



SLOVENSKI STANDARD

SIST EN 62541-10:2015

01-september-2015

Nadomešča:

SIST EN 62541-10:2012

Enotna arhitektura OPC - 10. del: Programi (IEC 62541-10:2015)

OPC Unified Architecture - Part 10:Programs (IEC 62541-10:2015)

OPC Unified Architecture - Teil 10: Programme (IEC 62541-10:2015)

Architecture unifiée OPC - Partie 10: Programmes (IEC 62541-10:2015)

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Ta slovenski standard je istoveten z: EN 62541-10:2015

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4de/sist-en-62541-10-2015>

ICS:

25.040.40	Merjenje in krmiljenje industrijskih postopkov	Industrial process measurement and control
35.240.50	Uporabniške rešitve IT v industriji	IT applications in industry

SIST EN 62541-10:2015

en,fr,de

iTeh STANDARD PREVIEW
(standards.iteh.ai)

SIST EN 62541-10:2015

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>

EUROPEAN STANDARD

EN 62541-10

NORME EUROPÉENNE

EUROPÄISCHE NORM

April 2015

ICS 25.040.40; 35.100

Supersedes EN 62541-10:2012

English Version

**OPC unified architecture - Part 10: Programs
(IEC 62541-10:2015)**Architecture unifiée OPC - Partie 10: Programmes
(IEC 62541-10:2015)OPC Unified Architecture - Teil 10: Programme
(IEC 62541-10:2015)

This European Standard was approved by CENELEC on 2015-04-14. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Foreword

The text of document 65E/383/FDIS, future edition 2 of IEC 62541-10, prepared by SC 65E "Devices and integration in enterprise systems" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 62541-10:2015.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2016-01-14
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2018-04-14

This document supersedes EN 62541-10:2012.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

iTeh STANDARD PREVIEW (standards.iteh.ai)

The text of the International Standard IEC 62541-10:2015 was approved by CENELEC as a European Standard without any modification. [SIST EN 62541-10:2015](https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015)

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>

Annex ZA (normative)

Normative references to international publications with their corresponding European publications

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE 1 When an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u> ¹⁾
IEC 62541-3	2015	OPC Unified Architecture -- Part 3: Address Space Model	EN 62541-3	2015 ¹⁾
IEC 62541-4	2015	OPC Unified Architecture -- Part 4: Services	EN 62541-4	2015 ¹⁾
IEC 62541-5	2015	OPC Unified Architecture -- Part 5: Information Model	EN 62541-5	2015 ¹⁾
IEC 62541-7	-	OPC Unified Architecture -- Part 7: Profiles	EN 62541-7	-
IEC/TR 62541-1	-	OPC unified architecture -- Part 1: Overview and concepts	CLC/TR 62541-1	-

iTeH STANDARD PREVIEW
(standards.iteh.ai)

[SIST EN 62541-10:2015](https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015)

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>

1) To be published.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

SIST EN 62541-10:2015

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>



INTERNATIONAL STANDARD

NORME INTERNATIONALE



**OPC Unified Architecture –
Part 10: Programs**

STANDARD PREVIEW
(standards.iteh.ai)

**Architecture unifiée OPC –
Partie 10: Programmes**

[SIST EN 62541-10:2015](https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015)
<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040.40; 35.100

ISBN 978-2-8322-2274-4

**Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

CONTENTS

FOREWORD	4
1 Scope	6
2 Normative references	6
3 Terms, definitions and conventions	6
3.1 Terms and definitions	6
3.2 Abbreviations	7
4 Concepts	7
4.1 General	7
4.2 Programs	8
4.2.1 Overview	8
4.2.2 Security considerations	9
4.2.3 Program Finite State Machine	9
4.2.4 Program states	10
4.2.5 State transitions	11
4.2.6 Program state transition stimuli	11
4.2.7 Program Control Methods	11
4.2.8 Program state transition effects	12
4.2.9 Program result data	12
4.2.10 Program lifetime	13
5 Model	13
5.1 General	13
5.2 ProgramType	14
5.2.1 Overview	14
5.2.2 ProgramType Properties	16
5.2.3 ProgramType components	16
5.2.4 ProgramType causes (Methods)	21
5.2.5 ProgramType effects (Events)	23
5.2.6 AuditProgramTransitionEventType	25
5.2.7 FinalResultData	26
5.2.8 ProgramDiagnostic DataType	26
5.2.9 ProgramDiagnosticType VariableType	27
Annex A (informative) Program example	28
A.1 Overview	28
A.2 DomainDownload Program	28
A.2.1 General	28
A.2.2 DomainDownload states	29
A.2.3 DomainDownload transitions	30
A.2.4 DomainDownload Methods	30
A.2.5 DomainDownload Events	31
A.2.6 DomainDownload model	31
Figure 1 – Automation facility control	8
Figure 2 – Program illustration	9
Figure 3 – Program states and transitions	10
Figure 4 – Program Type	14

Figure 5 – Program FSM References	17
Figure 6 – ProgramType causes and effects	21
Figure A.1 – Program example	28
Figure A.2 – DomainDownload state diagram	29
Figure A.3 – DomainDownloadType partial state model	35
Figure A.4 – Ready To Running model	38
Figure A.5 – Opening To Sending To Closing model	40
Figure A.6 – Running To Suspended model	41
Figure A.7 – Suspended To Running model	42
Figure A.8 – Running To Halted – Aborted model	43
Figure A.9 – Suspended To Aborted model	44
Figure A.10 – Running To Completed model	45
Figure A.11 – Sequence of operations	46
Table 1 – Program Finite State Machine	9
Table 2 – Program states	10
Table 3 – Program state transitions	11
Table 4 – Program Control Methods	12
Table 5 – ProgramType	15
Table 6 – Program states	17
Table 7 – Program transitions	19
Table 8 – ProgramType causes	22
Table 9 – ProgramTransitionEventType	23
Table 10 – ProgramTransitionEvents	24
Table 11 – AuditProgramTransitionEventType	25
Table 12 – ProgramDiagnosticDataType structure	26
Table 13 – ProgramDiagnosticDataType definition	26
Table 14 – ProgramDiagnosticType VariableType	27
Table A.1 – DomainDownload states	30
Table A.2 – DomainDownload Type	32
Table A.3 – Transfer State Machine Type	32
Table A.4 – Transfer State Machine – states	33
Table A.5 – Finish State Machine Type	33
Table A.6 – Finish State Machine – states	34
Table A.7 – DomainDownload Type Property Attributes variable values	34
Table A.8 – Additional DomainDownload transition types	36
Table A.9 – Start Method additions	38
Table A.10 – StartArguments	39
Table A.11 – IntermediateResults Object	40
Table A.12 – Intermediate result data Variables	41
Table A.13 – FinalResultData	44
Table A.14 – Final result Variables	45

INTERNATIONAL ELECTROTECHNICAL COMMISSION

OPC UNIFIED ARCHITECTURE –

Part 10: Programs

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as “IEC Publication(s)”). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-32db184cc387/iec-62541-10-2015>
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62541-10 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition published in 2012. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- a) Based on NIST review, security considerations have been included as 4.2.2;
- b) Fixed the definition of the Program Diagnostic Type into a data type (5.2.8) and added missing data type for the Program Diagnostic Variable in the ProgramType in Table 5.
- c) Corrected the BrowseName of the audit events for Program Transitions in Table 7.

The text of this standard is based on the following documents:

FDIS	Report on voting
65E/383/FDIS	65E/409/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts of the IEC 62541 series, published under the general title *OPC Unified Architecture*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

iTeh STANDARD PREVIEW

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

<https://standards.iteh.ai/catalog/standards/sist/c6bae7e1-4b3e-476a-b650-83ab8118e4dc/sist-en-62541-10-2015>

OPC UNIFIED ARCHITECTURE –

Part 10: Programs

1 Scope

This part of IEC 62541 is part of the overall OPC Unified Architecture (OPC UA) standard series and defines the information model associated with *Programs*. This includes the description of the *NodeClasses*, standard *Properties*, *Methods* and *Events* and associated behaviour and information for *Programs*.

The complete address space model including all *NodeClasses* and *Attributes* is specified in IEC 62541-3. The services such as those used to invoke the *Methods* used to manage *Programs* are specified in IEC 62541-4.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC TR 62541-1, *OPC Unified Architecture – Part 1: Overview and Concepts*

IEC 62541-3:2015, *OPC Unified Architecture – Part 3: Address Space Model*

IEC 62541-4:2015, *OPC Unified Architecture – Part 4: Services*

IEC 62541-5:2015, *OPC Unified Architecture – Part 5: Information Model*

IEC 62541-7, *OPC Unified Architecture – Part 7: Profiles*

3 Terms, definitions and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC TR 62541-1, IEC 62541-3, as well as the following apply.

3.1.1 function

programmatic task performed by a server or device, usually accomplished by computer code execution

3.1.2 Finite State Machine

sequence of states and valid state transitions along with the causes and effects of those state transitions that define the actions of a *Program* in terms of discrete stages

3.1.3 ProgramType

type definition of a *Program* and is a subtype of the *FiniteStateMachineType*

3.1.4**Program Control Method**

Method having specific semantics designed for the control of a *Program* by causing a state transition

3.1.5**Program Invocation**

unique *Object* instance of a *Program* existing on a *Server*

Note 1 to entry: A *Program Invocation* is distinguished from other *Object* instances of the same *ProgramType* by the object node's unique browse path.

3.2 Abbreviations

DA	Data Access
FSM	Finite State Machine
HMI	Human Machine Interfaces
PCM	Program Control Method
PGM	Program
PI	Program Invocation
UA	Unified Architecture

4 Concepts

iTeh STANDARD PREVIEW
(standards.iteh.ai)

4.1 General

Integrated automation facilities manage their operations through the exchange of data and the coordinated invocation of system functions as illustrated in Figure 1. *Services* are required to perform the data exchanges and to invoke the functions that constitute system operation. These functions may be invoked through Human Machine Interfaces, cell controllers, or other supervisory control and data acquisition type systems. OPC UA defines *Methods* and *Programs* as an interoperable way to advertise, discover, and request these functions. They provide a normalizing mechanism for the semantic description, invocation, and result reporting of these functions. Together *Methods* and *Programs* complement the other OPC UA *Services* and *ObjectTypes* to facilitate the operation of an automation environment using a client-server hierarchy.

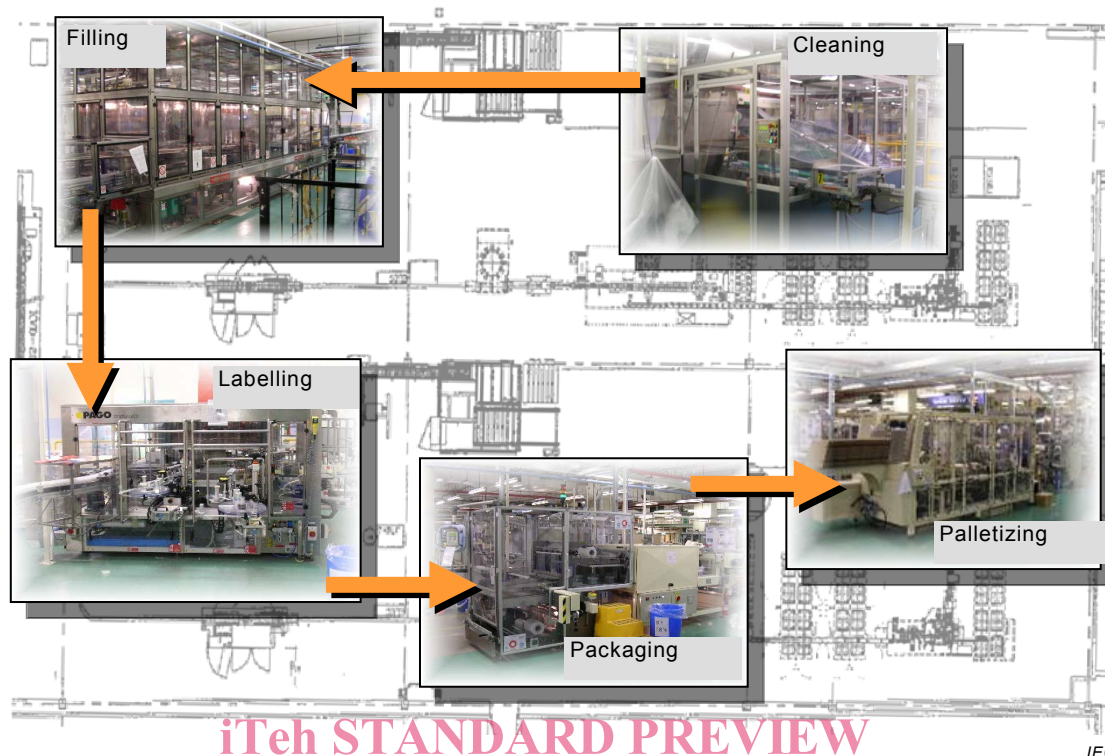


Figure 1 – Automation facility control

Methods and *Programs* model functions typically have different scopes, behaviours, lifetimes, and complexities in *OPC Servers* and the underlying systems. These functions are not normally characterized by the reading or writing of data which is accomplished with the OPC UA *Attribute* service set.

Methods represent basic functions in the *Server* that can be invoked by a *Client*. *Programs*, by contrast, model more complex and stateful functionality in the system. For example, a method call may be used to perform a calculation or reset a counter. A *Program* is used to run and control a batch process, execute a machine tool part program, or manage a domain download. *Methods* and their invocation mechanism are described in IEC 62541-3 and IEC 62541-4.

This standard describes the extensions to, or specific use of, the core capabilities defined in IEC 62541-5 as required for *Programs*.

4.2 Programs

4.2.1 Overview

Programs are complex functions in a server or underlying system that can be invoked and managed by a *Client*. *Programs* can represent any level of functionality within a system or process in which client control or intervention is required and progress monitoring is desired. Figure 2 illustrates the model.

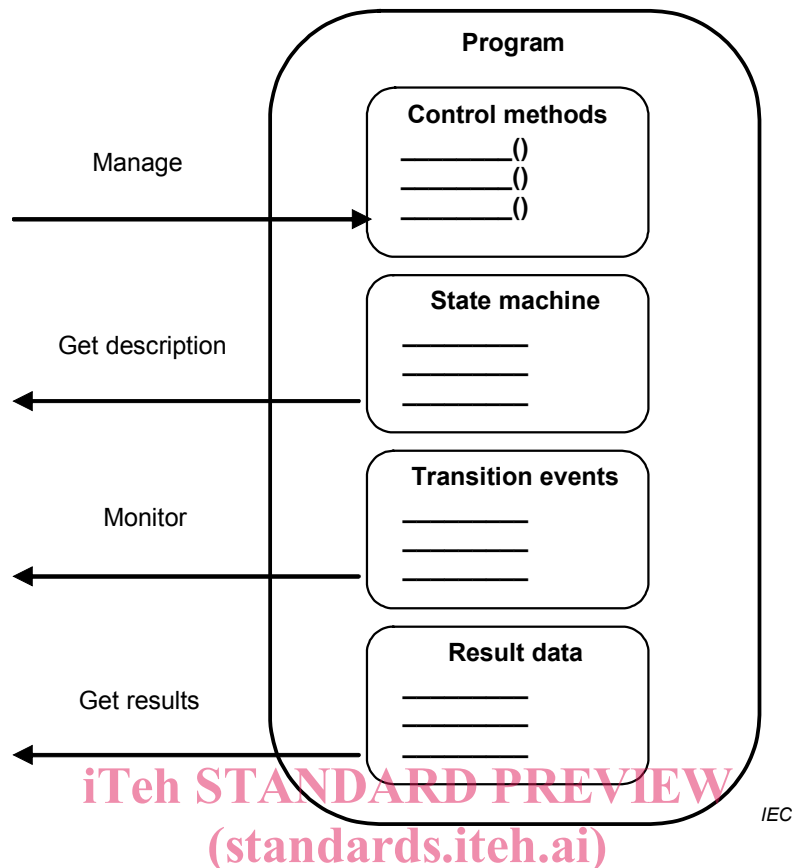


Figure 2 – Program illustration

SIST EN 62541-10:2015

Programs are state full and transition through a prescribed sequence of states as they execute. Their behaviour is defined by a *Program Finite State Machine (PFSM)*. The elements of the PFSM describe the phases of a *Program's* execution in terms of valid transitions between a set of states, the stimuli or causes of those transitions, and the resultant effects of the transitions.

4.2.2 Security considerations

Since *Programs* can be used to perform advanced control algorithms or other actions, their use should be restricted to personnel with appropriate access rights. It is recommended that *AuditUpdateMethodEvents* are generated to allow monitoring the number of running *Programs* in addition to their execution frequency.

4.2.3 Program Finite State Machine

The states, transitions, causes and effects that compose the *Program Finite State Machine* are listed in Table 1 and illustrated in Figure 3.

Table 1 – Program Finite State Machine

No.	Transition name	Cause	From state	To state	Effect
1	HaltedToReady	Reset Method	Halted	Ready	Report Transition 1 Event/Result
2	ReadyToRunning	Start Method	Ready	Running	Report Transition 2 Event/Result
3	RunningToHalted	Halt Method or Internal (Error)	Running	Halted	Report Transition 3 Event/Result
4	RunningToReady	Internal	Running	Ready	Report Transition 4 Event/Result
5	RunningToSuspended	Suspend Method	Running	Suspended	Report Transition 5