

# TM Forum Technical Report

## Intent Common Model - Vocabulary Reference

TR290V

<b>Maturity Level:</b> General availability (GA)	<b>Team Approved Date:</b> 04-Jul-2024
<b>Release Status:</b> Production	<b>Approval Status:</b> TM Forum Approved
<b>Version</b> 3.6.0	<b>IPR Mode:</b> RAND

## Notice

Copyright © TM Forum 2024. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to TM FORUM, except as needed for the purpose of developing any document or deliverable produced by a TM FORUM Collaboration Project Team (in which case the rules applicable to copyrights, as set forth in the [TM FORUM IPR Policy](#), must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by TM FORUM or its successors or assigns.

This document and the information contained herein is provided on an “AS IS” basis and TM FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

TM FORUM invites any TM FORUM Member or any other party that believes it has patent claims that would necessarily be infringed by implementations of this TM Forum Standards Final Deliverable, to notify the TM FORUM Team Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this deliverable.

The TM FORUM invites any party to contact the TM FORUM Team Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this TM FORUM Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this TM FORUM Standards Final Deliverable. TM FORUM may include such claims on its website but disclaims any obligation to do so.

TM FORUM takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this TM FORUM Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on TM FORUM's procedures with respect to rights in any document or deliverable produced by a TM FORUM Collaboration Project Team can be found on the TM FORUM website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this TM FORUM Standards Final Deliverable, can be obtained from the TM FORUM Team Administrator. TM FORUM makes no representation that any information or list

of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

Direct inquiries to the TM Forum office:

181 New Road, Suite 304  
Parsippany, NJ 07054, USA  
Tel No. +1 862 227 1648  
TM Forum Web Page: [www.tmforum.org](http://www.tmforum.org)

# Table of Contents

Notice .....	2
Table of Contents .....	4
Executive Summary.....	6
Introduction.....	7
1. Vocabulary reference .....	8
1.1. about.....	8
1.2. chooseFrom.....	8
1.3. ConditionReport.....	8
1.4. Context.....	8
1.5. context.....	8
1.6. ContextReport.....	9
1.7. contextReport .....	9
1.8. deliveryDescription.....	9
1.9. DeliveryExpectation .....	9
1.10. DeliveryExpectationReport.....	9
1.11. deliveryType .....	10
1.12. Expectation.....	10
1.13. ExpectationReport .....	10
1.14. Information.....	10
1.15. information.....	10
1.16. Intent.....	11
1.17. IntentReport.....	11
1.18. intentElements .....	11
1.19. intentHandlingState.....	11
1.20. intentUpdateState .....	11
1.21. ObservationReportingExpectation.....	11
1.22. PropertyExpectation.....	12
1.23. PropertyExpectationReport .....	12
1.24. Reason .....	12
1.25. reason.....	12
1.26. reportDestination .....	12
1.27. reportDestinations.....	12
1.28. reportGenerated .....	13
1.29. ReportingExpectation.....	13
1.30. ReportingExpectationReport .....	13

1.31.	reportNumber.....	13
1.32.	reportTriggers .....	13
1.33.	result.....	13
1.34.	resultFrom .....	14
1.35.	Target .....	14
1.36.	target .....	14
1.37.	targetCount.....	14
1.38.	TargetReport.....	14
1.39.	targetReport.....	14
1.40.	valuesOfTargetProperty.....	15
1.41.	Vocabulary.....	15
2.	Administrative Appendix.....	16
2.1.	Document History .....	16
2.1.1.	Version History.....	16
2.1.2.	Release History.....	16
2.2.	Acknowledgments.....	17

## Executive Summary

The intent common model is specified by the TR290 series of documents. This document contains the vocabulary reference. This means it contains an alphabetically sorted list of all model and intent specification language elements defined in the namespace of the intent common model. This includes the definition of the elements anchored in the underlying RDF standards and the wider TM Forum Intent Management Ontology (TIO).

## Introduction

This document serves as quick reference to all model elements of the Intent Common Model.

# 1. Vocabulary reference

This document contains a reference definition of all model vocabulary. It is sorted alphabetically:

## 1.1. about

The property `icm:about` references the intent element the intent report element is reporting about.

Instance of: `rdf:Property`

## 1.2. chooseFrom

The property `icm:chooseFrom` provides a container of resources in its range. These are the resources from which a choice can be made. The intent handler can choose a single resource or several depending on the use case specific needs. These resources would then be considered additions to the target assigned to the delivery expectation.

Instance of: `rdf:PropertyDomain: icm:DeliveryExpectation`

Range: `rdfs:Container`

## 1.3. ConditionReport

The report for a condition object is provided by an instance of class `icm:ConditionReport`.

Instance of: `rdfs:Class`

## 1.4. Context

The class `icm:Context` expresses that its instances are providing contextual information to other intent elements. Context objects provide additional information which impacts the interpretation of requirements. The intent common model does not define specific types of context, but intent extension models can define subclasses of `icm:Context` to do so.

Instance of: `rdfs:Class`

Subclass of: `icm:IntentElement`

## 1.5. context

The property `icm:context` allows assigning context objects to resources and in particular to elements of an intent. The intent common model does not specific types of context, but intent extension models can define sub properties of `icm:context` specific to subclasses of context.

Instance Of: `rdf:Property`  
Range: `icm:Context`

## 1.6. ContextReport

A report about a context is provided by an instance of class `icm:ContextReport`.

Instance of: `rdfs:Class`

## 1.7. contextReport

The property `icm:contextReport` can be used to assign context reports to elements of the intent report that reports about the elements of an intent that had the respective context.

Instance of: `rdf:Property` Range: `icm:ContextReport`

## 1.8. deliveryDescription

The property `icm:deliveryDescription` is a parameter used in delivery expectations. It expresses where a detailed description of the expected delivery can be found and specifies the resource that contains it.

Instance of: `rdf:Property`  
Domain: `icm:DeliveryExpectation`  
Range: `rdfs:Resource`

## 1.9. DeliveryExpectation

The class `icm:DeliveryExpectation` expresses that its instance is a special type of expectation for specifying the requirement that something needs to be delivered. The details are expressed by its properties.

Instance of: `rdfs:Class`  
Subclass of: `icm:Expectation`

## 1.10. DeliveryExpectationReport

The report for a delivery expectation is provided by an instance of class `icm:DeliveryExpectationReport` within an Intent Report.

Instant of: `rdfs:Class`  
Subclass of: `icm:ExpectationReport`

## 1.11. **deliveryType**

The property `icm:deliveryType` is a parameter used in delivery expectations. It expresses the requirement that the target resource shall be an instance of the specified class.

Instance of: `rdf:Property`  
Domain: `icm:DeliveryExpectation`  
Range: `rdfs:Class`

## 1.12. **Expectation**

The class `icm:Expectation` expresses that its intent is an expectation object. It defines a set of requirements within an Intent.

Instance of: `rdfs:Class` Subclass of: `icm:IntentElement`

## 1.13. **ExpectationReport**

The class `icm:ExpectationReport` with its subclasses represent the reports for expectations defined in an intent.

Instance of: `rdfs:Class`

## 1.14. **Information**

The instances of class `icm:Information` provide additional information about resources and in particular intent or intent report elements. Information objects provide additional insights and context but without impacting the interpretation of requirements. The intent common model does not define specific types of information, but intent extension models can add subclasses of `icm:Information` to do so.

Instance of: `rdfs:Class`  
Subclass of: `icm:IntentElement`

## 1.15. **information**

The property `icm:information` allows assigning information objects to resources and in particular to elements of an intent or intent report. The intent common model does not define specific types of information, but intent extension models can add subclasses of `icm:Information` and also define respective sub properties of `icm:information` with a range specific to subclasses of information.

Instance of: `rdf:Property`  
Range: `icm:Information`

## 1.16. Intent

The class `icm:Intent` expresses that its instance is an intent object. As such it defines requirements.

Instance of: `rdfs:Class`  
 Subclass of: `icm:IntentElement`

## 1.17. IntentReport

Intent reports are instances of the class `icm:IntentReport`.

Instance of: `rdfs:Class`

## 1.18. intentElements

The function `icm:intentElements` represents a container of all individual intent element resources used to express requirements for the intent elements given as arguments. This includes the given intent element object itself and its parents. It represents all intent elements in the subtree that contains the function argument.

Instance of: `fun:Function`  
 Result type: `rdfs:Container`  
 Arity: Any number of arguments  
 Argument Types: All arguments are of type `icm:IntentElement`

## 1.19. intentHandlingState

The intent handler maintains an intent handling state machine for every intent. An intent report states the intent handling state at the moment of intent report generation using of the property `icm:intentHandlingState`.

Instance of: `rdf:PropertyDomain: icm:IntentReportRange:`  
`imo:IntentHandlingState`

## 1.20. intentUpdateState

The intent handler maintains an intent update state machine for every intent. An intent report states the intent update state at the moment of intent report generation using of the property `icm:intentUpdateState`.

Instance of: `rdf:PropertyDomain: icm:IntentReportRange:`  
`imo:IntentHandlingState`

## 1.21. ObservationReportingExpectation

Requesting intent reports with reporting expectations of class `icm:ObservationReportingExpectation` leads to intent reports containing observations for metrics used in the intent.

Instance of: rdfs:Class Subclass of: icm:ReportingExpectation

## 1.22. PropertyExpectation

The class icm:PropertyExpectation expresses that its instance is a special type of expectation for specifying requirements based on properties such as, for example, metrics, KPI, states and other observations.

Instance of: rdfs:Class  
Subclass of: icm:Expectation

## 1.23. PropertyExpectationReport

The report for a property expectation is provided by an instance of class icm:PropertyExpectationReport within an intent report.

Instance of: rdfs:Class  
Subclass of: icm:ExpectationReport

## 1.24. Reason

Instances of class icm:Reason specify explanations for intent handling results.

Instance of: rdfs:Class  
Subclass of: icm:ExpectationReport

## 1.25. reason

The property icm:reason would assign a reason to an intent report element.

Instance of: rdf:PropertyRange: icm:Reason

## 1.26. reportDestination

The property icm:reportDestination is used with a reporting expectation, and it specifies where to send the required report to. The object of the property is a resource that represents a wanted receiver of the intent report. When a report is generated, it would be sent to this receiver.

Instance of: rdf:PropertyDomain: icm:ReportingExpectation

## 1.27. reportDestinations

The property icm:reportDestinations is used with a reporting expectation, and it specifies where to send the required report to. The object of the property is a container with member resources that represent the wanted receivers of the intent report. When a report is generated, it would be sent to all receivers specified in the container.

Instance of: rdf:PropertyDomain: icm:ReportingExpectationRange:  
rdfs:Container

## 1.28. reportGenerated

The property icm:reportGenerated assigns the timestamp of report generation to an intent report instance.

Instance of: rdf:PropertyDomain: icm:IntentReportRange: t:Instant

## 1.29. ReportingExpectation

The class icm:ReportingExpectation expresses that its instance is a special type of expectation for specifying the conditions and scope of intent reports. It allows subscribing to intent reports and determine its content and condition of generation.

Instance of: rdfs:ClassSubclass of: icm:Expectation

## 1.30. ReportingExpectationReport

The report for a reporting expectation is provided by an instance of class icm:ReportingExpectationReport.

Instance of: rdfs:ClassSubclass of: icm:ExpectationReport

## 1.31. reportNumber

The property icm:reportNumber assigns a sequence number to the intent report instance.

Instance of: rdf:PropertyDomain: icm:IntentReportRange: xsd:positiveInteger

## 1.32. reportTriggers

The property icm:reportTriggers specifies the events that shall initiate the generation and distribution of an intent report. It is a property used in reporting expectations.

Instance of: rdf:Property  
Domain: icm:ReportingExpectation  
Range: rdfs:Container with members of type imo:Event

## 1.33. result

The property icm:result states the evaluation result of the intent element at the time of report generation. It is a boolean truth value.

Instance of: rdf:PropertyRange: xsd:boolean

## 1.34. resultFrom

The property `icm:resultFrom` associates an element of the intent report with another element of an intent report. Its subject is the report of an intent element that has a boolean truth value resulting from a logical evaluation. The object of `icm:resultFrom` is the report of an element in the intent that directly contributed to that evaluation.

Instance of: `rdf:Property`

## 1.35. Target

The class `icm:Target` expresses that its instance is a target specification. Targets define a set of resources that are expected to fulfill the requirements they are associated with through an expectation.

Instance of: `rdfs:Class`

Subclass of: `rdfs:Container` Subclass of: `icm:IntentElement`

## 1.36. target

The property `icm:target` assigns an object of class `icm:Target` to an expectation. It therefore specifies the targets for all requirements within this expectation.

Instance of: `rdf:Property`

Domain: `icm:Expectation`

Range: `icm:Target`

## 1.37. targetCount

The property `icm:targetCount` specifies the number of member resources of the target report object.

Instance of: `rdf:Property`

Domain: `icm:TargetReport`

Range: `xsd:nonNegativeInteger`

## 1.38. TargetReport

The report for a target is provided by an instance of class `icm:TargetReport`.

Instance of: `rdfs:Class`

## 1.39. targetReport

The report for a target is assigned to an expectation report with the property `icm:targetReport`.

Instance of: `rdf:Property` Domain: `icm:ExpectationReport` Range: `icm:TargetReport`

## 1.40. valuesOfTargetProperty

The function `set:valuesOfTargetProperty` represents a container of all values that are assigned to members of the target container with the properties specified as function argument. If multiple properties are specified as arguments the result is a superset of all values from all specified properties.

Instance of: `fun:FunctionResult` type: `xsd:booleanArity`: At least one  
argumentArgument types: All arguments are instances of `rdf:Property`

## 1.41. Vocabulary

The object `icm:Vocabulary` is a container of all model elements.

Instance of: `rdfs:Container`

## 2. Administrative Appendix

### 2.1. Document History

#### 2.1.1. Version History

Version Number	Date Modified	Modified by:	Description of changes
3.0.0	07-Feb-2023	Alan Pope	Final edits prior to publication
3.1.0	11-Apr-2023	Alan Pope	Final edits prior to publication
3.2.0	15-Aug-2023	Alan Pope	Final edits prior to publication
3.4.0	29-Feb-2024	Alan Pope	Final edits prior to publication
3.5.0	03-May-2024	Alan Pope	Final edits prior to publication
3.6.0	04-Jul-2024	Alan Pope	Final edits prior to publication

#### 2.1.2. Release History

Release Status	Date Modified	Modified by:	Description of changes
Pre-production	07-Feb-2023	Alan Pope	Updated to version 3.0.0
Pre-production	17-Mar-2023	Adrienne Walcott	Updated to Member Evaluated status
Pre-production	11-Apr-2023	Alan Pope	Updated to version 3.1.0
Pre-production	15-May-2023	Adrienne Walcott	Updated to Member Evaluated status
Pre-production	15-Aug-2023	Alan Pope	Updated to version 3.2.0
Pre-production	18-Sep-2023	Adrienne Walcott	Updated to Member Evaluated status
Pre-production	29-Feb-2024	Alan Pope	Updated to version 3.4.0
Production	26-Apr-2024	Adrienne Walcott	Updated to reflect TM Forum Approved status
Pre-production	03-May-2024	Alan Pope	Updated to version 3.5.0
Production	28-Jun-2024	Adrienne Walcott	Updated to reflect TM Forum Approved status
Pre-production	04-Jul-2024	Alan Pope	Updated to version 3.6.0
Production	30-Aug-2024	Adrienne Walcott	Updated to reflect TM Forum Approved status

## 2.2. Acknowledgments

Team Member (@mention)	Company	Role*
<a href="#">Jörg Niemöller</a>	Ericsson	Author, Project Co-Chair
<a href="#">Kevin McDonnell</a>	Huawei	Project Co-Chair
<a href="#">Yuval Stein</a>	Amdocs	Project Co-Chair
<a href="#">Kamal Maghsoudlou</a>	Ericsson	Key Contributor
<a href="#">Leonid Mokrushin</a>	Ericsson	Key Contributor
<a href="#">Marin Orlić</a>	Ercisson	Key Contributor
<a href="#">Aaron Boasman-Patel</a>	TM Forum	Additional Input
<a href="#">Alan Pope</a>	TM Forum	Additional Input
<a href="#">Dave Milham</a>	TM Forum	Additional Input
<a href="#">Xiao Hongmei</a>	Inspur	Reviewer

\*Select from: Project Chair, Project Co-Chair, Author, Editor, Key Contributor, Additional Input, Reviewer