|  |  |
| --- | --- |
|  | Special Publication 500-xxx |
| **Cloud Computing Service Metrics Description** | |
| *NIST Cloud Computing Reference Architecture*  *and Taxonomy Working Group*  *NIST Cloud Computing Program*  *Information Technology Laboratory* | |
|  | |



This page is left intentionally blank

**NIST Special Publication 500-XXX**

**Cloud Computing Service**

**Metrics Description**

NIST Cloud Computing Reference Architecture

and Taxonomy Working Group

http://dx.doi.org/10.6028/NIST.SP.XXX

August 2014



U.S. Department of Commerce

*Penny Pritzker, Secretary*

National Institute of Standards and Technology

*Willie May, Acting Under Secretary of Commerce for Standards and Technology and Acting Director*

This page is left intentionally blank

**Reports on Computer Systems Technology**

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation’s measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL’s responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of other than national security-related information in Federal information systems. This Special Publication 500-series reports on ITL’s research, guidance, and outreach efforts in Information Technology and its collaborative activities with industry, government, and academic organizations.

**National Institute of Standards and Technology Special Publication 500-xxx**

Natl. Inst. Stand. Technol. Spec. Publ. 500-xxx, 24 pages (August 2014)

|  |
| --- |
| DISCLAIMER |
| This document has been prepared by the National Institute of Standards and Technology (NIST) and describes technical research in support of the NIST Cloud Computing Program.  Certain commercial entities, equipment, or material may be identified in this document in order to describe a concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that these entities, materials, or equipment are necessarily the best available for the purpose. |

**Acknowledgements**

This document reflects the contributions and discussions by the members of the NIST Cloud Computing Reference Architecture and Taxonomy Working Group – Cloud Service Metrics Sub Group, lead by Frederic de Vaulx of Prometheus Computing, LLC under contract with the U.S. Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory.

NIST SP 500-xxx has been collaboratively authored by the NIST Cloud Service Metrics Sub Group.

NIST would like to acknowledge the specific contributions from the following contributors:

|  |
| --- |
| Jane Siegel (CMU) |
| Jacques Durand (Fujitsu) |
| John Calhoon (Microsoft) |
| Jeff Perdue (CMU) |
| Steve Woodward (Cloud Perspectives) |
| Alan Sill (TTU) |
| Ken E. Stavinoha (Solutions Architect.  Cisco Systems) |
| Tom Rutt (Fujitsu) |
| Jenny Huang (AT&T) |
| Eric Roggenstroh (eGT for GSA) |
| Omar Fink (SAIC) |
| Steven J. McGee (SAW Concepts LLC) |
| Eric Simmon (NIST) |
| Scott Feuless (ISG/CSMIC) |
| Jim Watts (TSA) |
| Keyun Ruan (Chief Scientist, Espion Group, Ireland) |
| Jesus Luna (CSA) |
| William M. Fitzgerald (EMC Information Systems International, Ireland) |
| Kimberley Laris (Positive Assurance) |
| Massimiliano Rak (CerICT, Italy) |
| Neeraj Suri (Technical University of Darmstadt, Germany) |
| David Núñez (Universidad de Málaga, Spain) |
| Carmen Fernández-Gago (Universidad de Málaga, Spain) |
| Isaac Agudo (Universidad de Málaga, Spain)  Annie Sokol (NIST)  Robert Bohn (NIST) |

The NIST editors for this document were: Frederic de Vaulx and Eric Simmon.

Table of Contents

1 Executive Summary 1

2 Introduction 3

2.1 Audience 3

2.2 Background 3

3 Definitions 4

3.1 Abstract Metric 4

3.2 Cloud Service Property 5

3.3 Context 5

3.4 Measurement 5

3.5 Measurement Result 5

3.6 Metric 5

3.7 Observation 5

3.8 Unit of Measurement 5

4 The Role of Metrology in Cloud Services 6

4.1 The Cloud Service Trifecta 7

4.1.1 Metrics for Selecting Cloud Services 7

4.1.2 Metrics for Service Agreements (SAs) 7

4.1.3 Metrics for Service Measurement 8

4.2 Other Metrics 8

4.3 Scenario 9

5 Cloud Service Metric Model 10

5.1 Cloud Service Metric Ecosystem 11

5.2 Model Characteristics 11

5.2.1 Consistent Representation of Information 12

5.2.2 Explicit Relationships 12

5.2.3 Repository of Definitions 12

5.2.4 Comparablity 12

5.2.5 Flexiblity and Adaptablity 12

5.2.6 Composability 12

5.3 Cloud Service Metric Diagram 12

5.4 Cloud Service Metric Element Definitions 13

5.4.1 AbstractMetric Class 13

5.4.2 Metric Class 15

5.4.3 MetricParameter Class 16

5.4.4 MetricRule Class 16

5.4.5 ParameterDefinition Class 16

5.4.6 RuleDefinition Class 17

6 How to Use the CSM Model 18

7 Measurement Uncertainty 19

8 Other Considerations 20

8.1 Metrics used for Property Composition 20

8.2 Calibration & Measurement Standard (Etalon) 20

9 Conclusion 21

Annex A - Definitions Survey 22

Annex B - References 24

List of Figures

Figure 1 Metric and Property 9

Figure 2 Cloud Service Selection 10

Figure 3 Cloud Service Negotiation & Agreement 11

Figure 4 Cloud Service Objectives Monitoring 11

Figure 5 Scenario and Metric 13

Figure 6 Cloud Service Metric Ecosystem Model 14

Figure 7 Cloud Service Metric (CSM) 16

Figure 8 CSM Core Parts 21

Figure 9 Metric Definition Process 21

# ****Executive Summary****

As cloud computing becomes more and more mainstream, the quantity of computing resources being offered as services increases dramatically. More and more consumers of digital services are relying directly or indirectly upon cloud services. Complex systems are the result of interconnected or federated cloud services. For these new business arrangements to work both customers and providers of cloud services must be confident in selecting the services (understanding how service capabilities match to business requirements, and comparing services), in agreeing on purchased services (Service Level Agreements) and in the delivery of these purchased services (services performance and remedies).

NIST in its definition of cloud computing describes a “Measured Service” as being one of the five essential characteristics of the cloud computing model. This characteristic is addressed in part by the identification of what properties of a cloud service should be measured and what standards of measurement or metrics should be used. Metrology – the science of measurement – is thus very important for cloud computing not just for the measurement of properties of cloud services, but also to gain a common understanding of the properties themselves.

In this context, the role of metrics is very important to support decision-making including:

* Selecting cloud services
* Defining and enforcing service agreements
* Monitoring cloud services
* Accounting
* Auditing

A metric provides knowledge about aspects of a cloud property through both its definition (e.g. expression, unit, rules) and the values resulting from the observation of the property based on the metric. The use of a metric through an observation results in measurement results to estimate the property of an element. For instance a customer response time metric can be used to estimate a specific response time property (i.e. response time from customer to customer) of a cloud email service search feature. It also provides necessary information for repeatability and reproducibility of observations and measurement results.

A metric can be described using the model proposed in this document. It represents the necessary information to understand what cloud property is targeted and what constraints should be applied during observation. This model called the Cloud Service Metric model (CSM) contains concepts to describe abstract metric definitions for a specific cloud service property like service uptime. Abstract metric definitions show how abstract metrics are composed together along with parameters and rules to express a formal understanding the property of interest. The CSM model also contains concrete metric definitions that are based on abstract metric definitions. Concrete metric definitions add specific values to rules and parameters that make the metric usable for a given scenario.

A scenario represents a particular use case in which metrics will play a role. Stakeholders need to have a way to understand, assess, compare, combine and make decisions about cloud service properties. This means that for a given scenario (e.g. choosing a cloud service or setup a service agreement), a stakeholder needs to be able to get information on cloud service properties, which when measured (observed) will help the stakeholder choose the proper course of action. The scenario and cloud service property will determine the metric (standard of measurement) to be used.

The document describes a possible way to use the CSM model in order to create abstract metric definitions and concrete metric definitions.

The understanding of the relationships between different data elements of cloud service metrology is very important in order to create meaningful and traceable metrics

# Introduction

## Audience

This document proposes a framework that identifies and characterizes the information and relationships needed to describe and observe properties of cloud services that are representative, accurate and reproducible. These metrics can connect information intended for decision-making, for agreements between provider and customer, for runtime performance measurement and the underlying properties within the providers system.

This information can be used in a variety of ways including but not limited to collecting, comparing, analyzing gaps, assessing or describing metrics at the technical or business levels.

The measurement process and methodology necessary for performing the measurement of a given property is not the focus of this document.

This document may be used as a source of information to better understand metrology within the context of cloud services, and as a framework to describe, collect and access information related to metrics.

The targeted audience of this document includes but is not limited to:

* U.S. Government agencies
* Cloud service customers
* Cloud service auditors
* Cloud service providers

## Background

Cloud computing is shifting the use of compute resources from asset-based physical resources to service-based virtual resources. NIST in its definition of cloud computing [2] describes a “Measured Service” as being one of the five essential characteristics of the cloud computing model. Providing data on measurable capabilities (such as; quality of service, security features, availability and reliability) can give the cloud service customer the opportunity to make informed choices and to gain understanding of the state of the service being delivered. It can also give the cloud service provider the opportunity to present the properties of their cloud services to the cloud service customer.

However, as of today, cloud metrology is not necessarily well understood. Common terminologies (i.e. the definition of measurement, metric, and related concepts) or sets of measurement artifacts (i.e. unit of measurement, metric) often have several definitions, which makes it very difficult for the cloud service customer to compare services or rely on third party tools to monitor the health of the service. It can also make it difficult for the provider to show that the service is performing correctly or to allow its service to enter into complex cloud service chain or federation.

Furthermore organizations like U.S. agencies need a way to consistently define sets of metrics that they can rely on, trust and share. This can have the effect of increasing the confidence in the results of measurements of selected cloud service properties and thus increase the support of the decision-making process during the different stages of the cloud service lifecycle.

It is critical to have the capacity to represent what needs to be measured, what uses the result of the measurement provide, and how they would affect decisions (e.g. business or technical).

Cloud metrology is vast and takes into account many different components including:

* The definition of metrics and their use.
* The definition of measurement processes and methods.
* The calibration of measurement tools.
* The measurement operations.
* The processing of measurement results and associated consequences.

This document’s primary focus is on the first item and introduces one approach to define and represent the concepts and uses of measurement within the context of cloud services and their underlying components.

# Definitions

Currently, terminology related to cloud service measurements are not well defined. Different stakeholders in the Information Communication Technology (ICT) community use the same terms with slightly different (or sometimes greatly different) meanings. This may be due to wide variety of ICT’s technology domains (i.e. Software, Telecommunication, Manufacturing), each using it’s own language. It could also come from the lack of a common process to define new terminology. This leads to great confusion among cloud service providers, customers, and other cloud stakeholders.

The use of well-defined and understood terms within a given domain will enable the stakeholders to communicate more efficiently. It helps reduce the risk of the misinterpretation of information and facilitates the combination and comparison of information.

To improve this situation, some of the core terms applicable to cloud metrology and used throughout the document are defined below.

## Abstract Metric

An abstract standard of measurement used to assess a property. The standard of measurement describes what the result of the measurement means, but not how the measurement was performed. The Abstract Metric is not used by itself, but is instantiated using a Metric.

## Cloud Service Property

The property of a cloud service entity to be observed. A property may be qualitative or quantitative.

## Context

The circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood and assessed.

## Measurement

Set of operations having the object of determining a Measurement Result.

*Note:* Based on the definition of Measurement in ISO/IEC 15939:2007 [6]. Also used here to describe an actual instance of execution of these operations leading to the production of a Measurement Result instance.

## Measurement Result

Value that expresses a qualitative or quantitative assessment of a property of an entity.

*Note:* Based on the definition of Measurement Result in ISO/IEC 15939:2007 [6]

*Note:* The term measure is **not** used in this document. Measure is defined with so many divergent definitions it is difficult to use. Section 9 “Definitions Survey” shows a sample of the definitions related to “measure”.

## Metric

A standard of measurement that defines the conditions and the rules for performing the measurement and for understanding the results of a measurement.

*Note:* A metric implements a particular abstract metric concept.

*Note:* A metric is to be applied in practice within a given context that requires specific properties to be measured, at a given time(s) for a specific objective.

## Observation

Measurement based on a metric, at a point in time, on a measurement target.

## Unit of Measurement

Real scalar quantity, defined and adopted by convention, with which any other quantity of the same kind can be compared to express the ratio of the two quantities as a number [7].

*Note:* part of an Abstract Metric

# The Role of Metrology in Cloud Services

Metrology – the science of measurement – is important for cloud computing not just for the measurement of properties of cloud services, but also to gain a common understanding of the properties themselves.

Physical properties on one hand can be measured using a standardized metrology process. Software properties measurement on the other hand has some associated standards like functional size measurement methods [3][4][5] that are not exactly at the level of physical metrology.

Metrics are used to understand a particular measurement (or type of measurement) of a cloud service property and to understand the property itself by providing a standard for describing a measurement and measurement result.

Figure 1 shows the relationship between a metric and a property. The use of a metric through an observation results in measurement results to estimate the property of an element. For instance a customer response time metric can be used to estimate a specific response time property (i.e. response time from customer to customer) of a cloud email service search feature.

A metric provides knowledge about aspects of the property through its definition (e.g. expression, unit, rules). It also provides the necessary information for repeatability and reproducibility of observations and measurement results.

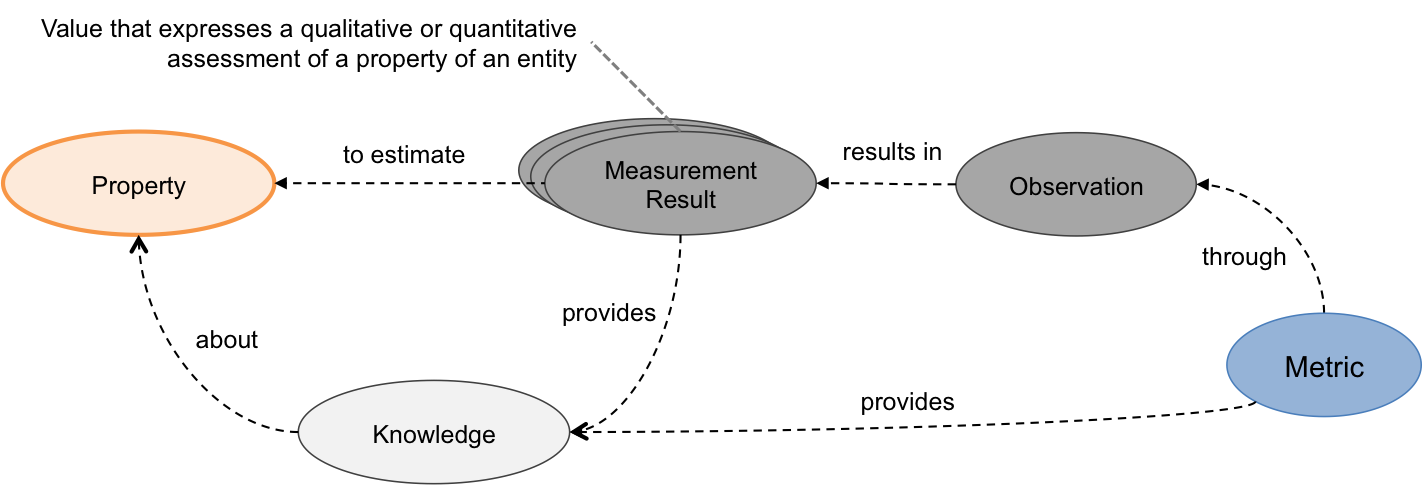


Figure Metric and Property

In this manner cloud metrics help providers communicate the properties of their cloud services that are measurable, help customers and providers agree on what will be provided, and allow cloud service features to be measured to ensure the agreement is met (and therefore the customers requirements are met).

Cloud system can leverage metrics – standards for measurements – for many different purposes. For instance metrics can be used at different layers of a cloud computing system (e.g. hardware layers, logic layers, governance layers or service layers). They can also be used at different stages of the cloud computing services life cycle (e.g. procurement, operation, audit and retirement).

## The Cloud Service Trifecta

The use of metrics for cloud computing systems at the service interface can be broken down into three general areas, service selection, service agreement, and service verification. Metrics are essential, not just to understand each of these areas, but to connect these three distinct parts of the cloud procurement process. The three aspects of the trifecta are described below.

### Metrics for Selecting Cloud Services

Metrics are essential at the stage of deciding what cloud offering should be best suited to meet the business and technical requirements. The customer of cloud services should be able to select and use metrics and their underlying measures to assess and decide which offering would be best. Solutions like the Cloud Services Measurement Initiative Consortium (CSMIC) Service Measurement Index (SMI) [8] could be used to determine what metrics could be relevant to the selection of a particular cloud offering.

Figure 2 shows how metrics are used to understand the factors necessary to distinguish and decide between two different cloud offerings. Such metrics may used to provide data on actual cloud operations (performance, responsiveness, scalability, availability…) e.g. as produced by some independent auditing or monitoring of the provider when servicing its current customers. The use of these metrics can also result in an assessment on the readiness and ability of a cloud service provider to ensure some level of service quality prior to and independently from actual operations (e.g. various aspects of security, accessibility, customer support, financial flexibility).

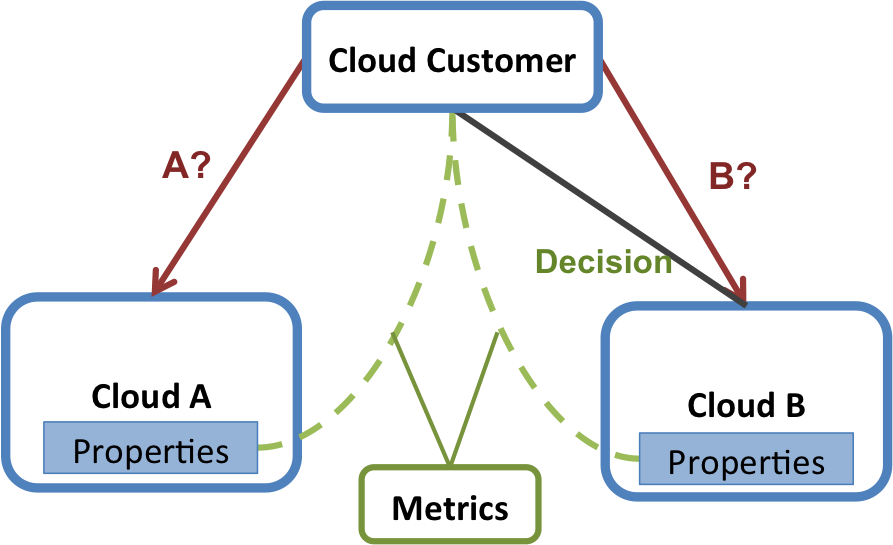


Figure Cloud Service Selection

### Metrics for Service Agreements (SAs)

A Service Agreement (SA) represents a binding agreement between the provider and customer of a cloud service. Among the things that it contains are the description of the service, the rights and responsibilities of both the provider and the customer and terms definition. It also contains essential information related to the measurement of different aspects of the cloud service (e.g., its business level objectives or its performance level). The definition and usage of appropriate metrics with their underlying measures are essential components of the Service Level Agreement (SLA) and Service Level Objectives (SLO), which are constituents of the SA. The references [9] and [10] describe, in additional details, the importance of and need for metrics in SLAs. At this point, the metrics are used to set the boundaries and margins of errors the provider of the service abides by and their limitations. For instance these metrics could be used at runtime for service monitoring and balancing, or remediation (e.g., financial). Using a standardized set of metrics or metric templates in SAs makes it easier and quicker to define SLAs and SLOs, and to compare them with others.

Figure 3 illustrates the use of metrics to support an SLA document that defines the expectations of the two parties, allowing the two parties to understand the characteristics of the specific service being provided.

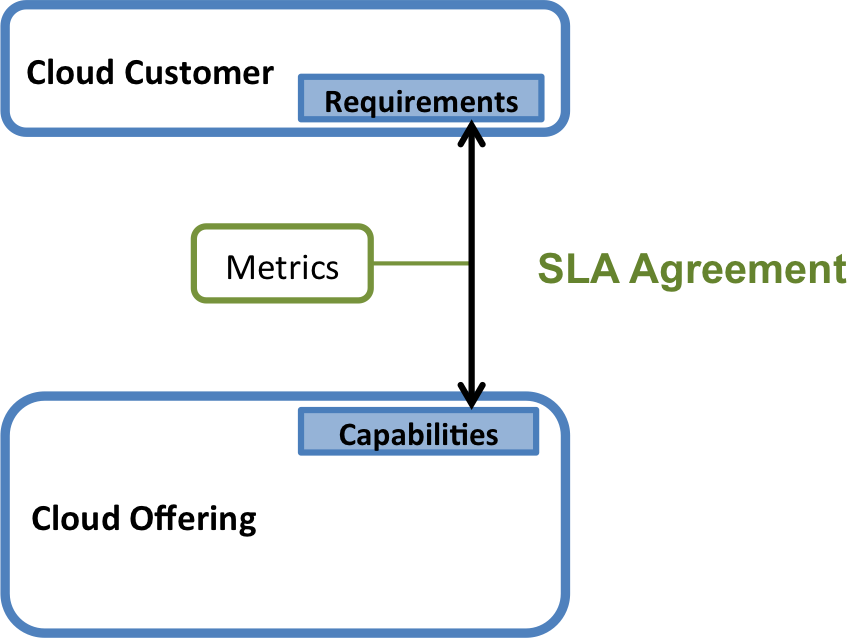


Figure Cloud Service Negotiation & Agreement

### Metrics for Service Measurement

Once the customer purchases a cloud service, the customer needs to verify he is getting what he agreed to and paid for. This is necessary to ensure the service level objectives are being met, or to initiate a remedy if the service level objectives are not being met.

Figure 4 illustrates the service being delivered to the cloud customer from the cloud provider. In this case, metrics are used when monitoring the service level objectives defined in the service agreements.

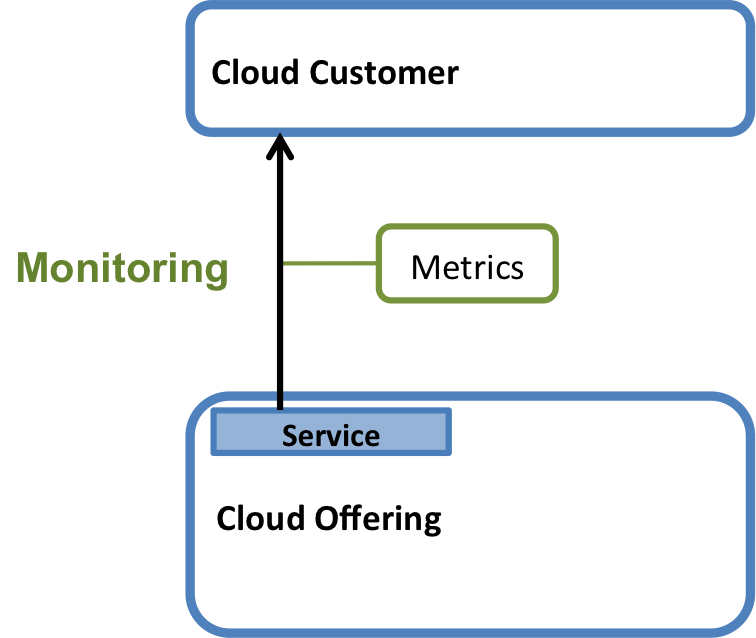


Figure Cloud Service Objectives Monitoring

## Other Metrics

As stated earlier in the document, metrics are used to help inform stakeholders on the performance and quality of a cloud service offering and to verify the service performance. Metrics can also be used internally to the cloud service itself. Here the metrics would be more technical and used only by the cloud service provider to monitor and understand the internal performance of their cloud system. References [11] or [12] pertain to specifications that show possible representation and usages of measurement concepts that can be used in cloud computing systems. Measurement results based on metrics for internal use may not be available to the cloud customer.

In addition, other parts of the cloud ecosystem can be influenced by the usage of metrics like accounting, auditing. In the case of accounting, metrics are for instance linked to the amount of usage of a particular service. In the case of auditing, metrics are for instance linked to the certification assessment of selected cloud service properties.

## Scenario

Stakeholders need to have a way to understand, assess, compare, combine and make decisions about cloud service properties. This means that for a given scenario (e.g. choosing a cloud service or setup a service agreement), a stakeholder needs to be able to get information on cloud service properties, which when measured (observed) will help the stakeholder choose the proper course of action. The scenario and cloud service property will determine the metric (standard of measurement) to be used. The metric relies on the abstract metric definitions that are related to the selected cloud service property. The measurement (observation) of the cloud service property through the metric will result in measurement results.

Figure 5 shows the scenario concept:

* The **Scenario** represents a particular use case (business process decision making, application monitoring, Service Level Agreements, etc.)
* The **Abstract Metric** describes the base concept that the Metric is based on.
* The **Metric** adds the data necessary to use the abstract metric in practice.
* The **Measurement** **Result** is data that results from making a measurement that follows a given metric.

More specifically, stakeholders (e.g. cloud customer or cloud provider) define the scenariofor which the metric will be needed.Thescenariorepresents:

* The expectations of an underlying business or operational process (e.g. SLA or Operation)
* How the metrics are used to assist in such a process
* What acceptable levels of the measured properties are

The scenario also includes the way the selected metrics are applied – what resource or service they support, under which conditions are their evaluation triggered and the frequency of the evaluation.

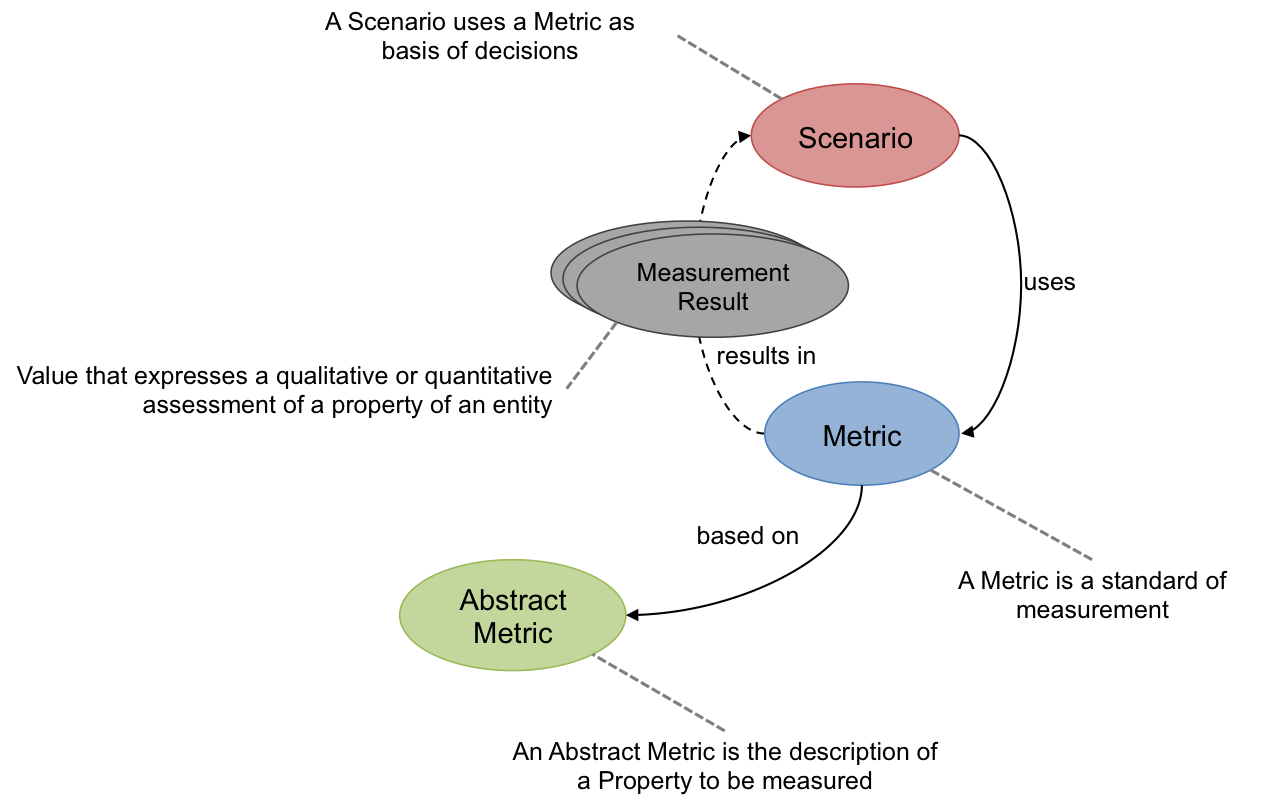


Figure Scenario and Metric

In other words, a metric is a standard set of procedures and rules that generates values for its associated abstractmetric. In practice, the metric is applied within a given scenario that determines specific conditions, such as a specific resource(s) being measured, at a given time(s) for a specific objective.

Possible scenarios could be the application of an availability metric for a performance objective of 99% in an SLA scenario or the application of an accessibility metric for a usability objective of value “high” in a decision process scenario.

# Cloud Service Metric Model

The understanding of the relationships between different data elements of cloud service metrology is very important in order to create meaningful and traceable metrics. This section introduces the Cloud Service Metric model (CSM), its general concept and a full element description of the foundation diagram that describes the Metric definition.

## Cloud Service Metric Ecosystem

As explained in the earlier *Section 3 “The Role of Metrology in Cloud Services”,* a metric is a fundamental concept that should provide information on how to understand a property being observed and how to estimate its value through observations.

The information making up the metric ecosystem can be broken down into different parts that focus on specific aspects:

* The description and definition the standard of measurement *(e.g. metric for customer response time) – CSM*
* The addition of the context of the standard of measurement *(e.g. objectives and applicability conditions of the customer response time metric) – CSM Context*
* The use of the standard of measurement to define observations *(e.g. the observation of response time property based on the customer response time metric) – CSM Observation*
* The use of the standard of measurement in a scenario *(e.g. the selection and use of the customer response time metric in an SLA scenario) – CSM Scenario*

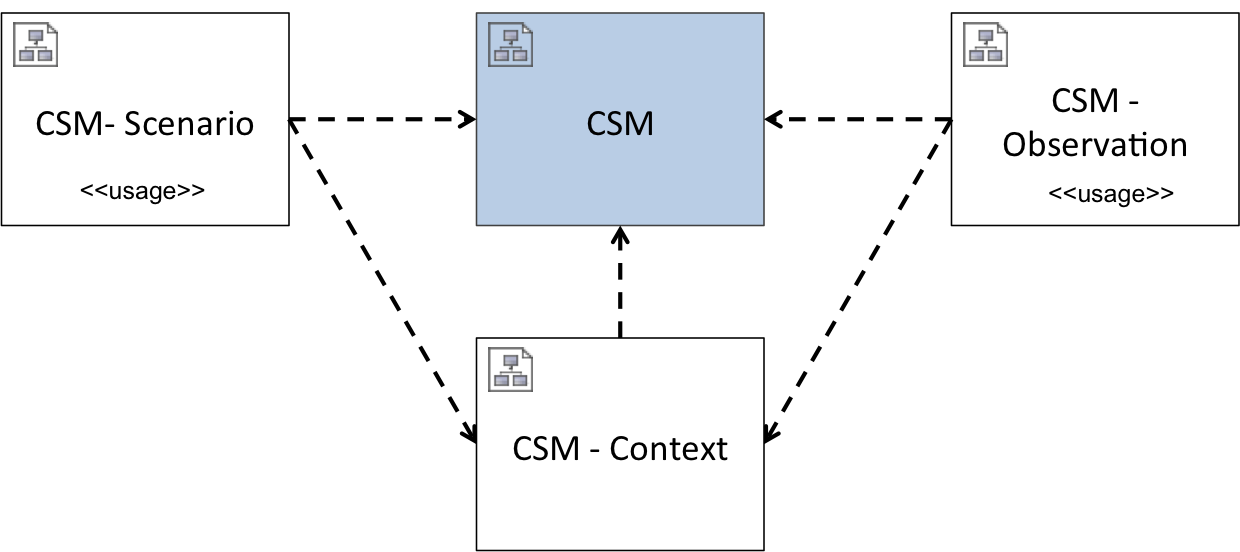


Figure Cloud Service Metric Ecosystem Model

Figure 6 describes the breakdown of these different aspects and how they relate to each other. The primary piece is the CSM diagram that contains the elements to define the standard of measurement. The CSM diagram can be enhanced with information from the CSM Context diagram that contains elements that describe the environment of a particular standard of measurement. The CSM Scenario diagram relies on the CSM diagram and the CSM Context diagram and contains elements that describe use cases that rely on standards of measurement. The CSM Observation diagram relies on the CSM Core diagram and the CSM Context diagram and contains elements that are used during measurement operations.

In this document, only the CSM diagram is explained. The remaining diagrams that compose the CSM Ecosystem model are part of another document complementary to this one

## Model Characteristics

The following are characteristics that were considered important when creating the CSM model.

### Consistent Representation of Information

Information related to metrics should be represented in a consistent, repeatable way in order to efficiently organize it, share it and use it.

### Explicit Relationships

Concepts like metrics should be represented in such a way that the relationships among them, if any, are explicit. This clarifies the effects these concepts have on one another and their importance.

### Repository of Definitions

There should be a way to organize metrics so they are reusable, searchable and derivable.

### Comparablity

The properties of the different concepts should allow its user to have enough information to efficiently compare them to find and understands either similarities or differences.

### Flexiblity and Adaptablity

The concept model should be sufficiently flexible and adaptable to allow for easy integration with other metric models. These models could be complementary to the concept model (e.g. represent measurement methods and process).

### Composability

The metrics should allow metrics definitions and instances to be reusable. Thus one should be able to use one or more metrics to build a composite metric. This metric that is composed of underlying metrics builds upon the information they contain. This results in metrics that could possibly be composed of underlying metrics of different kind (e.g. qualitative and quantitative). This consideration will be discussed in *Section 7 “Other Considerations”*.

## Cloud Service Metric Diagram

This subsection formally describes the CSM diagram, its elements, what these elements are composed of and the way they are connected to each other.

Figure 7 introduces the CSM concept as a UML class diagram [13]. The purpose of the CSM model is to capture the information needed to describe and understand a metric that is used for gaining knowledge about, and measuring cloud service properties.



Figure 7 Cloud Service Metric (CSM)

## Cloud Service Metric Element Definitions

The different elements of the CSM model are described below. In this section the use of the terms class, attribute and association and the model itself conform to the UML 2.0 specification.

### AbstractMetric Class

The AbstractMetric class holds the basic information necessary to understand the measurement of a property to be observed, but does not include the additional information to actually use the metric.

#### AbstractMetric Attributes

***definition***

A formal description of the AbstractMetric.

***expression***

The function used to assemble the underlying AbstractMetrics and the ParameterDefinitions that compose the AbstractMetric. RuleDefinitions can also be part of the expression. In its most simple form, the expression is a literal but it can also be a more formal expression language.

*(e.g. expression = Sum(ResponseTime)/n where “ResponseTime” is an underlying AbstractMetric element and “n” is a ParameterDefinition element)*

***name***

The name of the AbstractMetric.

*(e.g. TimeDuration)*

***note***

Additional information or comments related to the AbstractMetric.

***referenceId***

A unique identifier for the abstract metric defined by convention.

*(e.g. AM001)*

***scale***

Information on how the measurement value can be interpreted and what sort of operations can be performed on it. It is based on the theory of scales of measurement [14].

The scale also reflects the kind of the AbstractMetric, i.e. qualitative or quantitative.

**Quantitative** - A metric that has values of numeric type, with the semantics of a quantity. The expression (or formula) that determines how such a value is calculated is of numeric output with a quantitative meaning *(e.g. speed = distance / duration is of quantitative kind)*. The **interval** and **ratio** scales are viewed as quantitative.

**Qualitative** - A metric that has either nominal or ordinal values. When ordinal, the metric usually expresses a “score” *(e.g. on a scale from 1 to 10)*. When nominal, it expresses a quality *(e.g. “good”, “average”, “bad”)*. There is usually an expression (or formula) associated with each possible value, which is of qualitative nature. The **nominal** and **ordinal** scales are viewed as qualitative

Allowed values:

Nominal – Qualitative

Ordinal – Qualitative

Interval – Quantitative

Ratio – Quantitative

***unit***

The unit that will be associated with the AbstractMetric.

*(e.g. second)*

Note that not every AbstactMetric is associated with a scalar unit. For instance AbstractMetrics whose scale is nominal or ordinal (i.e. qualitative) could be associated to a list of elements (e.g. low, medium, high for data sensitivity) or a more complex construct.

#### AbstractMetric Associations

***parameterDefinitions***

An AbstractMetric may have zero or more ParameterDefinitions associated with it. ParameterDefinitions may be part of the expression of an AbstractMetric.

***ruleDefinitions***

An AbstractMetric may have zero or more RuleDefinions associated with it. RuleDefinitions may be part of the expression of an AbstractMetric to constrain it.

***underlyingAbstractMetrics***

Shows any AbstractMetrics that are used as a base for the AbstractMetric being defined. underlyingAbstractMetrics are part of the expression of an AbstractMetric.

### Metric Class

Defines the concrete standard of measurement for a specific cloud service property, It is based on the AbstractMetric concept, adding the specific parameters, and rules which are required to use the AbstractMetric.

#### Metric Attributes

***name***

The name of the Metric.

*(e.g. CustomerResponseTime)*

***note***

Additional information or comments related to the Metric.

***referenceId***

A unique identifier for the metric defined by convention.

#### Metric Associations

***metricParameters***

A Metric is associated with one or more MetricParamters. These MetricParameters are one piece of the implementation of an AbstractMetric through its ParameterDefinitions association.

***metricRules***

A Metric is associated with one or more MetricRules. These MetricRules are one piece of the implementation of an AbstractMetric through its RuleDefinitions association.

***primaryAbstractMetric***

The primary AbstractMetric that the Metric implements.

***underlyingMetrics***

Shows any Metrics that are used as a base for the Metric being described.

### MetricParameter Class

The element that represents a concrete parameter of the Metric based on information from the Metric’s primary AbstractMetric element.

#### MetricParameter Attributes

***note***

Additional information or comments related to the MetricParameter.

***value***

The value of the parameter defined by the associatedParameterDefinition.

*(e.g. value could be 30 for the associated parameterDefinition measurement\_timeframe)*

#### MetricParameter Associations

***parameterDefinition***

A MetricParameter is dependent on a single ParameterDefinition. This ParameterDefinition is selected from the parameterDefinitions of the Metric’s primaryAbstractMetric element.

### MetricRule Class

The element that represents a concrete rule of the Metric based on information from the Metric’s primary AbstractMetric element.

#### MetricRule Attributes

***note***

Additional information or comments related to the MetricRule.

***value***

The value of the rule defined by the associated RuleDefinition.

*(e.g. value could be “scheduled maintenance” for the associated ruleDefinition observation\_exclusion)*

#### MetricRule Associations

***ruleDefinition***

A MetricRule is dependent on a single RuleDefinition. This RuleDefinition is selected from the ruleDefinitions of the Metric’s primaryAbstractMetric element.

### ParameterDefinition Class

A ParameterDefinition element is used to define a parameter needed in the expression of an AbstractMetric. A ParameterDefinition may be used by more than one AbstractMetrics.

#### ParameterDefinition Attributes

***definition***

A formal description of the ParameterDefinition.

***name***

The name of the ParameterDefinition.

*(e.g. measurementTimeframe)*

***note***

Additional information or comments related to the ParameterDefinition.

***referenceId***

A unique identifier for the ParameterDefinition defined by convention.

*(e.g. PD001)*

***parameterType***

The type of the ParameterDefinition, the way it should be interpreted.

*(e.g. integer, string)*

### RuleDefinition Class

A RuleDefinition element is used to further constrain some parts of an AbstractMetric element and indicate possible method(s) for measurement. For instance an “AvailabilityDuringBusinessHour” Metric element could be defined with a scope that constrains some piece of a generic “Availability” AbstractMetric element that limits the observation period to defined business hours.

#### RuleDefinition Attributes

***definition***

A formal description of the RuleDefinition element.

***name***

The name of the RuleDefinition.

*(e.g. whenStart RuleDefinition for a TimeDuration AbstractMetric)*

***note***

Additional information or comments related to the RuleDefinition.

***referenceId***

A unique identifier for the abstract metric defined by convention.

*(e.g. RD001)*

# How to Use the CSM Model

The CSM model defines the fundamental elements needed to describe standards of measurement (i.e. metrics). These elements can be organized into two parts. The first part represents the elements that make up the abstract definition of a particular metric (e.g. Service Availability). This is the abstracted model for a category of metrics similar to a template. The second part represents the elements that make up a specific instantiation of this abstract metric.

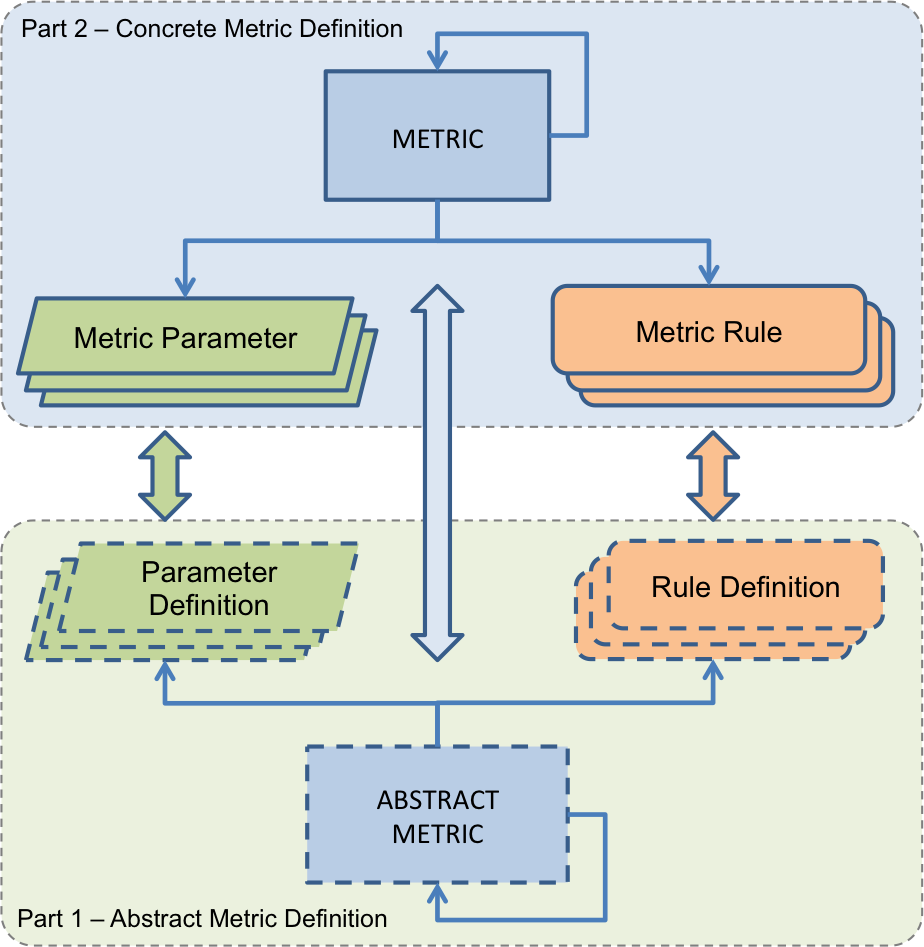


Figure CSM Core Parts

Figure 8 displays another perspective of the CSM diagram to show how the model can be divided into the two parts described earlier. For a given metric category, the abstract metric definition of the metric is composed of AbstractMetric(s) and associated ParameterDefinition(s) and RuleDefinition(s). A concrete metric definition then implements the abstract metric definition with specific MetricParameter(s) and MetricRule(s).

Figure 9 shows from a higher point of view the process followed to define metrics. The CSM model defines the core concepts and elements that constitute a standard of measurement. Specific instances of a subset of these elements (part 1) are then used to create an abstract metric definition. Then for a given abstract metric definition, implementation metrics are created using instances of another subset of the CSM elements (part 2).

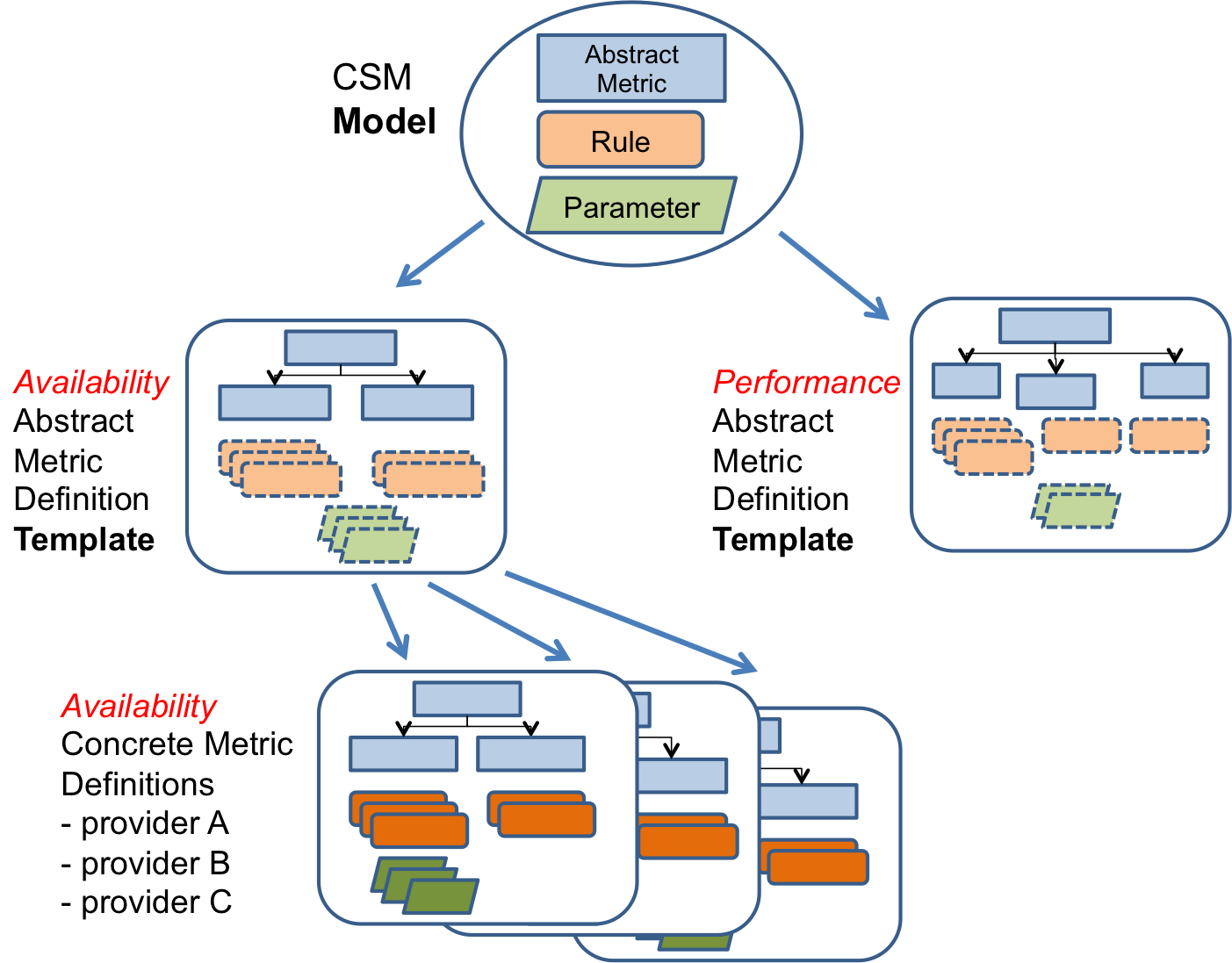


Figure Metric Definition Process

# Measurement Uncertainty

In metrology, the result of a measurement is not meaningful if a statement of the accuracy or uncertainty of the measurement is not specified. This statement allows anyone using the result of a measurement to assess its reliability. This user can be confident to compare results and use them within the range of the measurement accuracy.

The International Vocabulary of Metrology (VIM) [7] defines measurement uncertainty as “A non-negative parameter characterizing the dispersion of the quantity values being attributed to a measurand, based on the information used”

Furthermore:

* Measurement uncertainty includes components arising from systematic effects, such as components associated with corrections and the assigned quantity values of measurement standards, as well as the definitional uncertainty. Sometimes estimated systematic effects are not corrected for but, instead, associated measurement uncertainty components are incorporated.
* The parameter may be, for example, a standard deviation called standard measurement uncertainty (or a specified multiple of it), or the half-width of an interval, having a stated coverage probability.
* Measurement uncertainty comprises, in general, many components. Some of these may be evaluated by Type A evaluation of measurement uncertainty from the statistical distribution of the quantity values from series of measurements and can be characterized by standard deviations. The other components, which may be evaluated by Type B evaluation of measurement uncertainty, can also be characterized by standard deviations, evaluated from probability density functions based on experience or other information.
* In general, for a given set of information, it is understood that the measurement uncertainty is associated with a stated quantity value attributed to the measurand. A modification of this value results in a modification of the associated uncertainty.

In the context of cloud services, it is critical that the consumer of a measured resource be confident about the measurements operated on that resource. These measurements will feed metrics that could be used against thresholds to determine the range the acceptable results and trigger possible consequences.

The current CSM model starts addressing that aspect of metrology with an attribute “uncertainty” that is contained in the CSM Observation model .

# Other Considerations

## Metrics used for Property Composition

A key aspect of the CSM Core diagram is to allow metric definitions to be composed with other metric definitions. This is an effort to limit the duplication of information without too much of an increase on complexity. CSM Core allows metrics of different kind - qualitative or quantitative - to be defined, which in turns means that metrics of different kind can potentially be composed with one another. This can affect the estimation – measurement results – of a particular property in several ways like uncertainty, precision, accuracy etc.

## Calibration & Measurement Standard (Etalon)

Once new metrics and unit of measurements have been defined for cloud service properties that can be reusable and comparable, the next step could be the calibration of the measurement systems used for measurement of cloud service properties against established measurement standards. This would enable a better alignment of the understanding and comparison of the properties that compose different cloud service offerings.

# Conclusion

Metrics are a critical aspect of the selection, operation and use of cloud services. Metrics allow stakeholders to gain a better understanding of cloud service properties through consistent, reproducible and repeatable observations. Metrics can be used for a wide range of objectives from decision making to operation. For instance key performance indicator metrics can be used to measure specific achievements whereas benchmark metrics can be used as reference to compare features against one another.

Metrics need to be well defined and understood so that the different actors involved in cloud computing (i.e. cloud service customer and cloud service provider) can rely on them with confidence. The Cloud Service Metric (CSM) model proposed in this document is one approach to addressing this challenge. The CSM defines a small set of concepts and links them together to define what a metric is, what it is composed of and what constrains its expression. The model can be logically broken down into two parts. The first part addresses the definition of abstract metrics. It specifies what the abstract metric is about, if it composed of underlying abstract metrics, if it is expressed with additional parameters and if there are core rules that constrain it. The second part addresses the definition of concrete metrics. It specifies what primary abstract metric the metric is based on, what values for parameters and rules should be applied to the abstract metric parameters and rules definition.

The CSM model can be extended and integrated into other models that address other aspects of the metric ecosystem like the context of a metric, the observation and measurement results based on a metric or the scenarios that make use of metrics. These other aspects will be explored in future work.

# Annex A - Definitions Survey

Table 1 presents a sample collection of measurement terms and definitions coming from different domains including, information technology, software, software engineering and physical. The terms that were sampled are measure, metric, key performance indicator, benchmark, measurement and measurement unit. As result, the table shows that across and among domains there are many different definitions for the same term. Most of these definitions tend to have the same concepts in their descriptions however a few mix the terms and definitions. For instance the OMG SIMM document defines measure as “a method assigning comparable numerical or symbolic values to entities in order to characterize an attribute of the entities” and measurement as “a numerical or symbolic value assigned to an entity by a measure” and other documents used the same definitions but inverted the terms so in the case of the ISO/IEC 15939 document measurement is defined as “Set of operations having the object of determining a value of a measure” and measure as “variable to which a value is assigned as the result of measurement”.



Table - A sample of measurement related terms and definition in the IT space



# Annex B - References

1. Gray, M., “Applicability of Metrology to Information Technology,” *J. Res. Natl. Stand. Technol.*, Vol. 104,No. 6, pp. 567-578, 1999.

2. Mell, P., Grance, T., “The NIST Definition of Cloud Computing,” *NIST Special Publication 800-145*, 2011.

3. ISO/IEC 20926:2009, Software and systems engineering - Software measurement - IFPUG functional size measurement method.

4. ISO/IEC 19761:2011, Software engineering - COSMIC: A functional size measurement method.

5. ISO/IEC 29881:2010, Information technology - Software and systems engineering - FiSMA 1.1 functional size measurement method.

6. ISO/IEC 15939:2007, Systems and software engineering - Measurement Process.

7. JCGM 200:2012, International vocabulary of metrology - Basic and general concepts and associated terms (VIM) 3rd edition, 2012.

8. Siegel, J., Perdue, J., “Cloud Services Measures for Global Use: The Service Measurement Index (SMI),” *Service Research & Innovation Institute (SRII) Global Conference*, Published by IEEE, San Jose, CA, July 24-27, 2012.

9. NIST Cloud Computing Reference Architecture Contract and SLA (Draft), http://collaborate.nist.gov/twiki-cloud-computing/pub/CloudComputing/RATax\_Jan20\_2012/NIST\_CC\_WG\_ContractSLA\_Deliverable\_Draft\_v1\_7.pdf.

10. CSCC, Practical Guide to Service Level Agreements Version 1.0, http://www.cloudstandardscustomercouncil.org/2012\_Practical\_Guide\_to\_Cloud\_SLAs.pdf.

11. DMTF DSP0263, Cloud infrastructure Management Interface (CIMI) Model and RESTful HTTP-based Protocol, 2012.

12. Mach, R., et al., “Usage Record - Format Recommendation,” *GFD. 98*, 2006.

13. Unified Modeling Language version 2.0, http://www.omg.org/spec/UML/2.0/. Last accessed, November 2, 2012.

14. Stevens, S., “On the Theory of Scales of Measurement,” *Science*, Vol. 103, No. 2684, pp. 677-680, 1946.