametersType* (defined in OPC 40083) for the temperature zones of the barrel. The *BrowseNames* of the objects shall be “BarrelTemperatureZoneCycleParameters_<Nr>” (starting with 1 for each injection unit).

18 ProductionDatasetManagement

The *ProductionDatasetManagement Object* is used for managing production datasets which are files containing the configuration of a machine. The *ProductionDatasetManagementType* is defined in OPC 40083.

19 Profiles and Conformance Units

This chapter defines the corresponding profiles and conformance units for the OPC UA Information Model for OPC 40077. *Profiles* are named groupings of conformance units. Facets are profiles that will be combined with other *Profiles* to define the complete functionality of an OPC UA *Server* or *Client*. The following tables specify the facets available for *Servers* that implement the OPC 40077 Information Model companion specification.

NOTE: The names of the supported profiles are available in the *Server Object* under *ServerCapabilities.ServerProfileArray*

Table 9 lists all Profiles defined in this document and defines their URIs.

Table 9 – Profile URIs for OPC 40077

Profile	URI
OPC 40077 Basic Server Profile	http://opcfoundation.org/UA-Profile/PlasticsRubber/IMM2MES/Server/Basic
OPC 40077 Jobs Server Facet	http://opcfoundation.org/UA-Profile/PlasticsRubber/IMM2MES/Server/Jobs
OPC 40077 ProductionDatasetManagement Server Facet	http://opcfoundation.org/UA-Profile/PlasticsRubber/IMM2MES/Server/ProductionDatasetManagement

Table 10 – OPC 40077 Basic Server Profile Definition

Conformance Unit	Description	Optional/ Mandatory
OPC 40077 Basic	Support of IMM_MES_InterfaceType and all mandatory child elements giving information on the injection moulding machine itself, the current configuration and status and the installed injection units, mould and power units.	M
Profile		
ComplexType Server Facet (defined in OPC UA Part 7)		M
Standard Event Subscription Server Facet (defined in OPC UA Part 7)		M
Method Server Facet (defined in OPC UA Part 7)		M
BaseDevice_Server_Facet (defined in OPC UA Part 100)		M

Table 11 – OPC 40077 Jobs Server Facet Definition

Conformance Unit	Description	Optional/ Mandatory
OPC 40077 Jobs	Support of JobsType (defined in OPC 40083) for the status and management of jobs as well as providing cycle parameters (support of CycleParametersEventType)	M

Table 12 – OPC 40077 ProductionDatasetManagement Server Facet Definition

Conformance Unit	Description	Optional/ Mandatory
OPC 40077 ProductionDatasetManagement	Support of ProductionDatasetManagementType (defined in OPC 40083) for the management and transfer of production datasets between MES and IMM	M

20 Namespaces

20.1 Namespace Metadata

Table 13 defines the namespace metadata for this specification. The *Object* is used to provide version information for the namespace and an indication about static *Nodes*. Static *Nodes* are identical for all *Attributes* in all *Servers*, including the *Value Attribute*. See Part5 for more details.

The information is provided as *Object* of type *NamespaceMetadataType*. This *Object* is a component of the *Namespaces Object* that is part of the *Server Object*. The *NamespaceMetadataType ObjectType* and its *Properties* are defined in Part5.

The version information is also provided as part of the *ModelTableEntry* in the *UANodeSet XML* file. The *UANodeSet XML* schema is defined in Part 6.

Table 13 – NamespaceMetadata Object for this Specification

Attribute	Value		
BrowseName	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/		
References	BrowseName	DataType	Value
0:HasProperty	NamespaceUri	String	http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/
0:HasProperty	NamespaceVersion	String	1.01
0:HasProperty	NamespacePublicationDate	DateTime	2020-06-01 00:00:00
0:HasProperty	IsNamespaceSubset	Boolean	False
0:HasProperty	StaticNodeIdTypes	IdType[]	{Numeric}
0:HasProperty	StaticNumericNodeIdRange	NumericRange[]	Null
0:HasProperty	StaticStringNodeIdPattern	String	Null

20.2 Handling of OPC UA Namespaces

Namespaces are used by OPC UA to create unique identifiers across different naming authorities. The *Attributes* *NodeId* and *BrowseName* are identifiers. A *Node* in the UA *AddressSpace* is unambiguously identified using a *NodeId*. Unlike *NodeIds*, the *BrowseName* cannot be used to unambiguously identify a *Node*. Different *Nodes* may have the same *BrowseName*. They are used to build a browse path between two *Nodes* or to define a standard *Property*.

Servers may often choose to use the same namespace for the *NodeId* and the *BrowseName*. However, if they want to provide a standard *Property*, its *BrowseName* shall have the namespace of the standards body although the namespace of the *NodeId* reflects something else, for example the *EngineeringUnits Property*. All *NodeIds* of *Nodes* not defined in this document shall not use the standard namespaces.

Table 14 provides a list of mandatory and optional namespaces used in an OPC 40077 OPC UA Server.

Table 14 – Namespaces used in an OPC 40077 Server

NamespaceURI	Description	Use
http://opcfoundation.org/UA/	Namespace for <i>NodeIds</i> and <i>BrowseNames</i> defined in the OPC UA specification. This namespace shall have namespace index 0.	Mandatory
Local Server URI	Namespace for nodes defined in the local server. This may include types and instances used in a device represented by the server. This namespace shall have namespace index 1.	Mandatory
http://opcfoundation.org/UA/DI/	Namespace for <i>NodeIds</i> and <i>BrowseNames</i> defined in OPC UA Part 100. The namespace index is server specific.	Mandatory
http://opcfoundation.org/UA/PlasticsRubber/GeneralTypes/	Namespace for <i>NodeIds</i> and <i>BrowseNames</i> defined in OPC 40083. The namespace index is server specific.	Mandatory
http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/	Namespace for <i>NodeIds</i> and <i>BrowseNames</i> defined in this specification. The namespace index is server specific.	Mandatory
Vendor specific types and instances	A server may provide vendor specific types like types derived from <i>MachineType</i> or <i>MachineStatusType</i> or vendor specific instances of devices in a vendor specific namespace.	Optional

Table 15 provides a list of namespaces and their index used for *BrowseNames* in this specification. The default namespace of this specification is not listed since all *BrowseNames* without prefix use this default namespace.

Table 15 – Namespaces used in this specification

NamespaceURI	Namespace Index	Example
http://opcfoundation.org/UA/	0	0:NodeVersion
http://opcfoundation.org/UA/DI/	2	2:DeviceClass
http://opcfoundation.org/UA/PlasticsRubber/GeneralTypes/	3	3:MachineInformationType

Annex A (normative)

OPC 40077 Namespace and mappings

A.1 Namespace and identifiers for OPC 40077 Information Model

This appendix defines the numeric identifiers for all of the numeric *NodeIds* defined in this specification. The identifiers are specified in a CSV file with the following syntax:

<SymbolName>, <Identifier>, <NodeClass>

Where the *SymbolName* is either the *BrowseName* of a *Type Node* or the *BrowsePath* for an *Instance Node* that appears in the specification and the *Identifier* is the numeric value for the *NodeId*.

The *BrowsePath* for an *Instance Node* is constructed by appending the *BrowseName* of the instance *Node* to the *BrowseName* for the containing instance or type. An underscore character is used to separate each *BrowseName* in the path. Let's take for example, the *MachineInformationType ObjectType Node* which has the *ControllerName Property*. The **Name** for the *ControllerName InstanceDeclaration* within the *MachineInformationType* declaration is: *MachineInformationType_ControllerName*.

The *NamespaceUri* for all *NodeIds* defined here is <http://opcfoundation.org/UA/PlasticsRubber/IMM2MES/>

The CSV released with this version of the specification can be found here:

- <http://www.opcfoundation.org/UA/schemas/PlasticsRubber/IMM2MES/1.01/NodeIds.csv>

NOTE: The latest CSV that is compatible with this version of the specification can be found here:

- <http://www.opcfoundation.org/UA/schemas/PlasticsRubber/IMM2MES/NodeIds.csv>

A computer processible version of the complete Information Model defined in this specification is also provided. It follows the XML Information Model schema syntax defined in Part 6.

The Information Model Schema released with this version of the specification can be found here:

- <http://www.opcfoundation.org/UA/schemas/PlasticsRubber/IMM2MES/1.01/Opc.Ua.PlasticsRubber.IMM2MES.NodeSet2.xml>

NOTE: The latest Information Model schema that is compatible with this version of the specification can be found here:

- <http://www.opcfoundation.org/UA/schemas/PlasticsRubber/IMM2MES/Opc.Ua.PlasticsRubber.IMM2MES.NodeSet2.xml>
-