Cloud Application Management for Platforms (CAMP) Test Assertions Version 1.1

Committee Specification Draft 01 / Public Review Draft 01

12 February 2014

Abstract:
This document defines the Test Assertions for Version 1.1 of the OASIS Cloud Application Management for Platforms (CAMP) specification. Test Assertions are testable or measurable expressions related to normative statements in a specification for evaluating if an implementation (or part of it) conforms to that specification. These Test Assertions are intended to be used for defining precisely the conditions required by a CAMP implementation in order to conform. They support the testing activity by acting as a bridge between the normative statements in the specification and the executable test cases that are parts of a conformance test suite.

Notices

Table of Contents

1 Introduction

This document defines the Test Assertions for version 1.1 of the OASIS Cloud Application Management for Platforms (CAMP) specification [CAMP]. Test Assertions are testable or measurable expressions related to normative statements in a specification for evaluating if an implementation (or part of it) conforms to this specification. These Test Assertions are intended to be used for defining precisely the conditions required by a CAMP implementation in order to conform. They support the testing activity by acting as a bridge between the normative statements in the specification and the executable test cases that are parts of a conformance test suite.Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in **[RFC2119]**.

1.1 Normative References

[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997. http://www.ietf.org/rfc/rfc2119.txt.
[TA-Model]	*Test Assertions Model Version 1.0*. 15 October 2012. OASIS Standard. http://docs.oasis-open.org/tag/model/v1.0/os/testassertionsmodel-1.0-os.html.
[CAMP]	*Cloud Application Management for Platforms Version 1.1*, February 2014, OASIS Committee Specification Draft 04,. http://docs.oasis-open.org/camp/camp-spec/v1.1/csd04/camp-spec-v1.1-csd04.pdf

1.2 Non-Normative References

[TA-guidelines]	*Test Assertions Guidelines Version 1.0*. June 2013. OASIS Committee Note. http://docs.oasis-open.org/tag/guidelines/v1.0/cn02/guidelines-v1.0-cn02.pdf

2 Test Assertions Design and Objectives

2.1 Testing Objectives

A Test Assertions is a testable or measurable expression for evaluating the adherence of an implementation (or part of it) to a normative statement(s) in a specification.

The Test Assertions presented here are intended to help a precise understanding of what is expected from implementations of the CAMP specification.

They are also intended as a starting point for actual test suites. Testing for conformance to the CAMP specification occurs typically in two major contexts:

1. **Product Conformance assessment**. In this set-up the primary goal is to assess if an implementation behaves in conformance to CAMP. A test driver simulates a Consumer, when the implementation being tested is a CAMP Provider. Conversely, a Consumer may also be tested for conformance, in which case there is no test driver but merely some scenarios that the Consumer is expected to execute along with a Provider. The objective of testing is not always to establish conformance or interoperability. Conformance testing may also include an assessment of which options are or are not supported by an end-point, even if all the mandatory requirements are met.

2. **Product Interoperability test rounds**. In this set-up the ability of several implementations to interoperate with each-other is evaluated. A set of implementations is tested pair-wise based on test scenarios between a Consumer and Provider. Even if the exchanges are successful, they must be validated as conforming to CAMP (successful interoperability does not imply successful conformance to CAMP, and vice-versa.)

In both cases, some Consumer-Provider exchanges are executed according to some pre-defined scenarios. In both cases, these exchanges must be verified for conformance to CAMP.

There is also a 3rd context where testing for CAMP conformance is likely to occur: routine regression testing on systems in production. In this context, real business exchanges between a Consumer and a Provider are being monitored and tested for conformance. The rationale is that Cloud end-points may often be upgraded (patches, revisions) or their operational context be modified (configuration change, recovery, etc.), which might cause a deterioration of their conformance to CAMP. Since it is impractical to stop operations and undergo a full round of testing each time some end-point is upgrading, it is often sufficient to detect as early as possible a regression, whenever it occurs.

2.2 Design Principles

CAMP test assertions are intended to be usable by test suites designed for the various testing contexts and objectives described in the previous section. To achieve this versatility, the following design principles are used:

- The test assertions intended to verify conformance of Provider and Consumer are focusing on messages or artifacts exchanged between these roles. They are designed so that they could – ideally – be evaluated with a log of messages as sole input.

- Each test assertion is designed to apply to a particular pattern of messages (except for those directly targeting the PDP or Plan artifacts). The test assertion does not make any assumption about how to produce this pattern of message: it only describes it. A concrete test suite written from such test assertions could enforce (script) the test scenarios that will produce the expected message pattern, or could just discover and analyze such patterns in a message log.

- Each test assertion references the normative statement(s) in the original specification that it is addressing. The test assertion "interprets" the normative requirement in a testable way – i.e. translates the normative requirements in terms of what is a "correct (or incorrect) message exchange pattern" to be observed between a Consumer and a Provider. Clearly an important aspect of designing test assertions is to derive correct and testable interpretations of the original specification requirements.

- A test assertion may address more than one normative statement, as it is often difficult to design test assertions with the level of granularity fine enough to address, or verify, just one normative statement. When such a test assertion "fails" (i.e. related test case(s) fail), it means that at least one of the referred normative statements has been violated.

- Conversely, in order to address the same normative statement in a well-rounded way, several test assertions may be needed that focus on different aspects or sub-cases of the normative statement. In order to have confidence that a normative statement is satisfied by an implementation, all of the test assertions referring to a normative statement should be exercised successfully (i.e. test case(s) pass).

2.3 Anatomy of a Test Assertion

The structure of test assertions follows the recommendation from OASIS TAG [TA-guidelines].

The general structure used for Test Assertions is as follows:
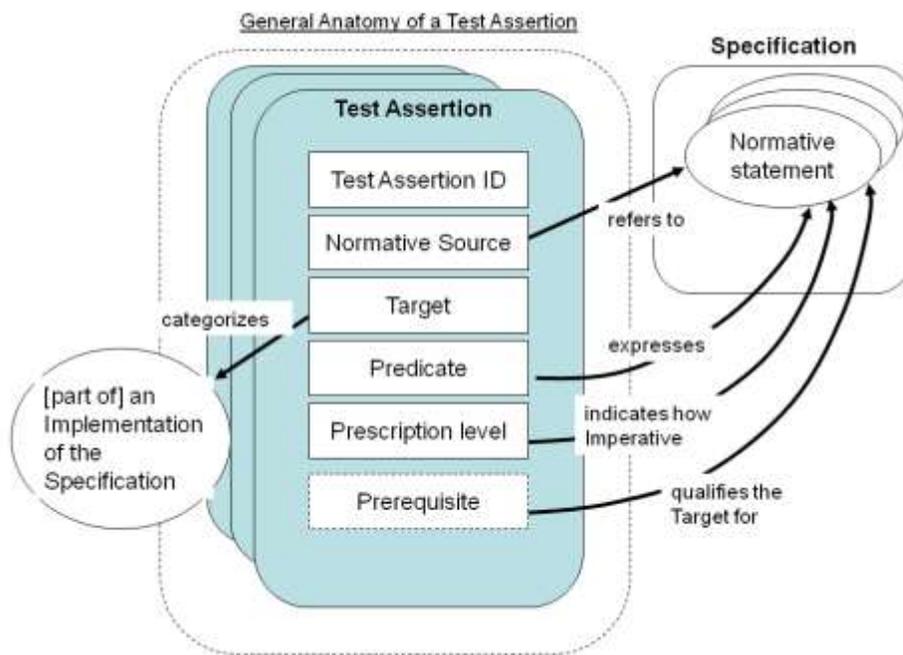
2.3.1 Core Elements

The core elements of a Test Assertion are:

Test Assertion ID (or Identifier, or TA_Id)

> The unique identifier of the test assertion. It facilitates the mapping of assertions to specification statements. It is recommended that the identifier be made universally unique.

Normative Source(s)

> These refer to the precise specification requirement(s) or normative statement(s) that the test assertion addresses.

Target

> The target – or test target - categorizes an implementation or a part of an implementation of the referred specification that is the main object of the test assertion and of its Normative Sources. In CAMP, the target will typically be a message exchanged between Consumer and Provider (e.g. a response to an HTTP GET), or the PDP or Plan artifact.

NOTE: the "*test [assertion] target*" should not be confused with the "*conformance target*". The test target is the artifact or component actually subject to the particular testing described in the test assertion (e.g. a message), while the conformance target is the entity ultimately evaluated for conformance, (e.g. a CAMP Provider implementation). Of course, the test target can be traced to a conformance target.

Predicate

It contains the actual conformance logic. It is worded as a fact (not as a rule or requirement – i.e. no MUST/SHOULD). A predicate asserts, in the form of an expression that evaluates to either "true" or "false", the feature (a behavior or a property) described in the specification statement(s) referred by the Normative Sources, concerning an instance of Target. If the predicate evaluates to "true" over the test assertion Target, this means that the Target exhibits this feature. "False" means the Target does not exhibit this feature.

2.3.2 Optional Elements

The following parts of a test assertion are optional:

Description

An informal definition of the role of the test assertion, with some optional details on some of its parts. This description must not alter the general meaning of the test assertion and its parts. This description may be used to annotate the test assertion with any information useful to its understanding. It does not need to be an exhaustive description of the test assertion.

Prescription Level

A keyword that indicates how imperative it is that the Normative Statement referred to in the Normative Source, be satisfied. Three prescription levels are used:

- **Mandatory**: this value applies to test assertions which address normative statements that include strongly imperative keywords such as "MUST", "REQUIRED", and "SHALL", but also their negative counterparts ("MUST NOT", "SHALL NOT", etc.) as the Predicate is where the negative aspect is expressed.

- **Preferred:** applies to test assertions which address normative statements that can be interpreted as recommendations, using such keywords as "SHOULD" and "RECOMMENDED" as well as their negative counterparts.

- **Permitted:** applies to test assertions about normative statements that concern optional features about which the specification remains neutral (no recommendation) using such keywords as "MAY" or "OPTIONAL". Note that the negative counterparts of these keywords (e.g. "NEED NOT" or "CAN NOT") often have a **mandatory** character.

Prerequisite

A test assertion Prerequisite is a logical expression (similar to a Predicate) which further qualifies the Target for undergoing the core test (expressed by the Predicate) that addresses the Normative Statement. It may include references to the outcome of other test assertions. If the Prerequisite evaluates to "false" then the Target instance is not qualified for evaluation by the Predicate. Prerequisites have also been called "preconditions".

Tags

Test assertions may be assigned 'tags' or 'keywords', which may in turn be given values. These tags allow for categorizing the test assertions.

The above test assertion model is formally described in the OASIS test assertions standard [TA-model].

2.3.3 Example Test Assertion

TA_Id: CAMP11-NNN

NormativeSource: <a normative TAG, and optionally a section name>

Target: A response message to a GET assembly URI message.

Predicate: the target message content satisfies the JSON "schema" for assembly resources.

PrescriptionLevel: mandatory

Tag: conformance=Provider

2.3.4 CAMP Profile

The above test assertion model is profiled for CAMP in the following way:

1. The "**NormativeSource**" element refers to the section(s) in the specification that contain(s) the normative statements addressed by the test assertion. As CAMP has tagged its normative statements, there will be one or more "normative tags" in this field. (Sometimes it does more than make a reference, i.e. makes some "interpretation" of the original statement(s) that can be narrower, and that reflects more concretely the actual test expression). There may be some additional normative tags in parenthesis: this indicates that such normative statements are assumed to be satisfied before using this test assertion – i.e. some kind of prerequisite, or indicating a precedence order for using test assertions.

2. The "**Target**" may be the PDP or the Plan (for PDP and Plan conformance). For conformance of the Provider or Consumer (indicated by the "conformance" tag) , the target will be a particular message pattern (e.g. as expected to be captured in a log) that would "trigger" the verification.

3. The "**Predicate**" -should be evaluated each time a sequence of message appears in the log that satisfies the message pattern described in the Target, and in case the Prerequisite is also satisfied (if any). The Predicate is expressed using any content of the Target message and also contents from the Prerequisite messages if any (see below). Predicate=false means the implementation failed the normative requirement.

4. The "**Prerequisite**" in most cases is a contextual exchange of additional messages between Provider and Consumer that must have occurred along with the target message. These contextual messages must occur often prior to the target message (by default), but sometimes after the target message (this is indicated).

5. Test assertions "**Tags**" (not to be confused with CAMP normative tags in the "normativeSource" field), as accessory information. Only one tag named *conformance* is used in this document. In the above example, the *conformance* tag has value "Provider". This means that the actual entity under evaluation for conformance by this assertion is the Provider (as opposed to the Consumer): the failure of the verification (i.e. predicate value="false") means a failure of the Provider to conform. Note that the conformance target is not always same as the test assertion target: in the above example, the conformance target is "Provider" while the test target is a message.

6. When the same test assertion logic applies to several resource types, a variable will be defined to represent any item over the set of resource types. This amounts to parameterizing the test assertions, so that it does not need to be repeated for each resource. This parameterizing may also apply for other elements when under the same logic.

In the example below, a *resource* variable is defined that can take either one among four resource names (`sensor, plan, assembly, or component`), thus avoiding to write four different test assertions that would only differ by the name of the resource.

```
Var: (resource) in {
    sensor,
    plan,
    assembly,
    component
}
```

Target: A message (m1) of the form PUT *resource* URI (with no `select_attr` query parameter)

2.4 Categories of Test Assertions

Four general categories of test assertions can be distinguished, depending on their conformance target (value of the *conformance* tag):

- **Provider TAs**: addressing normative statements concerning behavior of, and messages generated by the Provider, such as serialization of resources.
- **Consumer TAs**: addressing normative statements concerning behavior of, and messages generated by the Consumer, such as queries, operations on resources.
- **PDP TAs**: addressing normative statements concerning the structure and content of PDP artifacts as well as of their archive format.
- **Plan TAs**: addressing normative statements concerning the structure and content of Plan files.

NOTE: Provider TAs and Consumer TAs may also contain their own tests similar to those found in PDP and Plan TAs. The difference is that these tests will target the messages containing PDPs and Plans (or referring to these) instead of directly the PDP and Plan artifacts. In such cases, the Test Assertions will have an ID that indicates their close relationship. For example:

- **CAMP11-405**, is the ID of a Test Assertions about the Consumer's ability to send a Plan artifact that is correctly formed. Its target is a message referring to such a Plan.
- **CAMP11-405b ,** is the ID of a similar Test Assertions about the Plan artifact itself. It describes the same test concerning the Plan as CAMP11-405 (addressing the same normative statement, using the same predicate). Its target is however the Plan itself, instead of a message containing or referring to the Plan.

2.5 Notational Conventions

2.5.1 Identification scheme for Test Assertions

Test Assertions are identified according to the scheme:

`<specificationID>` "`-`" `<number>`

Where `specificationID` is here "CAMP11", and `number` takes the following ranges:

- For **Model Serialization and Integrity** Test Assertions: $100 <$ `number` < 200
- For **Protocol** Test Assertions: $200 <$ `number` < 300
- For **Resource State Changes and Lifecycle** Test Assertions: $300 <$ `number` < 400
- For **PDP** Test Assertions: $400 <$ `number` < 500

2.5.2 Identifying a message inside a Test Assertion

When referring to an HTTP message, either request or response, the following notation is used:

"`(`" `messageID` "`)`"

where messageID is a symbolic identifier typically of the form. "m1", "m2" etc. Such an identifier is only relative to the test assertion; they are used to distinguish messages from each other in the same test assertion (i.e. this identifier does not appear in the message itself).

2.5.3 Identifying an attribute of a resource in a message

When referring to an attribute of a resource serialized in a message, the `Courier New` font is used to indicate a name proper to the CAMP model. The following notation is used:

"`(`" `messageID` "`)`" `<resource_type>` "`.`" `<attribute_name>`

Where:

- `messageID` is a symbolic message identifier.
- `resource_type` is a reminder of the resource type expected to be found for the top-level serialized resource in this message.
- `attribute_name` is the attribute name of interest for the resource (defined for this `resource_type` and expected to be found on the serialized resource).

For example:

(m2)`platform.assemblies_uri`

2.6 Coverage of the Specification

In this document, only "mandatory" normative statements (SHALL / SHALL NOT) have been addressed.

3 Test Assertions

3.1 Provider Test Assertions

These Test Assertions concern the Provider as conformance target.

3.1.1 Basic Schema Compliance for Platform Resources, Provider Side

3.1.1.1 platform_endpoints (CAMP11-101)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-101

NormativeSource: RE-70, RE-06 (section: platform_endpoints)

Target: A successful response message (HTTP code 200) to a GET `platform_endpoints` message.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `platform_endpoints`, AND has a `type` attribute of value = "`platform_endpoints`".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.1.2 `platform_endpoint` serialization (CAMP11-102)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

Each *platform_endpoint resource* SHALL refer to exactly one *platform resource*, and indicate the versions supported by the Platform. **[RE-20]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-102

NormativeSource: RE-70, RE-20, RE-06 (section: platform_endpoint)

Target: A successful response message (HTTP code 200) to a GET `platform_endpoint` message (m1).

Prerequisite: the URI used in message (m1) exists as a link in `platform_endpoints.platform_endpoint_links`[] as result to a previous GET `platform_endpoints` URI.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `platform_endpoint`, AND has a `type` attribute of value = "`platform_endpoint`" AND non-empty `platform_uri,  implementation_version, specification_version` attribute values.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.1.3 `platform` Serialization (CAMP11-103)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-103

NormativeSource: RE-70, RE-06, (RE-53) (Section: `platform` Resource definition)

Target: A successful response message (HTTP code 200) to a GET `platform` message.

Predicate: the target message content satisfies the serialization schema (JSON) for `platform`, AND has a `type` attribute of value = "platform".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.1.4 service Serialization (CAMP11-104)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-104

NormativeSource: RE-70, (RE-06, RE-53) (Section: `service` Resource definition)

Target: A successful response message (HTTP code 200) to a GET `service` message (m1).

Prerequisite: the URI used in message (m1) exists as a link in `services.service_links` where `services` is obtained from `platform.services_uri` as shown in the result to a previous GET `platform` URI message.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `service` , AND has a `type` attribute of value = "service".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.2 Basic Schema Compliance for Application Resources

3.1.2.1 `assembly` Serializations (CAMP11-111)[CHG]

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-111

NormativeSource: RE-70, RE-06, (RE-53) (Section: `assembly` Resource definition)

Target: A successful response message (HTTP code 200) to a GET `assembly` message (m1).

Prerequisite: the URI used in the message (m1) exists as a link in `assemblies.assembly_links`, as shown in the result to a previous GET `assemblies` message (obtained from `platform.assemblies_uri`), OR there is a POST to an `assemblies` that returns this URI.

Predicate: the target message content satisfies the serialization schema (JSON) for `assembly`, AND has a `type` attribute of value = "assembly".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.2.2 `component` Serialization (CAMP11-112)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-112

NormativeSource: RE-70, RE-06, (RE-53,) (Section: `component` Resource definition)

Target: A successful response message (HTTP code 200) to a GET `component` message (m1).

Prerequisite: the URI used in the message (m1) exists as a link in `assembly.components[]` as shown in the result to a previous GET `assembly` message.

Predicate: the target message content satisfies the serialization schema (JSON) for `component` AND has a link `assembly` to the parent `assembly` URI , AND has a `type` attribute of value = "component".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.2.3 `plan` Serializations (CAMP11-113)

Addressing normative statements:

The schema of the *plan resource* returned from a CAMP Provider SHALL conform to the schema for Plans described in Section 4.3, "Plan Schema", with the following additional requirements: **[RMR-07]**

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-113

NormativeSource: RMR-07, RE-70, RE-06, (RE-53) (Section: `plan` Resource definition)

Target: A successful response message (HTTP code 200) to a GET `plan` message (m1).

Prerequisite: the URI used in the message (m1) exists in a link in `plans.plan_links` as shown in the result to a previous GET `plans` message (obtained from `platform.plans_uri`), OR there is a POST to an `plans` Resource that returns this URI.

Predicate: the target message content satisfies the serialization schema (JSON) for `plan`, AND has a `type` attribute of value = "plan".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.3 Consistency of Serialization with Metadata

3.1.3.1 Consistent Use of Formats (CAMP11-114)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

Providers SHALL respond in the requested Supported Format. **[PR-05]**

TA_Id: CAMP11-114

NormativeSource: RE-70, PR-05, (section: Media Types / Supported Formats)

```
Var: (resource) in {
    assembly,
    component,
    platform,
    service
}
```

Target: A response message (m4) to a GET *resource* message (m3).

Prerequisite:

- A successful response message (m2) to a previous GET `platform.supported_formats_uri` message (m1).

- The negotiated format (using the HTTP Accept request header) in the GET *resource* message (m3), is defined in (m2).

Predicate: in the target message (m4), the resource data in case of a successful response message (HTTP code 200), or the error data in case of an error code (4xx or 5xx), is serialized consistently with the requested format.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.3.2 Consistent Use of Type_definition names (CAMP11-150)

Addressing normative statement:

The value of the `name` attribute in a *type_definition resource* SHALL match the value of the `type` attribute for the resource type that it describes. **[RE-75]**

TA_Id: CAMP11-150

NormativeSource: RE-75, (section: `type_definition` resources)

```
Var: (resource) in {
    assembly,
    component,
    platform,
    service
}
```

Target: A response message (m4) to a GET *resource* message (m3).

Prerequisite:

A successful response message (m2) to a previous GET `type_definitions` (m1) where `type_definitions` is obtained from `platform.type_definitions_uri`.

Predicate: in the target message (m4), the resource data in case of a successful response message (HTTP code 200) shows a `type` attribute with a value that is same as the `name` attribute in one of the `type_definition` items in (m1), AND each attribute of the *resource* in (m4) is also described by an

`attribute_definition` resource referred in the
`type_definition.attribute_definition_links`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.3.3 Inheritance from `camp_resource` (CAMP11-151)

Addressing normative statement:

All CAMP resources SHALL inherit directly or indirectly from this resource. **[MO-05]**

TA_Id: CAMP11-151

NormativeSource: MO-05, (section: `camp_resource` resource)

Target: A successful response message (m2) to a previous GET `type_definitions` (m1) where `type_definitions` is obtained from `platform.type_definitions_uri`.

Prerequisite:

- A successful response message (m4) to a previous GET `type_definition` (m3) where the `name` attribute value shown in (m4) is `camp_resource`.

Predicate: Every link in `type_definitions.type_definition_links` uses an URI that refers to a `type_definition` resource that satisfies either condition: (the `name` attribute has value `camp_resource` OR the `inherits_from` attribute has a link showing same URI as the URI used in (m3).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.3.4 Non-Mutability Enforcement (Provider) (CAMP11-115)

Addressing normative statement:

This boolean indicates the mutability of the attribute's value(s). "false" indicates that the value of the attribute, once set, SHALL NOT change for the lifetime of the resource. **[RE-07]**

TA_Id: CAMP11-115

NormativeSource: RE-07, (RE-53) (section: Mutable)

```
Var: (resource) in {
    assembly,
    component,
    plan
}
```

Target: A response message (m4) to a PUT request (m3) to a *resource*.

Prerequisite: The following message sequence occurred:

- A previous message (m1) of the form GET *resource* was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).
- The PUT request (m3) is providing different value than shown in (m2) for a consumer-mutable=false attribute.

Predicate: the target message (m4) has an 4xx HTTP code.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4 Metadata Integrity and Serialization

3.1.4.1 `parameters_definitions` Serialization and Integrity (CAMP11-117)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-117

NormativeSource: RE-70, RE-06, (RE-53) (section: parameter_definitions)

Target: A successful response message (HTTP code 200)(m4) to a GET `parameter_definitions` message (m3).

Prerequisite: The following message sequence occurred:

- A successful response (HTTP 200) message (m2) to a GET `assemblies` message (m1)
- The GET `parameter_definitions` (m3) is on the URI returned as (m2)`assemblies.parameter_definitions_uri`.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `parameter_definitions`, and has a `type` attribute of value = "parameter_definitions"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.2 `parameters_definition` Serialization and Integrity (CAMP11-118)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

If the Required boolean constraint for an attribute of a resource type has a value of "true", then a resource of this type SHALL have the attribute present. **[RE-06]**

TA_Id: CAMP11-118

NormativeSource: RE-70, RE-06, (RE-53,) (section: `parameter_definition`)

Target: A successful response message (HTTP 200) (m6) to a GET `parameter_definition` message (m5).

Prerequisite: The following message sequence occurred:

- A successful response (HTTP 200) message (m2) to a GET `assemblies` message (m1)
- A successful response (HTTP 200) message (m4) to a GET `parameter_definitions` (m3) on the URI returned by (m2) as `assemblies.parameter_definitions_uri`.
- The GET `parameter_definition` (m5) is against on one of the URIs returned in (m4) `parameter_definitions.parameter_definition_links`

Predicate: the target message (m6) content satisfies the serialization schema (JSON) for `parameter_definition`, AND has a `type` attribute of value = "parameter_definition"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.3 `formats` Serialization (CAMP11-119)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-119

NormativeSource: RE-70 (section: `formats`)

Target: A successful response message (HTTP code 200) to a GET `formats` message (m1).

Prerequisite: the URI used in the message (m1) exists as value of a `platform.supported_formats_uri` attribute as result to a previous GET `platform` message.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `formats`, and has a `type` attribute of value = "formats".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.4 `format` Serialization (CAMP11-120)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-120

NormativeSource: RE-70 (section: `format`)

Target: A successful response message (HTTP code 200)(m4) to a GET `format` message (m3).

Prerequisite: the URI used in the message (m3) exists as an item in `formats.format_links` as shown in a result to a previous GET `formats` message (m2), itself obtained from a previous GET `platform` message (m1) as `platform.supported_formats_uri`.

Predicate: the target message (m4) content satisfies the serialization schema (JSON) for `format`, and has a `type` attribute of value = "format".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.5 Presence of JSON Format (CAMP11-121)

Addressing normative statement:

The *name*, *mime_type*, *version*, and *documentation* attribute values for the JSON Format Resource SHALL reflect the above values. **[RE-42]**

TA_Id: CAMP11-121

NormativeSource: RE-42 (section: Required JSON Format Resource)

Target: A successful response message (HTTP code 200)(m4) to a GET `format` message (m3) with query for "name"="JSON".

Prerequisite: the URI used in the message (m3) exists as an item in `formats.format_links` as a result to a previous GET `Formats` message (m2), itself obtained from a previous GET `platform` message (m1) as `platform.supported_formats_uri`.

Predicate: the target message (m4) has: "`mime_type`", "`version`", "`documentation`" set respectively to the expected values ("application/json", "RFC4627", "http://www.ietf.org/rfc/rfc4627.txt") .

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.6 Position of JSON Format (CAMP11-122)

Addressing normative statement:

The Required JSON Format Resource SHALL be listed first in the format_links array. **[RE-41]**

TA_Id: CAMP11-122

NormativeSource: RE-41 (section: format_links)

Target: A successful response message (HTTP code 200)(m2) to a GET `formats` message (m1).

Prerequisite: the URI used in the message (m1) exists as `platform.supported_formats_uri` value, as a result to a previous GET `platform` message.

Predicate: the target message (m2) has its first link item with "`target_name`" set to "JSON".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.7 `type_definitions` Serialization (CAMP11-123)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-123

NormativeSource: RE-70, (RE-43) (section: `type_definitions` Resource)

Target: A successful response message (HTTP code 200)(m2) to a GET `type_definitions` message (m1).

Prerequisite: the URI used in the message (m1) exists as `platform.type_definitions_uri` in a result to a previous GET `platform` message.

Predicate: the target message (m2) content satisfies the serialization schema (JSON) for `type_definitions`, AND has a `type` attribute of value = "type_definitions".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.8 `type_definition` Serialization (CAMP11-124)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-124

NormativeSource: RE-70 (section: Resources)

Target: A successful response message (HTTP code 200) to a GET `type_definition` message (m1).

Prerequisite: the URI used in the message (m1) exists as an item in `type_definitions. type_definitions_links` as a result to a previous GET `type_definitions` message, itself obtained from `platform.type_definitions_uri` in a previous GET Platform message.

Predicate: the target (response) message content satisfies the serialization schema (JSON) for `type_definition`, AND has a `type` attribute of value = "type_definition".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.4.9 `attribute_definition` Serialization (CAMP11-125)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-125

NormativeSource: RE-70 (section: Resources)

Target: A successful response message (HTTP code 200)(m2) to a GET `Attribute_definition` message (m1).

Prerequisite: the URI used in the message (m1) exists as an item in Type_definition. attribute_definition s_links as a result to a previous GET Type_definition.

Predicate: the target message (m2) content satisfies the serialization schema (JSON) for `Attribute_definition` , AND has a `type` attribute of value = "attribute_definition ".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5 Various Model Referential and Semantic Constraints

3.1.5.1 `type_definitions` reference in `platform` Resource (CAMP11-126)

Addressing normative statement:

The *platform resource* SHALL provide a Link to the *type_definitions resource* in the required attribute named `type_definitions_uri`. **[RE-43]**

TA_Id: CAMP11-126

NormativeSource: RE-43, (section: `type_definitions` Resource)

Target: A successful response message (HTTP code 200)(m2) to a GET `platform` message (m1).

Predicate: the target message (m2) content has :

- an attribute `platform.type_definitions_uri`, that contains a valid URI referring to a `type_definitions` Resource.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.2 `platform_endpoint` specification_version (CAMP11-127)

Addressing normative statements:

For Platforms that implement this version of the CAMP specification, the value of this attribute SHALL be "CAMP 1.1" as defined in Section 1.8, "Specification Version". **[RE-22]**

platform_endpoint resources that reference Specification Version "CAMP 1.1" Platform Resources SHALL NOT include this attribute because no previous versions are compatible. **[RE-24]**

TA_Id: CAMP11-127

NormativeSource: RE-22, RE-24 (sections: `platform_endpoint` Resource / `specification_version`, / `backward_compatible_specification_versions`)

Target: A successful response message (HTTP code 200)(m2) to a GET `platform_endpoint` message (m1).

Prerequisite: the URI used in the message (m1) exists as a link in `platform_endpoints.platform_endpoint_links`[] as shown in the result to a previous GET `platform_endpoints` URI.

Predicate: the target message (m2) content has a `specification_version` attribute = "CAMP 1.1" AND does not contain an `backward_compatible_specification_versions` attribute.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.3 Platform `specification_version` (CAMP11-128)

Addressing normative statement:

For Platforms that implement this version of the CAMP specification, the value of this attribute SHALL be as defined in Section 1.8, "Specification Version". **[RE-26]**

TA_Id: CAMP11-128

NormativeSource: RE-26 (section: `Platform` / `specification_version`)

Target: A successful response message (HTTP code 200)(m2) to a GET `platform` message (m1).

Predicate: the target message (m2) content has a `specification_version` attribute = "CAMP 1.1".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.4 `platform_endpoint` and `platform` matching `specification_version` (CAMP11-129)

Addressing normative statement:

The value of this attribute SHALL exactly match the value of the `specification_version` attribute of any *platform_endpoint resource* that references this *platform resource*. **[RE-27]**

TA_Id: CAMP11-129

NormativeSource: RE-27 (section: `Platform` / `specification_version`),

Target: A successful response message (HTTP code 200) (m4) to a GET `platform` message (m3).

Prerequisite: The sequence of messages occurred:

- A successful response message (HTTP code 200) (m2) to a GET `platform_endpoint` message.
- The URI in GET `platform` message (m3), is same as the `platform_endpoint.platform_uri` value in (m2)

Predicate: the target (response) message (m4) content has a `platform.specification_version` attribute = (m2)`platform_endpoint.specification_version`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.5 `platform_endpoint` and `platform` matching `implementation_version` (CAMP11-130)

Addressing normative statement:

The value of this attribute SHALL exactly match the value of the `implementation_version` attribute of any *platform_endpoint resource* that references this *platform resource*. **[RE-29]**

TA_Id: CAMP11-130

NormativeSource: RE-29 (section: `Platform / implementation_version`),

Target: A successful response message (HTTP code 200) (m4) to a GET `platform` message (m3).

Prerequisite: The sequence of messages occurred:

- A successful response message (HTTP code 200) (m2) to a GET `platform_endpoint` message,
- The URI in GET `platform` message (m3), is same as the `platform_endpoint.platform_uri` value in (m2)

Predicate: the target message (m4) content has a `platform.implementation_version` attribute = (m2)`platform_endpoint.implementation_version`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.6 Cross-references `platform_endpoints` and `platform` (CAMP11-131)[REN]

Addressing normative statements:

This attribute is an array of Links to *platform_endpoint resources*. This array SHALL contain at least one Link. **[RE-18]**

References between the resources (*platform_endpoints*, *platform_endpoint*, and *platform*) SHALL be self-consistent. **[RE-19]**

TA_Id: CAMP11-131

NormativeSource: RE-19, RE-18 (section: `platform_endpoint_links`),

Target: A successful response (HTTP 200) message (m2) to a GET `platform` message (m1)

Prerequisite: The following message sequence occurred:

- A successful response message (HTTP code 200)(m4) to a GET `platform_endpoints` message (m3).

Predicate: there is at least one link in `platform_endpoints.platform_endpoint_links` that refers to a `platform_endpoint` resource with `platform_endpoint.platform_uri` attribute value same as the URI used in (m1).AND the (m2) `platform.platform_endpoints_uri` value is same URI as the URI used in (m3).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.7 Parameter Definition for `pdp_uri` or `plan_uri in assemblies` (CAMP11-132)

Addressing normative statement:

The *assemblies resource* SHALL indirectly reference *parameter_definition resources* that describes the pdp_uri, plan_uri, pdp_file, and plan_file parameters. **[RMR-03]**

TA_Id: CAMP11-132

NormativeSource: RMR-03 (section: `Assemblies / parameter_definitions_uri`),

Target: A successful response message (HTTP code 200) (m2) to a GET `assemblies` message (m1), with a `representation_skew`=NONE.

Prerequisite: The following message sequence occurred after (m2):

- A successful response (HTTP 200) message (m4) to a GET `parameter_definitions` message (m3) using the (m2) `assemblies.parameter_definitions_uri` URI.

Predicate: There exists at least four link items in (m4)`parameter_definitions.parameter_definitions_links`, that describe the "`pdp_uri`", "`plan_uri`", "`pdp_file`" and "`plan_file`" parameters, i.e. for which respectively a GET `parameter_definition` message (m5) returns a successful response (m6) with `parameter_definition.name` = "pdp_uri", "plan_uri", "pdp_file" and "plan_file".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.8 Parameter Definition for `pdp_uri` or `plan_uri` in `plans` (CAMP11-152)

Addressing normative statement:

The *plans resource* SHALL indirectly reference *parameter_definition resources* that describe the pdp_uri, plan_uri, pdp_file, and plan_file parameters. **[RMR-06]**

TA_Id: CAMP11-152

NormativeSource: RMR-06 (section: `plans` Resource),

Target: A successful response message (HTTP code 200) (m2) to a GET `plans` message (m1), with a `representation_skew`=NONE.

Prerequisite: The following message sequence occurred after (m2):

- A successful response (HTTP 200) message (m4) to a GET `parameter_definitions` message (m3) using the (m2) `plans.parameter_definitions_uri` URI.

Predicate: There exists at least four link items in (m4)`parameter_definitions.parameter_definitions_links`, that describe the "`pdp_uri`", "`plan_uri`", "`pdp_file`" and "`plan_file`" parameters, i.e. for which respectively a GET `parameter_definition` message (m5) returns a successful response (m6) with `Parameter_definition.name` = "`pdp_uri`", "`plan_uri`", "`pdp_file`" and "`plan_file`".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.9 Required Parameter Definitions in `service` (CAMP11-134)

Addressing normative statement:

The *parameter_definitions* resource referenced by this attribute SHALL define parameters to allow setting the 'name', 'description', and 'tags' attributes of any new resource created in the course of interacting with this resource. **[RE-37]**

TA_Id: CAMP11-134

NormativeSource: RE-37 (section: `Service / parameter_definitions_uri`),

Target: A successful response message (HTTP code 200) (m2) to a GET `service` message (m1), with a `representation_skew`=NONE.

Prerequisite: The following message sequence occurred after (m2):

- A successful response (HTTP 200) message (m4) to a GET `parameter_definitions` message (m3) using the (m2) `service.parameter_definitions_uri` URI.

Predicate: There exists link items in (m4)`parameter_definitions.parameter_definitions_links`, for which a GET `parameter_definition` message (m5) returns a successful response (m6) with `parameter_definition.name` = "name", AND another (m5') returns `parameter_definition.name` = "description", AND another (m5'') returns `parameter_definition.name` = "tags".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.10   UTC Timestamps, Provider side (CAMP11-136)

Addressing normative statement:

Consumers and Providers SHALL express Timestamps in UTC (Coordinated Universal Time), with the special UTC designator ("Z"). [RE-65]

TA_Id: CAMP11-136

NormativeSource: RE-65 (section: Timestamp)

Target: A successful response message (HTTP code 200)(m2) that has a resource serialization showing the "`timestamp`" attribute, in response to a GET message (m1).

Predicate: the Timestamp value conforms to the UTC (Universal Time Coordinated) with a "zero" timezone offset, i.e. with format terminated with the designator "Z", of the form: yyyy '-' mm '-' dd 'T' hh ':' mm ':' ss 'Z' (Example: `1994-11-05T13:15:30Z`)

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.11 `representation_skew` Content (CAMP11-137)

Addressing normative statement:

If present, representation_skew SHALL have one of the following values: [RE-11]

TA_Id: CAMP11-137

NormativeSource: RE-11, (section: `representation_skew`)

Target: A successful response message (HTTP code 200) message to a GET resource message.

Prerequisite: The `representation_skew` attribute is present.

Predicate: the value of `representation_skew` is among { CREATING, NONE, UNKNOWN, DESTROYING }

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.12 References to `format` Resources associated with `a platform` (CAMP11-140)

Addressing normative statement:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. [RE-70]

TA_Id: CAMP11-140

NormativeSource: RE-70

Target: A successful response message (HTTP code 200) (m4) to a GET `formats` message (m3) with URI same as in (m2) `platform.supported_formats_uri`

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1)

Predicate: For each link in (m4) `formats.format_links`, a GET `format` on the URI referenced in the link, returns an HTTP code 2xx AND a `type` attribute = "format".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.13 Presence of `format` Resources for every supported format (CAMP11-149)

Addressing normative statement:

For every format that the Platform supports, there SHALL be a Format resource Link that represents such a format. **[RE-40]**

TA_Id: CAMP11-149

NormativeSource: RE-40 (section: `Formats` Resource / `format_links`)"

Target: A successful response message (HTTP code 200) (m4) to a GET message (m3) with HTTP Content-type in (m4) other than "application/json"

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1) where this resource (m4) is defined.

Predicate: There exists a link in (m4) `formats.format_links`, where `formats` is referred to by (m2) `platform.supported_formats_uri` , that refers to a `format` resource with `mime_type` attribute value same as Content-type in (m4).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.14 Presence of `type_definition` Resources associated with a `platform` (CAMP11-141)

Addressing normative statement:

If the array is non-empty, for every resource type that the Platform supports, there SHALL be a *type_definition resource* Link that represents such a resource type. **[RE-44]**

TA_Id: CAMP11-141

NormativeSource: RE-44 (section: `type_definitions` Resource / `type_definitions_links`)

Target: A successful response message (HTTP code 200) (m4) to a GET *resource* message (m3) of any kind.

Prerequisite:

A successful response message (HTTP code 200) (m2) to a GET `type_definitions` message (m1) with URI same as in `platform.type_definitions_uri,` showing a `type_definition_links` array with at least one link.

Predicate: There exists a link in (m2) `type_definitions.type_definition_links`, that references a `type_definition` resource with a `name` attribute value same as the `type` value in the resource returned in (m4).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.15 Inheritance of `type_definition` Resources (CAMP11-148)

Addressing normative statement:

Links in this array SHALL NOT either directly or transitively point to the described resource. [MO-06]

TA_Id: CAMP11-148

NormativeSource: MO-06 (section: `type_definition` Resource / `inherits_from`)

Target: A successful response message (HTTP code 200) (m4) to a GET `type_definition` message (m3).

Prerequisite: The `type_definition` URI belongs to `type_definitions.type_definition_links` as obtained from `platform.type_definitions_uri` in previous message exchanges.

Predicate: If the `inherits_from` link is set in (m4) then every link present in it satisfies: (the link refers to a Resource of type `type_definition` that has an attribute "type" of value "typeDefinition" AND an attribute "`name`" value different from the "`name`" value in (m4) AND an attribute "`uri`" value different from the URI used in (m3).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.16 Presence of Attribute_definition Resources associated with a Platform (CAMP11-142)

Addressing normative statement:

For every attribute of the type not inherited from its super-types, there SHALL be an AttributeLink that references the *attribute_definition resource* that defines that attribute. [RE-45]

TA_Id: CAMP11-142

NormativeSource: RE-45 (section: `Type_definition` Resource / `attribute_definitions_links`)"

Target: A successful response message (HTTP code 200) (m6) to a GET `type_definition` message (m5), on a URI present in (m4) `type_definitions.type_definitions_links`

Prerequisite: A sequence of messages:

- A successful response (m2) to a previous GET `platform` message (m1)
- A successful response (m4) to a previous GET `type_definitions` message (m3) on the URI `platform.type_definitions_uri`.

Predicate: For each attribute of the `type_definition` in (m6) , there exists a link in `type_definition.attribute_definitions_links`, for which a GET `attribute_definition` on the URI referenced in the link, returns an HTTP code 2xx AND a `type` attribute = "attribute_definition " AND a `name` attribute = the attribute name

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.17 Presence of Extension Resources associated with a `platform` (1) (CAMP11-143)

Addressing normative statement:

For every extension available, there SHALL be an *extension resource* Link that represents the extension. **[EX-08]**

TA_Id: CAMP11-143

NormativeSource: EX-08 (section: `formats` Resource / `format_links`)

Target: A successful response message (HTTP code 200) (m6) to a GET `extension` message (m5) i.e. where (m6) shows an `type` attribute of value "extension".

Prerequisite: A previous sequence of messages such as:

- A successful response (m2) to a previous GET `platform` message (m1)
- A successful response (m4) to a previous GET `extensions` message (m3) where the URI is same as (m2)`platform.extensions_uri`.

Predicate: There exists a link in (m4) `Extensions`.extension_links, with a reference URI same as the URI used in (m5).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.18 Presence of `extension` Resources associated with a `platform` (2) (CAMP11-144)

Addressing normative statements:

For every extension available, there SHALL be an *extension resource* Link that represents the extension. **[EX-08]**

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-144

NormativeSource: EX-08, RE-70 (section: `Formats` Resource / `format_links`)

Target: A successful response message (HTTP code 200) (m4) to a GET `extensions` message (m3) with URI same as in (m2) `platform.extensions_uri`

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1)

Predicate: For each link in (m4) `extensions.extension_links`, a GET `extension` message on the URI referenced in the link returns an HTTP code 2xx AND shows an `type` attribute of value "extension".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.19 Presence of `extensions` Resource associated with a platform (CAMP11-145)

Addressing normative statement:

The *platform resource* SHALL provide a Link to the *extensions resource* in the required attribute named `extensions_uri`. **[EX-09]**

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

TA_Id: CAMP11-145

NormativeSource: EX-09, RE-70 (section: `Formats` Resource / `format_links`)

Target: a successful response message (HTTP code 200) (m4) to a GET `extensions` message (m3) with URI same as in (m2) `platform.extensions_uri`.

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1)

Predicate: The message (m4) has HTTP code 2xx, AND the returned serialization shows a `type` attribute of value "`extensions`" AND the serialization is conforming to the `extensions` schema.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.20 Unique Name of `extension` Resources associated with a `platform` (CAMP11-146)

Addressing normative statement:

Entities added by an extension SHALL NOT interfere with names of existing entities, including any added by another extension. **[EX-06]**

TA_Id: CAMP11-146

NormativeSource: EX-06, (EX-08) (section: Unique Name Requirement)

Target: A successful response message (m4) to a GET `extensions` message (m3) with URI same as in (m2) `platform.extensions_uri`

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1)

Predicate: For any pair of links in (m4) `extensions.extension_links`, the `target_name` values in these links are different AND different from any predefined resource or attribute name in the CAMP specification.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.5.21 Presence of an `extension` Resource advertising support for `plans` (CAMP11-147)

Addressing normative statement:

Providers that support the *plans* and *plan resources* SHALL advertise such support using the following *extension resource*: **[RMR-12]**

TA_Id: CAMP11-147

NormativeSource: RMR-12 (section: Advertising Support for the Plan Resource)

Target: A successful response message (HTTP code 200) (m4) to a GET `extensions` message (m3) with URI same as in (m2) `platform.extensions_uri`

Prerequisite: A successful result (m2) to a previous GET `platform` message (m1), shows an `plans_uri` attribute set to a valid URI.

Predicate: There exists a link in (m4) `extensions.extension_links`, for which a GET `extension` message on the URI referenced in the link returns an `extension` that shows a `name` attribute of value "CAMP Plans Extension".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6 Support for HTTP Methods

3.1.6.1 GET method (CAMP11-201)

Addressing normative statements:

When supporting such a Resource, a Provider SHALL implement it and serialize it as described in the corresponding sub-section. **[RE-70]**

Providers SHALL support the HTTP GET, PUT, and PATCH methods on all of the resources defined in this section. **[RE-53]**

TA_Id: CAMP11-201

NormativeSource: RE-70, RE-53, (section: `platform_endpoints` resource)

Target: A response message to a GET `platform_endpoints` message.

Predicate: the target message has HTTP code 200 AND its content is valid for `platform_endpoints` serialization schema (JSON).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6.2 PDP Registration by reference, Provider side (CAMP11-410)

Addressing normative statement:

Providers SHALL support the deployment of applications via HTTP POST requests on the *assemblies resource* as described in this section. **[PR-49]**

TA_Id: CAMP11-410

NormativeSource: PR-49 (section: Registering an Application by Reference)

Target: A response (m2) to a POST message (m1) to an `assemblies` URL, with a "`pdp_uri`" or a "`plan_uri`" attribute.

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the new `assembly` resource, OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK")

Predicate: the HTTP code in target (m2) is NOT 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6.3 Supported HTTP Post for PDP or Plan Registration by Reference: Provider side (CAMP11-435)

Addressing normative statements:

Providers that support the *plans resource* and *plan resources* SHALL support the registration of Plans via an HTTP POST request on the *plans resource* as described in this section. **[PR-56]**

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP POST method on the *plans resource* as described in Section 6.12, "Registering a Plan". **[RMR-05]**

TA_Id: CAMP11-435

NormativeSource: PR-56, RMR-05

Target: A response message (m2) to a POST `plans` message (m1) where (m1) contains a "pdp_uri" or a "plan_uri" attribute.

Predicate: the HTTP code in target (m2) is NOT 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6.4 Supported HTTP Post for PDP or Plan Registration by value: Provider side (CAMP11-436)

Addressing normative statement:

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP POST method on the *plans resource* as described in Section 6.12, "Registering a Plan". **[RMR-05]**

Providers that support the *plans resource* and *plan resources* SHALL support the registration of Plans via HTTP POST requests on the *plans resource* in which the entity body of the request contains the PDP or the Plan file that is being registered. **[PR-61]**

TA_Id: CAMP11-436

Var: *archtype* in { application/x-zip, application/x-tar, application/x-tgz, application/x-yaml }

NormativeSource: PR-61, RMR-05

Target: A response message (m2) to a POST `Plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type = *archtype*.

Predicate: the HTTP code in target (m2) is NOT 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6.5 Supported HTTP Post for Application Deployment by value: Provider side (CAMP11-434)

Addressing normative statement:

Providers SHALL support the deployment of applications via HTTP POST requests on the *assemblies resource* in which the entity body of the request contains the PDP or the Plan file that is being deployed. **[PR-60]**

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP POST method on the *assemblies resource* as described in Section 6.11, "Deploying an Application". **[RMR-02]**

TA_Id: CAMP11-434

Var: *archtype* in { application/x-zip, application/x-tar, application/x-tgz, application/x-yaml }

NormativeSource: PR-60, RMR-02

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type = *archtype*.

Prerequisite: enough time elapsed to allow for the creation to complete.

Predicate: the HTTP code in target (m2) is NOT 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.6.6 HTTP POST support for `operation` Resource (1) (CAMP11-311)

Addressing normative statement:

The Provider SHALL update the `plan_links` attribute of the *plans resource* to include a reference to the newly created *plan resource*. **[PR-64]**

TA_Id: CAMP11-311

NormativeSource: RE-64 (section: `operation` Resource)

Target: A response message to a POST `operation` message.

Prerequisite: the Operation URI exists as an item in `operations.operation_links[]` as shown in the result to a previous GET `operations` URI, itself obtained from a GET `assembly.operations_uri`.

Predicate: the HTTP code in target (m2) is NOT 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.7 Support for JSON

3.1.7.1 JSON media-type for Resources (1) (CAMP11-202)

Addressing normative statement:

Providers SHALL provide representations of all available resources in JSON. **[PR-01]**

TA_Id: CAMP11-202

NormativeSource: PR-01 (section: Required Formats)

Target: A successful response message (HTTP code 200)(m2) to a GET resource message (m1) that has no HTTP "Accept" header or has one set to "application/json".

Predicate: In the target (m2) the HTTP Content-Type value is "application/json", AND the content if any is valid JSON.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.7.2 JSON media-type for Deployments of Applications (1) (CAMP11-220)

Addressing normative statement:

To support the deployment of applications via a reference to either a PDP, Plan file, or *plan resource*, Providers SHALL accept the "application/json" media type. **[PR-68]**

TA_Id: CAMP11-220

NormativeSource: PR-68 (section: Deploying an Application by Reference)

Target: A response message (m2) to a POST `assemblies` message (m1) that has no HTTP "Accept" header or has one set to "application/json" and contains a "pdp_uri" or a "plan_uri" attribute.

Predicate: The HTTP code of (m2) is NOT code 415, "Unsupported Media Type".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.7.3 JSON media-type for Plan Registration (1) (CAMP11-221)

Addressing normative statement:

To support the registration of Plans via a reference to either a PDP or a Plan file, Providers SHALL accept the "application/json" media type. **[PR-69]**

TA_Id: CAMP11-221

NormativeSource: PR-69 (section: Registering a Plan by Reference)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) has no HTTP "Accept" header or has one set to "application/json" and contains a "pdp_uri" or a "plan_uri" attribute.

Predicate: The HTTP code of (m2) is NOT code 415, "Unsupported Media Type".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.7.4 Duplicate Keys in JSON, Provider side (CAMP11-204)

Addressing normative statement:

Consumers and Providers SHALL NOT transmit JSON objects that contain duplicate keys. **[PR-02]**

TA_Id: CAMP11-204

NormativeSource: PR-02 (section: Required Formats / Duplicate Keys in JSON Objects)

Target: A successful response message (m2) with JSON content (Content-Type = "application/json") to a GET resource request (m1).

Predicate: there is no duplicate JSON keys at same level in the JSON content of the target message (m2).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.7.5 Duplicate Keys in JSON, Provider side, Error generation (CAMP11-205)

Addressing normative statement:

If a Consumer sends a Provider a request containing duplicate keys in a JSON object, the Provider SHOULD reject the request by sending back a '400 Bad Request' status code. **[PR-03]**

TA_Id: CAMP11-205

NormativeSource: PR-03 (section: Required Formats / Duplicate Keys in JSON Objects)

Target: A response message (m2) to a request message (m1) with JSON content that has duplicate keys at same level.

Predicate: The HTTP code in the target (m2) is '400 Bad Request'.

PrescriptionLevel: preferred

Tag: conformance=Provider

3.1.8 Resource Queries and HTTP

3.1.8.1 `select_attr` Positive Case Provider side (CAMP11-206)

Addressing normative statement:

A Provider SHALL return only the attributes of the queried resource that match the attribute names passed as arguments of 'select_attr'. **[PR-47]**

TA_Id: CAMP11-206

NormativeSource: PR-47 ("*If a GET message is sent with a select_attr request parameter in the URI, then only the attributes of the queried resource that match these parameters must be returned.*") (derived from Section: Request parameters)

Target: A successful response message (HTTP code 200) (m4) to a GET *resource* query message (m3) that uses "`select_attr`" query parameter.

Prerequisite: A previous message (m1) of the form GET *resource* URI (with no `select_attr` query parameter) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).

Predicate: the body of the target message (m4) contains only the attributes of the resource within those returned in (m2), that match those attributes in the "`select_attr`" query parameter.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.8.2 `select_attr` Negative Case, GET (CAMP11-208)

Addressing normative statement:

If an attribute is not part of the resource, an HTTP 4XX status code SHALL be returned. **[PR-09]**

TA_Id: CAMP11-208

NormativeSource: PR-09 (section: Request Parameters)

Target: A response message (m4) to a GET *resource* query message (m3) that uses "select_attr " query parameter.

Prerequisite:

- A previous message (m1) of the form GET *resource* URI (with no `select_attr` query parameter) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).
- In the request message (m3), there are attribute names (in `select_attr` query parameter) that do not appear in the response (m2).

Predicate: the HTTP code in the target message (m4) is of the form 4XX.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.8.3 `select_attr` Negative Case, PUT (CAMP11-209)

Addressing normative statement:

Upon receiving such a request the Provider SHALL respond with a 400 status code. **[PR-13]**

TA_Id: CAMP11-209

NormativeSource: PR-13 (section: Request Parameters)

Target: A response message (m4) to a PUT *resource* query message (m3) that uses "`select_attr` " query parameter.

Prerequisite:

- A previous message (m1) of the form GET *resource* URI (with no `select_attr` query parameter) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).
- In the PUT *resource* query message (m3), there are attribute names (in "`select_attr` " query parameter) that do not appear in the response (m2).

Predicate: the HTTP code in the target message (m4) is 400.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.9 Resource Updates

3.1.9.1 ETag HTTP Header in Responses (CAMP11-211)

Addressing normative statement:

All HTTP responses that return representation of a resource SHOULD use strong Etag response header field indicating the current value of the entity tag for the resource. **[PR-20]**

TA_Id: CAMP11-211

NormativeSource: PR-20 (section: Response Headers)

Target: A successful response message (HTTP code 200) (m2) to a GET *resource* message (m1).

Predicate: the target (m2) has an *ETag* HTTP header, with a valid entity tag value.

PrescriptionLevel: preferred

Tag: conformance=Provider

3.1.9.2 If-Match semantics (CAMP11-212)

Addressing normative statement:

If the If-Match header field value in the request does not match the one on the server-side, the Provider SHALL send back a '412 Precondition Failed' status code. **[PR-07]**

TA_Id: CAMP11-212

NormativeSource: PR-07 (section: Request Headers)

Target: A response message (m2) to a PUT request message (m1) that has an empty *If-Match* HTTP header value.

Prerequisite: Some previous responses to GET *resource* messages show the use of *ETag* by the provider.

Predicate: the target message (m2) has HTTP code '412 Precondition Failed'.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.9.3 PUT semantics, regular, Provider (1)(CAMP11-213)

Addressing normative statement:

On successfully processing an HTTP PUT request a Provider SHALL update all the Consumer-mutable attributes of the target resource, and only these, with the values of the matching attributes in the request. **[PR-48]**

On successfully processing the request the Provider SHALL create an *assembly resource* and return a 201 *Created* status code in the HTTP response. **[PR-53]**

TA_Id: CAMP11-213

NormativeSource: PR-48, RE-53 (section: Updating Resources)

Var: (resource) in {
    sensor,
    plan,
    assembly,
    component
}

Target: A successful response message (HTTP code 200)(m4) to a GET *resource* message (m3).

Prerequisite:

- A previous message (m1) of the form PUT *resource* URI (with no select_attr query parameter) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).

Predicate: the target message (m4) shows for consumer-mutable attributes (e.g. name. description) the same attribute values as their counterpart in the message (m1).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.9.4 PUT semantics, regular, Provider (2) (CAMP11-214)

Addressing normative statement:

If a resource attribute is present on a resource and if an HTTP PUT request omits that attribute, it SHOULD be treated by the Provider as a request to delete the attribute. **[PR-25]**

TA_Id: CAMP11-214

NormativeSource: PR-25, (section: Updating Resources)

```
Var: (resource) in {
    sensor,
    plan,
    assembly,
    component
}
```

Target: A successful response message (HTTP code 200)(m4) to a GET *resource* message (m3).

Prerequisite:

A previous message (m1) of the form PUT *resource* URI (with no `select_attr` query parameter) and with some consumer-mutable attribute missing (e.g. `description`) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).

Predicate: the target message (m4) does not show any (consumer-mutable) attribute name or value, for attribute names that were missing in (m1).

PrescriptionLevel: preferred

Tag: conformance=Provider

3.1.9.5 PUT semantics, immutable attributes, Provider (CAMP11-216)

Addressing normative statement:

On receiving such a request the Provider SHALL generate an HTTP response with 403 HTTP status code. **[PR-22]**

TA_Id: CAMP11-216

NormativeSource: PR-22 (section: Mutability of Resources Attributes)

```
Var: (resource) in {
    assembly,
    component,
    plan
}
```

Target: A response message (m4) to a PUT *resource* request (m3).

Prerequisite:

- A successful previous response message (m2) to a GET *resource* message (m1) (same resource URI as target) (and assuming no "delete *resource*" message after m1).
- The message (m3) is either missing some consumer-immutable attributes for this resource type (compared with those present in the message m2), or is providing some values different from those in (m2).

Predicate: The response (m4) has HTTP code 203

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.9.6 PATCH support, Provider (CAMP11-218)

Addressing normative statement:

Providers SHALL support the 'add', 'remove', and 'replace' operations. **[PR-27]**

Providers SHALL support the HTTP GET, PUT, and PATCH methods on all of the resources defined in this section. **[RE-53]**

TA_Id: CAMP11-218

NormativeSource: PR-27, RE-53 (section: Updating with JSON Patch)

```
Var: (resource) in {
    assembly,
    component,
    plan
}
```

Target: A response message (m4) to a PATCH *resource* request (m3) that contains no other operations than 'add', 'remove', and 'replace'.

Prerequisite:

- A previous GET *resource* message (m1) was sent on same resource URI as (m3) with a successful response message (m2). (and assuming no "delete *resource*" message after m1).
- The PATCH message (m3) only attempts to update consumer-mutable attributes.

Predicate: Target message (m4) has HTTP code 2xx .

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.9.7 PATCH semantics, Provider (CAMP11-219)

Addressing normative statement:

Providers SHALL support the 'add', 'remove', and 'replace' operations. **[PR-27]**

TA_Id: CAMP11-219

NormativeSource: PR-27, (RE-53) (section: Updating with JSON Patch)

```
Var: (resource) in {
    assembly,
    component,
    plan,
    service
}
```

Target: A successful response message (HTTP code 200)(m4) to a GET *resource* request message (m3).

Prerequisite:

A previous message (m1) of the form PATCH *resource* on same URI as (m3) was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).

Predicate: the target message (m4) shows for consumer-mutable attributes the same attribute values as the message (m1) for "replace" and "add" ops, and it does not show any (consumer-mutable) attribute that was in an "remove" op in (m1).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10 Resource Creation and Consistency of Representations

3.1.10.1 Creation of a `plan` by Reference (1) (CAMP11-301)

Addressing normative statements:

On successfully processing the request the Provider SHALL create a *plan resource* and return a 201 Created status code in the HTTP response. **[PR-57]**

TA_Id: CAMP11-301

NormativeSource: PR-57 (section: Registering a Plan by Reference)

Target: A response message (m2) to a POST `plans` message (m1).

Prerequisite: the target message (m2) has not an HTTP code 202 "accepted"

Predicate: the target message (m2) has HTTP code 201 "created"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10.2 Creation of a `plan` by Reference (2) (CAMP11-321)

Addressing normative statements:

On successfully processing the request the Provider SHALL create a *plan resource* and return a 201 Created status code in the HTTP response. **[PR-57]**

TA_Id: CAMP11-321

NormativeSource: PR-57 (section: Registering a Plan by Reference)

Target: A response message (m4) to a GET `plan` message (m3), with a `representation_skew`=NONE.

Prerequisite: A sequence of messages occurred as follows:

- A POST `plans` message (m1) with a `pdp_uri` has been sent, and a response message (m2) received with HTTP code 201 "Created" and with a location header with a URI.
- The GET message (m3) is sent to the location URI returned by (m2)

Predicate: the target message (m4) has HTTP code 200 AND its content satisfies the serialization schema (JSON) for `Plan`, AND has a `type` attribute of value = "plan".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10.3 Consistency of `assemblies` Links after Deployment of a PDP by Reference (CAMP11-302)

Addressing normative statement:

The Provider SHALL update the `assembly_links` attribute of the *assemblies resource* to include a reference to the newly created *assembly resource*. **[PR-52]**

TA_Id: CAMP11-302

NormativeSource: PR-52 (section: Deploying an Application by Reference)

Target: A response message (m4) to a GET `assemblies` message (m3), with a `representation_skew`=NONE.

Prerequisite: A sequence of messages previously occurred as follows:

- A POST `assemblies` message (m1) with either a `pdp_uri` or a `plan_uri` has been sent,

- A successful (HTTP code 201 "Created") response message (m2) received with a location header with a URI.
- The GET `assemblies` message (m3) is sent on same platform as (m1).

Predicate: the target message (m4) has HTTP code 200 AND its `assemblies.assembly_links` resource contains the location URI returned by (m2).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10.4 Consistency of `assemblies` Links after Deployment of a PDP or Plan by Value (CAMP11-303)

Addressing normative statement:

The Provider SHALL update the `assembly_links` attribute of the *assemblies resource* to include a reference to the newly created *assembly resource*. **[PR-55]**

TA_Id: CAMP11-303

NormativeSource: PR-55 (PR-53, PR-54), (section: Deploying an Application by Value)

Target: A response message (m4) to a GET `assemblies` message (m3), with a `representation_skew`=NONE.

Prerequisite: A sequence of messages occurred as follows:
- A POST `assemblies` message (m1) with either an embedded PDP or a Plan has been sent,
- A successful (HTTP code 201 "Created") response message (m2) received with a location header with a URI.
- The GET `assemblies` message (m3) is on same platform URI as (m1).

Predicate: the target message (m4) has HTTP code 200 AND its `assemblies.assembly_links` resource contains the location URI returned by (m2).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10.5 Effective Creation of an `assembly` (2), setting Links (CAMP11-306)

Addressing normative statement:

An *assembly resource* SHALL have at least one reference to a *component resource*. **[RE-39]**

Providers that support Plans SHALL include this attribute (plan_uri) in all *assembly resources*. **[RMR-04]**

TA_Id: CAMP11-306

NormativeSource: RE-39, RMR-04

Target: A response message (m4) to a GET `assembly` message (m3), with a `representation_skew`=NONE.

Prerequisite: A sequence of messages occurred as follows:
- A POST `plan` message (m1) has been sent (the `plan` URI exists in the `plans.plan_links` obtained from `platform.plan_uris`), with a successful (HTTP code 201 "Created") response message (m2) received with a location header with a URI.
- The GET `assembly` message (m3) is sent on the URI returned by (m2).

Predicate: the target message (m4) has HTTP code 200 AND its `assembly.components`[] has at least one link to a Component resource, AND its `assembly.plan_uri` link is set to the URI used in (m1).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.10.6 Direct Creation of a `component` (1) (CAMP11-310)

Addressing normative statement:

If this attribute is present in the resource, Providers SHALL support the POST method on that resource in addition to the methods defined in Section 5.5, "HTTP Method Support". **[RE-38]**

TA_Id: CAMP11-310

NormativeSource: RE-38 (section: `Service` Resource / `parameter_definitions_uri`)

Target: A response message (m2) to a POST `service` message (m1) that has the `parameter_definitions_uri` defined.

Predicate: the target message has HTTP code 201 "Created" and a location header with a URI.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.11 Use of Parameters

3.1.11.1 Consistent Use of Parameters in POST (CAMP11-312)

Addressing normative statement:

If a POST request body does not contain a value for a required parameter, a "400 Bad Request" response SHALL be returned. **[PR-18]**

TA_Id: CAMP11-312

NormativeSource: PR-18, (section: POST Body Parameters / Parameter Handling)

Var: (resource) in {
    assembly,
    plan,
    service
}

Target: A response message (m6) to a POST *resource* message (m5), where (m5) is not setting a required parameter as defined in (m4).

Prerequisite:

- A successful response message (m2) to a previous GET *resource* message (m1).

- A successful response message (m4) to a GET *resource.*`parameter_definitions_uri` message (m3). In the array (m4)`parameter_definitions.parameter_definitions_links` some link(s) refer to a `parameter_definition` resource with a required attribute set to "true".

Predicate: the target message (m6) has an HTTP error code 400 "Bad request".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.11.2 Acceptable Value of Parameters in POST (CAMP11-313)

Addressing normative statement:

If a POST request body does not contain an acceptable value for a parameter, a "400 Bad Request" response SHALL be returned. **[PR-19]**

TA_Id: CAMP11-313

NormativeSource: PR-19, (section: POST Body Parameters / Parameter Handling)

Var: (resource) in {
    `assemblies,`
    `plans,`
    `service`
}

Target: A response message (m6) to a POST *resource* message (m5), where (m5) is setting a parameter with a value incompatible with its `parameter_type` as defined in (m4).

Prerequisite:

* A successful response message (m2) to a previous GET *resource* message (m1).

* A successful response message (m4) to a GET *resource.*`parameter_definitions_uri` message (m3).

* Successful response messages to access enough `parameter_definition` resources referred In the links (m4)`parameter_definitions.parameter_definitions_links`, in order to show incompatibility of a parameter value used in (m5)

Predicate: the target message (m6) has an HTTP error code 400 "Bad request".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.12 Resource Deletion

3.1.12.1 HTTP-level Deletion of an `assembly` resource (CAMP11-317)

Addressing normative statement:

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP DELETE method on the *assembly resource*. **[RE-61]**

TA_Id: CAMP11-317

NormativeSource: RE-61 (section: Deleting an Application Instance and a Deployed Application)

Target: A response message to a DELETE `assembly` message.

Prerequisite: The `assembly` URI exists as a link in `assemblies.assembly_links` obtained from `platform.assemblies_uri` from prior successful (HTTP code 200 "OK") response message to a GET `platform` message.

Predicate: the target message has NOT an HTTP code 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.12.2 Effective Deletion of an `assembly` resource (CAMP11-318)

Addressing normative statement:

On reception of a DELETE request a Provider SHALL remove the reference to the *assembly resource* from the *assemblies resource's* `assembly_links` array. **[RE-74]**

TA_Id: CAMP11-318

NormativeSource: RE-74 (RE-61)(section: Deleting an Application Instance and a Deployed Application)

Target: A response message (m4) to a GET `assembly` URI message (m3).

Prerequisite: The following sequence of messages occurred:

- Prior to the target message:

- Prior to the target message: A successful (HTTP code 2xx) response (m2) to a DELETE `assembly` message (m1). The `assembly` URI is same as in (m3) and exists in a link in `assemblies.assembly_links` obtained from `platform.assemblies_uri` as shown in a previous GET `platform` message.

- After the target message: A successful (HTTP code 2xx) response (m6) to a GET `assemblies` message (m5).

Predicate: the target message (m4) has HTTP code 5xx, AND the URI does not exist anymore in a link in (m6) `assemblies.assembly_links`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.12.3 Effective Deletion of `component` resources of an `assembly` resource (CAMP11-340)

Addressing normative statement:

On reception of a DELETE request a Provider SHALL remove the *assembly resource* from the system along with any *component resources* referenced by that assembly resource. (i.e. the tree of resources that was created when the application was instantiated). **[RE-73]**

TA_Id: CAMP11-340

NormativeSource: RE-73, (RE-61)(section: `assembly` Resource)

Target: A response message (m6) to a GET `component` URI message (m5), where the `component` URI belongs to m2:`assembly.components[]` (see prerequisite).

Prerequisite: The following sequence of messages occurred prior to the target message:

- A successful (HTTP code 2xx) response (m2) to a GET `assembly` message (m1).
- A successful (HTTP code 2xx) response (m4) to a DELETE `assembly` message (m3).

Predicate: the target message (m6) has HTTP code 5xx.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.12.4 Effective Deletion of a `component` resource (CAMP11-320)

Addressing normative statement:

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP DELETE method on the *component resource*. **[RE-62]**

TA_Id: CAMP11-320

NormativeSource: RE-62 (section: Component Resource)

Target: A response message (m4) to a GET `component` URI message (m3).

Prerequisite: The following sequence of messages occurred:

- Prior to the target message: A successful (HTTP code 2xx) response (m2) message to a DELETE `component` message (m1). The URI is same as in (m3), and exists in a link in an `assembly.components[]` as shown in a previous GET `assembly` message (m0).

- After the target message occurred: a successful (HTTP code 200 "OK") response message (m6) to a GET `assembly` message (m5) on same URI as in (m0).

Predicate: the target message (m4) has HTTP code 5xx, AND the URI does not exist anymore as a link in (m6)`assembly.components[]`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.12.5 HTTP Deletion of a `component` resource (CAMP11-323)

Addressing normative statement:

In addition to the methods defined in Section 5.5, "HTTP Method Support", Providers SHALL support the HTTP DELETE method on the *component resource*. **[RE-62]**

TA_Id: CAMP11-323

NormativeSource: RE-62 (section: Component Resource)

Target : A response message to a DELETE `component` message.

Prerequisite: The `component` URI exists as a link in an `assembly.components[]` as shown in a previous GET `assembly` message.

Predicate: the target message has NOT an HTTP code 405 "Method Not Allowed"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.13 Representation Skew Semantics

3.1.13.1 representation_skew Semantics (1) (CAMP11-329)

Addressing normative statement:

The following table lists the methods that SHALL be supported for each representation_skew value. **[RE-12]**

representation_skew value	Methods Available
CREATING	GET, DELETE
NONE	All supported methods for that resource.
UNKNOWN	All supported methods for that resource.
DESTROYING	GET

TA_Id: CAMP11-329

NormativeSource: RE-12, (RE-53) (section: representation_skew)

```
Var: (resource) in {
    plan,
    component,
    assembly
}
```

Target: A response message (m4) to a PUT request (m3) to a *resource*.

Prerequisite: The following message sequence occurred:

- Prior to the target: A message (m1) of the form GET *resource* URI was sent with a successful response message (m2), showing `representation_skew`=CREATING.

- After the target: An message (m5) similar to (m1) with a successful response message (m6), still showing `representation_skew`=CREATING.

Predicate: the target message has an 4xx HTTP code.

 PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.13.2 representation_skew Semantics (2) (CAMP11-330)

Addressing normative statement:

(see CAMP11-329)

The following table lists the methods that SHALL be supported for each representation_skew value. **[RE-12]**

TA_Id: CAMP11-330

NormativeSource: RE-12, (RE-53) (section: representation_skew)

Var: (resource) in {
```
    plan,
    component,
    assembly
}
```
Target: A response message (m4) to a PATCH *resource* message (m3).

Prerequisite: The following message sequence occurred:

- Prior to the target: A message (m1) of the form GET *resource* URI was sent with a successful response message (m2), showing `representation_skew`=CREATING.

- After the target: An message (m5) similar to (m1) with a successful response message (m6), still showing `representation_skew`=CREATING.

Predicate: the target message has an 4xx HTTP code.

 PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.13.3 representation_skew Semantics (3) (CAMP11-331)

Addressing normative statement:

(see CAMP11-329)

The following table lists the methods that SHALL be supported for each representation_skew value. **[RE-12]**

TA_Id: CAMP11-331

NormativeSource: RE-12, (RE-53) (section: representation_skew)

Var: (resource) in {
```
    plan,
    assembly
}
```
Target: A response message (m4) to a PUT *resource* message (m3).

Prerequisite: The following message sequence occurred:

- Prior to the target: A message (m1) of the form GET *resource* URI was sent with a successful response message (m2), showing `representation_skew`=DESTROYING.

- After the target: An message (m5) similar to (m1) with a successful response message (m6), still showing `representation_skew`=DESTROYING.

Predicate: the target message has an 4xx HTTP code.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.13.4 representation_skew Semantics (4) (CAMP11-332)

Addressing normative statement:

(see CAMP11-329)

The following table lists the methods that SHALL be supported for each representation_skew value. **[RE-12]**

TA_Id: CAMP11-332

NormativeSource: RE-12, (RE-53) (section: representation_skew)

Var: `(resource) in {`
`    plan,`
`    assembly`
`}`

Target: A response message (m4) to a PATCH *resource* message (m3).

Prerequisite: The following message sequence occurred:

- Prior to the target: A message (m1) of the form GET *resource* URI was sent with a successful response message (m2), showing `representation_skew`=DESTROYING.

- After the target: An message (m5) similar to (m1) with a successful response message (m6), still showing `representation_skew`=DESTROYING.

Predicate: the target message has an 4xx HTTP code.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.13.5 representation_skew Semantics (5) (CAMP11-333)

Addressing normative statement:

(see CAMP11-329)

The following table lists the methods that SHALL be supported for each representation_skew value. **[RE-12]**

TA_Id: CAMP11-333

NormativeSource: RE-12, (RE-53) (section: representation_skew)

Var: `(resource) in {`
`    plan,`
`    assembly`
`}`

Target: A response message (m4) to a DELETE *resource* message (m3).

Prerequisite: The following message sequence occurred:

- Prior to the target: A previous message (m1) of the form GET *resource* URI was sent with a successful response message (m2), showing `representation_skew`=DESTROYING.

- After the target: An message (m5) similar to (m1) with a successful response message (m6), still showing `representation_skew`=DESTROYING.

Predicate: the target message has an 4xx HTTP code.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14 PDP and Plan Registration

3.1.14.1 PDP Registration by reference: Provider side (1) (CAMP11-412)

Addressing normative statement:

On successfully processing the request the Provider SHALL create a *plan resource* and return a 201 Created status code in the HTTP response. **[PR-57]**

TA_Id: CAMP11-412

NormativeSource: PR-57 (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) has a "`pdp_uri`" attribute.

Prerequisite: the HTTP code in target (m2) is NOT 202 "accepted".

Predicate: the HTTP code in target (m2) is 201 "Created".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.2 PDP Registration by reference: Provider side (2) (CAMP11-414)

Addressing normative statement:

On successfully processing the request the Provider SHALL create a *plan resource* and return a 201 Created status code in the HTTP response. **[PR-57]**

The Provider SHALL include the *Location* header in the HTTP response and the value of this header SHALL reference the newly created *plan resource*. **[PR-58]**

TA_Id: CAMP11-414

NormativeSource: PR-57, PR-58 (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) has a "`pdp_uri`" attribute.

Prerequisite: the HTTP code in target (m2) is 201 "Created".

Predicate: the target message contains a location header with the URI for the new `plan` resource AND this URI resolves (GET `plan`) to a `plan` resource with `type` attribute = "plan"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.3 PDP Registration by reference: Provider side (CAMP11-425)

Addressing normative statement:

The Provider SHALL update the `plan_links` attribute of the *plans resource* to include a reference to the newly created *plan resource*. **[PR-59]**

TA_Id: CAMP11-425

NormativeSource: PR-59

Target: A response message (m4) to a GET `plans` message (m3), with a `representation_skew`=NONE.

Prerequisite: A sequence of messages previously occurred as follows:

- A POST `plans` message (m1) with either a `pdp_uri` or a `plan_uri` has been sent,

- A successful (HTTP code 201 "Created") response message (m2) received with a location header with a URI.
- The GET `plans` message (m3) is sent on same `platform` as (m1).

Predicate: the target message (m4) has HTTP code 200 AND its `plans.plan_links` attribute contains the location URI of the new `plan` resource returned by (m2).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.4 PDP Registration by value: Provider side (1) (CAMP11-418)

Addressing normative statement:

On successfully processing the request the Provider SHALL create a *plan resource* and return a 201 Created status code in the HTTP response. **[PR-62]**

The Provider SHALL include the *Location* header in the HTTP response and the value of this header SHALL reference the newly created *plan resource*. **[PR-63]**

TA_Id: CAMP11-418

NormativeSource: PR-62, PR-63 (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type: `application/x-zip` or `application/x-tar` or `application/x-tgz`, or `application/x-yaml`.

Prerequisite: the HTTP code in target (m2) is 201 "Created".

Predicate: the target message contains a location header with the URI for the new `plan` resource AND this URI resolves (GET `plan`) to a `plan` resource with `type` attribute = "plan".

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.5 PDP Registration by value: Provider side (2) (CAMP11-413)

Addressing normative statement:

To support the registration of Plans using a PDP, Providers SHALL accept the media types associated with the various formats as follows:
- ZIP: "application/x-zip" **[PR-70]**
- TAR: "application/x-tar" **[PR-71]**
- GZIP compressed TAR: "application/x-tgz" **[PR-72]**

TA_Id: CAMP11-413

NormativeSource: PR-70, PR-71, PR-72, (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type: `application/x-zip` or `application/x-tar` or `application/x-tgz`.

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the new `plan` resource, OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK").

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.6 PDP Registration by value: Provider side (2) (CAMP11-419)

Addressing normative statement:

To support the registration of Plans using a Plan file, Providers SHALL accept the use of the "application/x-yaml" media type. **[PR-73]**

TA_Id: CAMP11-419

NormativeSource: PR-73 (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type: `application/x-yaml`.

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the new `plan` resource, OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK").

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.7 Plan Registration by value using MIME Provider side (CAMP11-433)

Addressing normative statement:

Providers that support the plans resource and plan resources SHALL support the registration of Plans via HTTP POST requests on the plans resource as described in this section. **[PR-75]**

TA_Id: CAMP11-433

NormativeSource: PR-75, (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type = `multipart/form-data`, and the `Content-Disposition` header has name set to "plan_file"

Predicate: (the HTTP code in target (m2) is 201 "Created" AND the target message m2 contains a location header with the URI for the new `plan` resource), OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK").

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.8 PDP Registration by value using MIME Provider side (CAMP11-437)

Addressing normative statement:

Providers that support the plans resource and plan resources SHALL support the registration of Plans via HTTP POST requests on the plans resource as described in this section. **[PR-75]**

TA_Id: CAMP11-437

NormativeSource: PR-75, (RMR-05)

Target: A response message (m2) to a POST `plans` message (m1) where (m1) does not contain a "pdp_uri" nor a "plan_uri" attribute, and contains a Content-Type = `multipart/form-data`, and the `Content-Disposition` header has name set to "pdp_file"

Predicate: (the HTTP code in target (m2) is 201 "Created" AND the target message m2 contains a location header with the URI for the new `plan` resource), OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK") .

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.9 PDP Registration: URI scheme for Artifact Content reference (1) (CAMP11-415)

Addressing normative statement:

Providers SHALL support the "https" URI scheme as defined in RFC 2818 [RFC2818]. **[PDP-27]**

NOTE: partial test.

TA_Id: CAMP11-415

NormativeSource: PDP-27 (section: ContentSpecification)

Target: A response message (m2) to a POST `platform` message (m1) that uses "https" scheme, and refers to a PDP or passes a PDP or a Plan by value, where some `ArtifactSpecifiation` element has a content element with an "href" attribute.

Prerequisite: the PDP and contained Plan are well-formed.

Predicate: the HTTP code in target message is 2xx.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.10 PDP Registration: pdp scheme for Artifact Content reference (Provider) (CAMP11-417)

Addressing normative statement:

Providers SHALL understand this delimiter and SHALL NOT resolve any content if the archive format is unsupported. **[PDP-29]**

TA_Id: CAMP11-417

NormativeSource: PDP-29 (section: ContentSpecification)

Target: A response to a POST `platform` message that refers to a PDP or passes a PDP or a Plan by value, where some `ArtifactSpecification` element has a content element with an "href" attribute, that uses the URI scheme "pdp".

Prerequisite: the PDP and contained Plan are well-formed AND the archive format used in the first part of the URI path (before "!") is known to be supported .

Predicate: the HTTP code in target is 2xx.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.11 Registering PDP by Value with .zip format (CAMP11-422)

Addressing normative statement:

A Provider SHALL support the following archive formats for a PDP:

- A PDP as a ZIP archive [ZIP] **[PDP-02]**

To support the registration of Plans using a PDP, Providers SHALL accept the media types associated with the various formats as follows:

- ZIP: "application/x-zip" **[PR-70]**

TA_Id: CAMP11-422

NormativeSource: PDP-02, PR-70, (RMR-02, PR-63)(section: Supported Archive Formats)

Target: A response message (m2) to a A POST `platform` message (m1) with no "`pdp_uri`" nor a "`plan_uri`" attribute, and with an embedded PDP of type .zip (Content-Type: `application/x-zip`).

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the deployed `plan`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.12 Registering a PDP by Value with .tar format (CAMP11-423)

Addressing normative statement:

A Provider SHALL support the following archive formats for a PDP:

- A PDP as a TAR archive [TAR] **[PDP-03]**

To support the registration of Plans using a PDP, Providers SHALL accept the media types associated with the various formats as follows:

- TAR: "application/x-tar" **[PR-71]**

TA_Id: CAMP11-423

NormativeSource: PDP-03, PR-71 (PR-63) (section: Supported Archive Formats)

Target: A response message (m2) to a A POST `platform` message (m1) with no "`pdp_uri`" nor a "`plan_uri`" attribute, and with an embedded PDP of type .tar (Content-Type: `application/x-tar`).

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the deployed `plan`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.14.13 Registering a PDP by Value with .tar.gz format (CAMP11-424)

Addressing normative statement:

A Provider SHALL support the following archive formats for a PDP:

- A PDP as a GZIP [RFC1952] compressed TAR archive **[PDP-04]**

To support the registration of Plans using a PDP, Providers SHALL accept the media types associated with the various formats as follows:

- GZIP compressed TAR: "application/x-tgz" **[PR-72]**

TA_Id: CAMP11-424

NormativeSource: PDP-04, PR-72 (PR-63) (section: Supported Archive Formats)

Target: A response message (m2) to a A POST `platform` message (m1) with no "`pdp_uri`" nor a "`plan_uri`" attribute, and with an embedded PDP of type .tar.gz (Content-Type: `application/x-tgz`).

Predicate: the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the deployed `plan`.

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15 Application Deployment

3.1.15.1 Application Deployment by reference: Provider side (1) (CAMP11-439)

Addressing normative statement:

On successfully processing the request the Provider SHALL create an *assembly resource* and return a 201 Created status code in the HTTP response. **[PR-50]**

TA_Id: CAMP11-439

NormativeSource: PR-50 (section: Deploying an Application by Reference)

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) has a either a "`pdp_uri`" or a "`plan_uri`" attribute.

Prerequisite: the target message (m2) has not an HTTP code 202 "accepted"

Predicate: the target message (m2) has HTTP code 201 "created"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15.2 Application Deployment by reference: Provider side (2) (CAMP11-440)

Addressing normative statement:

On successfully processing the request the Provider SHALL create an *assembly resource* and return a 201 Created status code in the HTTP response. **[PR-50]**

The Provider SHALL include the *Location* header in the HTTP response and the value of this header SHALL reference the newly created *assembly resource*. **[PR-51]**

TA_Id: CAMP11-440

NormativeSource: PR-50, PR-51

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) has a either a "`pdp_uri`" or a "`plan_uri`" attribute.

Predicate: the target message contains a location header with the URI for the new `assembly` resource, OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK".)

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15.3 Application Deployment by value: Provider side (1) (CAMP11-441)

Addressing normative statement:

On successfully processing the request the Provider SHALL create an *assembly resource* and return a 201 *Created* status code in the HTTP response. **[PR-53]**

TA_Id: CAMP11-441

Var: *archtype* in { `application/x-zip`, `application/x-tar`, `application/x-tgz`, `application/x-yaml` }

NormativeSource: PR-53, (PR-60), {PR-29, PR-30, PR-31, PR-32}

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) does not contain a "`pdp_uri`" nor a "`plan_uri`" attribute, and contains a Content-Type = *archtype*.

Prerequisite: HTTP code in (m2) is NOT 202 "accepted" i.e. enough time elapsed to allow for the creation to complete.

Predicate: the HTTP code in target (m2) is 201 "Created"

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15.4 Application Deployment by value: Provider side (2) (CAMP11-442)

Addressing normative statement:

On successfully processing the request the Provider SHALL create an *assembly resource* and return a 201 *Created* status code in the HTTP response. **[PR-53]**

The Provider SHALL include the *Location* header in the HTTP response and the value of this header SHALL reference the newly created *assembly resource*. **[PR-54]**

TA_Id: CAMP11-442

Var: *archtype* in { `application/x-zip, application/x-tar, application/x-tgz, application/x-yaml` }

NormativeSource: PR-53, PR-54, (PR-60), {PR-29, PR-30, PR-31, PR-32}

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) does not contain a "`pdp_uri`" nor a "`plan_uri`" attribute, and contains a Content-Type = *archtype*.

Predicate: (the HTTP code in target (m2) is 201 "Created" AND the target message contains a location header with the URI for the new `assembly` resource), OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK")

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15.5 Application Deployment by value using MIME: Provider side (CAMP11-443)

Addressing normative statement:

Providers SHALL support the deployment of applications via HTTP POST requests on the *assemblies resource* as described in this section. **[PR-74]**

TA_Id: CAMP11-443

NormativeSource: PR-74, { PR-29, PR-30, PR-31, PR-32 }

Target: A response message (m2) to a POST `assemblies` message (m1) where (m1) does not contain a "`pdp_uri`" nor a "`plan_uri`" attribute, and contains a Content-Type = `multipart/form-data`.

Prerequisite: The name parameter of (m1) `Content-Disposition` header is set to "`pdp_file`".

Predicate: (the HTTP code in target (m2) is 201 "Created" AND the target message m2 contains a location header with the URI for the new `assembly` resource), OR (code = 202 "accepted" AND a subsequent GET on the returned Location header URI shows either `representation_skew`=CREATING or HTTP code 200 "OK")

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.1.15.6 Creation of `plan` when Deploying an Application (1) (CAMP11-444)

Addressing normative statement:

Regardless of whether a Consumer attempts to create an *assembly resource* by POSTing to the *assemblies resource* or creates a *plan resource* by POSTing to the *plans resource*, a Provider that supports *plans* and *plan resources* SHALL create a *plan resource* for every deployed application. **[RMR-11]**

TA_Id: CAMP11-444

NormativeSource: RMR-11

Target: A response message (m4) to a GET `assembly` message (m3), with HTTP code 200 and with a `representation_skew`=NONE.

Prerequisite: A set of messages previously occurred as follows:

- A successful POST `assemblies` message (m1) has been sent (response HTTP code 201 "Created"), that returns a Location header URI (m2) used in (m3) (GET `assembly`)

- A successful response message (m2) received from a GET `platform` message (m1) shows a `plan_uris` attribute set to a valid `plans` Resource.

Predicate: the target message (m4) has its `assembly.plan_uri` link set to a `plan` Resource (a GET on that URI returns HTTP code 200 "OK") AND the `plan` URI also exists in the `plans.plan_links` obtained from `platform.plan_uris`, (m2).

PrescriptionLevel: mandatory

Tag: conformance=Provider

3.2 Consumer Test Assertions

These Test Assertions concern the Consumer as conformance target.

3.2.1 Basic Schema Compliance for Platform Resources, Consumer Side

3.2.1.1 Resource serialization with PUT (CAMP11-105)

TA_Id: CAMP11-105

NormativeSource: RE-71 (section: Resources)

```
Var: (resource) in {
    sensor,
    plan,
    assembly,
    component
}
```

Target: A message (m1) of the form PUT *resource* URI (with no `select_attr` query parameter)

Predicate: the target message (m1) content satisfies the serialization schema (JSON) for *resource*, AND has a `type` attribute of value = <*resource* name>*.*

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.2 Consistency of Serialization with Metadata

3.2.2.1 Non-Mutability Enforcement (Consumer) (CAMP11-116)

Addressing normative statement:

TA_Id: CAMP11-116

NormativeSource: RE-09, (RE-53) (section: Consumer-mutable)

```
Var: (resource) in {
    assembly,
    component,
    platform
}
```

Target: A PUT request (m3) to a *resource*.

Prerequisite: The following message sequence occurred:

A previous message (m1) of the form GET *resource* URI was sent with a successful response message (m2). (and assuming no "delete *resource*" message after m1).

Predicate: The PUT request (m3) is providing same values as shown in (m2) for `consumer-mutable=false` attributes.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.3 Various Model Referential and Semantic Constraints

3.2.3.1 UTC Timestamps, Consumer side (CAMP11-135)

Addressing normative statement:
Consumers and Providers SHALL express Timestamps in UTC (Coordinated Universal Time), with the special UTC designator ("Z"). **[RE-65]**

TA_Id: CAMP11-135

NormativeSource: RE-65 (section: Timestamp)

Target: A POST or PUT message that has a resource serialization showing the "`timestamp`" attribute.

Predicate: the Timestamp value conforms to the UTC (Universal Time Coordinated) with a "zero" timezone offset, i.e. with format terminated with the designator "Z", of the form: yyyy '-' mm '-' dd 'T' hh ':' mm ':' ss 'Z' (Example: `1994-11-05T13:15:30Z`)

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.4 Support for JSON

3.2.4.1 Duplicate Keys in JSON, Consumer side (CAMP11-203)

Addressing normative statement:
Consumers and Providers SHALL NOT transmit JSON objects that contain duplicate keys. **[PR-02]**

TA_Id: CAMP11-203

NormativeSource: PR-02 (sections: Required Formats / Duplicate Keys in JSON Objects)

Target: A POST or a PUT request with JSON content to any platform resource.

Predicate: there is no duplicate JSON keys at same level in the JSON content of the target message.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

NOTE: the corresponding error handling TA (Provider side) is addressing PR-03.

3.2.5 Resource Queries and HTTP

3.2.5.1 `select_attr` Positive Case Consumer side (CAMP11-207)

Addressing normative statement:

When one or more request parameters are specified for a PUT request, a Consumer SHALL NOT include attributes in the request entity body that are not specified in the request parameter. **[PR-12]**

Note: Requirement about request parameters (6.5): "*In order to address a subset of attributes in a resource, the client can use request parameters in PUT … When one or more request parameters are specified for a PUT request, a client shall not include attributes in the request entity body that are not specified in the request parameter.*"

TA_Id: CAMP11-207

NormativeSource: PR-12 ("*If a PUT message is sent with a select_attr request parameter in the URI, then only the attributes that appear in the select_attr request parameter must be used in the body of the update message.*") (derived from section: Request parameters)

Target: A PUT request message to a *resource*, that uses "`select_attr`" query parameter.

Predicate: every attribute present in the body of the target message also appears in the "`select_attr`" query parameter.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.6 Resource Updates

3.2.6.1 If-Match HTTP Header in PUT Requests (CAMP11-210)

Addressing normative statement:

All PUT requests that update a resource SHOULD contain the If-Match header field with a single entity tag value. **[PR-06]**

TA_Id: CAMP11-210

NormativeSource: PR-06 (section: Request Headers)

Target: A PUT request message to any resource existing in the platform.

Predicate: the target has an *If-Match* HTTP header.

PrescriptionLevel: preferred

Tag: conformance=Consumer

3.2.6.2 PUT semantics, immutable attributes, Consumer (CAMP11-215)

Addressing normative statement:

Consumers SHALL NOT send a request that changes the value of a resource attribute that is declared with a constraint of 'Mutable=false' or 'Consumer-mutable=false'. **[PR-21]**

TA_Id: CAMP11-215

NormativeSource: PR-21 (section: Mutability of Resources Attributes)

```
Var: (resource) in {
    plan,
    assembly,
```

```
        component
}
```

Target: A PUT *resource* request message.

Prerequisite:

A successful previous response message (m2) to a GET *resource* message (m1) (same resource URI as target) (and assuming no "delete *resource*" message after m1).

Predicate: the target message has all the consumer-immutable attributes for this resource type, as already present in the message (m2), and with same values as in (m2).

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.6.3 PATCH syntax, Consumer (CAMP11-217)

Addressing normative statement:

Providers SHALL support the HTTP PATCH method in conjunction with the "application/json-patch+json" media type with the following, additional provisions with respect to the operations defined in Section 4 of the JSON Patch specification: **[PR-26]**

TA_Id: CAMP11-217

NormativeSource: PR-26 (section: Updating with JSON Patch)

```
Var: (resource) in {
    assembly,
    component,
    plan,
    service
}
```

Target: A PATCH *resource* request.

Predicate: the media-type (Content-Type header) application/json-patch

 is used in HTTP headers, AND the JSON content complies with JSON patch syntax (at least: an array of triples with keys "op", "path" and "value").

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7 PDP and Plan Contents

3.2.7.1 PDP Content by reference, Consumer side (1) (CAMP11-401)

Addressing normative statements:

The Plan file SHALL be located at the root of the PDP archive. **[PLAN-01]**

The Plan file SHALL be named "camp.yaml". **[PLAN-02]**

TA_Id: CAMP11-401

NormativeSource: PLAN-01, PLAN-02 (section: Plan Schema)

Target: A POST message with a "`pdp_uri`" attribute to a `plans` resource (obtained from `platform.plan_uris`), or an `assemblies` resource (obtained from `platform.assemblies_uri`).

Predicate: the PDP identified by `pdp_uri` is an archive that contains at its root a file named "`camp.yaml`", with a YAML structure recognizable as a Plan (same top-level attributes, and "tags").

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.2 PDP Content, by reference, Consumer side (2) (CAMP11-402)

Addressing normative statement:

A Platform Deployment Package (PDP) SHALL contain a single Plan file. **[PDP-11]**

TA_Id: CAMP11-402

NormativeSource: PDP-11 (section: Plan Schema)

Target: A POST message with a "`pdp_uri`" attribute to a `plans` resource (obtained from `platform.plan_uris`, or to an `assemblies` resource (obtained from `platform.assemblies_uri`).

Predicate: the PDP identified by `pdp_uri` is an archive that contains at its root a file named "`camp.yaml`", and does not contain any other `yaml` file of this name or of a structure recognizable as a `plan` (same top-level attributes, and "tags").

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.3 PDP Content, by value, Consumer side (1) (CAMP11-403)

Addressing normative statements:

The Plan file SHALL be located at the root of the PDP archive. **[PLAN-01]**

The Plan file SHALL be named "camp.yaml". **[PLAN-02]**

TA_Id: CAMP11-403

NormativeSource: PLAN-01, PLAN-02, (section: Plan Schema)

Target: A POST message without a "`pdp_uri`" or "`plan_uri`" attribute, to a `plans` resource (obtained from `platform.plan_uris`, or to an `assemblies` resource (obtained from `platform.assemblies_uri`).

Predicate: the PDP in the embedded archive contains at its root a file named "`camp.yaml`", with a structure recognizable as a `plan` (same top-level attributes, and "tags").

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.4 PDP Content, by value, Consumer side (2) (CAMP11-404)

Addressing normative statement:

A Platform Deployment Package (PDP) SHALL contain a single Plan file. **[PDP-11]**

TA_Id: CAMP11-404

NormativeSource: PDP-11 (section: Plan Schema)

Target: A POST message without a "`pdp_uri`" or "`plan_uri`" attribute, to a `plans` resource (obtained from `platform.plan_uris`, or to an `assemblies` resource (obtained from `platform.assemblies_uri`)..

Predicate: the PDP in the embedded archive contains at its root a file named "`camp.yaml`", and does not contain any other `yaml` file of this name or of a structure recognizable as a `plan` (same top-level attributes, and "tags").

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.5 Plan Content, Consumer side (1) (CAMP11-405)

Addressing normative statement:

A Plan file SHALL contain a single instance of a Plan. **[PLAN-03]**

TA_Id: CAMP11-405

NormativeSource: PLAN-03 (PDP-11) (section: Plan)

Target: A POST message to a `plans` URL, containing or referring to a Plan file.

Predicate: the Plan file identified in this POST contains a single `yaml plan` node (satisfies the `yaml` schema for this node).

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.6 Plan Content, Consumer side (2) (CAMP11-406)

Addressing normative statement:

For Plans that conform to this document, the value of this node (camp_version) SHALL be as defined in Section 1.8 "Specification Version". **[PLAN-05]**

TA_Id: CAMP11-406

NormativeSource: PLAN-05 (PLAN-03) (section: campVersion)

Target: A POST message to a `plans` URL, containing or referring to a Plan file.

Predicate: the Plan file identified in this POST has a `yaml` node that has a `camp_version` tag with value "CAMP 1.1".

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.7 Plan Content, Consumer side (3) (CAMP11-430)

Addressing normative statement:

Plans SHALL use id values that are unique within the scope of the Plan. **[PLAN-06]**

TA_Id: CAMP11-430

NormativeSource: PLAN-06 (section: serviceSpecification)

Target: A POST message to a `plans` URL, containing or referring to a Plan file.

Predicate: the `services` array (*ServiceSpecification*[]) contained in the Plan file identified in this POST does NOT have two *ServiceSpecification* items with same id node value.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.8 Plan Content, Consumer side (4) (CAMP11-431)

Addressing normative statement:

If a Consumer includes this node in a Plan, the value of this node SHALL reference a Consumer-visible resource within the target Platform. **[RMR-01]**

TA_Id: CAMP11-431

NormativeSource: RMR-01 (section: ServiceSpecification)

Target: A POST message to a `Plans` URL, containing or referring to a Plan file.

Predicate: Any `ServiceSpecification` items that has an `href` node in the `services` array (`ServiceSpecification[]`) contained in the Plan file identified in this POST, has the `href` value resolve to an existing resource on the target `platform`.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.9 Plan Artifact Type Content Specification, Consumer side (CAMP11-407)

Addressing normative statement:

The Plan file SHALL conform to the description provided in this section. **[PLAN-09]**

TA_Id: CAMP11-407

NormativeSource: PLAN-09 (section: ContentSpecification)

Target: A POST message to a `plans` URL, containing or referring to a Plan file.

Predicate: in the Plan file identified in this POST , every `ArtifactSpecification` item (`artifacts` array) has a `content` node that has either an `href` node, or a `data` node but not both.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.10 PDP Manifest, Consumer side (CAMP11-408)

Addressing normative statement:

The format of the manifest file and the certificate file SHALL be as defined by the OVF specification [OVF]. **[PDP-10]**

TA_Id: CAMP11-408

NormativeSource: PDP-10 (PDP-06) (section: Validating Integrity)

Target: A POST message to a `plans` URL, with either (a) a "`pdp_uri`", or (b) a Content-Type: `application/x-zip` or `application/x-tar` or `application/x-tgz`.

Prerequisite: the PDP referenced in the target message contains a manifest ("`camp.mf`") at its root.

Predicate: the manifest ("`camp.mf`") at the root of the PDP conforms to the structure defined in OVF [**DSP0243**] for OVF manifest files.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.7.11 PDP Certificate, Consumer side (CAMP11-409)

Addressing normative statement:

The format of the manifest file and the certificate file SHALL be as defined by the OVF specification [OVF]. **[PDP-10]**

TA_Id: CAMP11-409

NormativeSource: PDP-10 (section: Validating Integrity)

Target: A POST message to a `plans` URL, with either (a) a "`pdp_uri`", or (b) a Content-Type: `application/x-zip` or `application/x-tar` or `application/x-tgz`.

Prerequisite: the PDP referenced in the target message contains a certificate (named "`camp.cert`") at its root.

Predicate: the certificate (named "`camp.cert`") at the root of the PDP conforms to the structure defined in OVF [**DSP0243**] for OVF certificate files.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.2.8 PDP and Plan Registration

3.2.8.1 PDP Registration: pdp scheme for Artifact Content reference (Consumer) (CAMP11-416)

Addressing normative statement:

Consumers SHALL follow the syntax and semantics described here when using URIs with a "pdp" scheme. **[PLAN-07]**

TA_Id: CAMP11-416

NormativeSource: PLAN-07 (section: ContentSpecification)

Target: A POST `platform` message that refers to a PDP or passes a PDP or a Plan by value, where some `ArtifactSpecifiation` element has a content element with an "href" attribute, that uses the URI scheme "pdp" and contains the "!" character.

Predicate: the URI either shows "!" just after the scheme, or just after an archive name AND the right part after "!" is a path for a file that is resolvable from the root of the archive.

PrescriptionLevel: mandatory

Tag: conformance=Consumer

3.3 PDP and Plan Test Assertions

3.3.1 PDP Test Assertions

3.3.1.1 PDP and YAML Plan (CAMP11-401b)

Addressing normative statements:

The Plan file SHALL be located at the root of the PDP archive. **[PLAN-01]**

The Plan file SHALL be named "camp.yaml". **[PLAN-02]**

TA_Id: CAMP11-401b

NormativeSource: PLAN-01, PLAN-02 (section: Plan Schema)

Target: A PDP artifact.

Prerequisite: the PDP is in archive format (ZIP, TAR or GZIP).

Predicate: the PDP archive contains at its root a file named "`camp.yaml`", with the YAML structure of a Plan.

PrescriptionLevel: mandatory

Tag: conformance=PDP

3.3.1.2 PDP and Plan file (CAMP11-402b)

Addressing normative statement:

A Platform Deployment Package (PDP) SHALL contain a single Plan file. **[PDP-11]**

TA_Id: CAMP11-402b

NormativeSource: PDP-11 (section: Plan Schema)

Target: A PDP artifact.

Predicate: the PDP contains (at its root if an archive) a file named "`camp.yaml`", and does not contain any other `yaml` file of this name or of a structure recognizable as a `plan`.

PrescriptionLevel: mandatory

Tag: conformance=PDP

3.3.1.3 PDP Manifest (CAMP11-408b)

Addressing normative statement:

The format of the manifest file and the certificate file SHALL be as defined by the OVF specification [OVF]. **[PDP-10]**

TA_Id: CAMP11-408b

NormativeSource: PDP-10 (PDP-06) (section: Validating Integrity)

Target: A PDP artifact.

Prerequisite: the PDP contains a manifest ("`camp.mf`") at its root.

Predicate: the manifest ("`camp.mf`") at the root of the PDP conforms to the structure defined in OVF [**DSP0243**] for OVF manifest files.

PrescriptionLevel: mandatory

Tag: conformance=PDP

3.3.1.4 PDP Certificate (CAMP11-409b)

Addressing normative statement:

The format of the manifest file and the certificate file SHALL be as defined by the OVF specification [OVF]. **[PDP-10]**

TA_Id: CAMP11-409b

NormativeSource: PDP-10 (section: Validating Integrity)

Target: A PDP artifact.

Prerequisite: the PDP contains a certificate (named "`camp.cert`") at its root.

Predicate: the certificate (named "`camp.cert`") at the root of the PDP conforms to the structure defined in OVF [**DSP0243**] for OVF certificate files.

PrescriptionLevel: mandatory

Tag: conformance=PDP

3.3.2 Plan Test Assertions

3.3.2.1 Plan File Content (1) (CAMP11-405b)

Addressing normative statement:

A Plan file SHALL contain a single instance of a Plan. **[PLAN-03]**

TA_Id: CAMP11-405b

NormativeSource: PLAN-03 (PDP-11) (section: Plan)

Target: A Plan file.

Predicate: the Plan file contains a single `yaml plan` node and satisfies the `yaml` schema for this node.

PrescriptionLevel: mandatory

Tag: conformance=Plan

3.3.2.2 Plan CAMP version (CAMP11-406b)

Addressing normative statement:

For Plans that conform to this document, the value of this node (camp_version) SHALL be as defined in Section 1.8 "Specification Version". **[PLAN-05]**

TA_Id: CAMP11-406b

NormativeSource: PLAN-05 (PLAN-03) (section: campVersion)

Target: A Plan file.

Predicate: the Plan file has a `yaml` node that has a `camp_version` tag with value "CAMP 1.1".

PrescriptionLevel: mandatory

Tag: conformance=Plan

3.3.2.3 Plan and Unique Service IDs (CAMP11-430b)

Addressing normative statement:

Plans SHALL use id values that are unique within the scope of the Plan. **[PLAN-06]**

TA_Id: CAMP11-430b

NormativeSource: PLAN-06 (section: ServiceSpecification)

Target: A Plan file.

Predicate: the `services` array (*ServiceSpecification*[]) contained in the Plan file does NOT have two *ServiceSpecification* items with same id node value.

PrescriptionLevel: mandatory

Tag: conformance=Plan

3.3.2.4 Plan References (CAMP11-431b)

Addressing normative statement:

If a Consumer includes this node in a Plan, the value of this node SHALL reference a Consumer-visible resource within the target Platform. **[RMR-01]**

TA_Id: CAMP11-431b

NormativeSource: RMR-01 (section: ServiceSpecification)

Target: A Plan file.

Prerequisite: the Plan is originating on Consumer side.

Predicate: Any `ServiceSpecification` items that has an `href` node in the `services` array (`ServiceSpecification[]`) contained in the Plan file identified in this POST, has the `href` value resolve to an existing resource on the target `platform`.

PrescriptionLevel: mandatory

Tag: conformance=Plan

3.3.2.5 Plan Artifact integrity (CAMP11-407b)

Addressing normative statement:

The Plan file SHALL conform to the description provided in this section. **[PLAN-09]**

TA_Id: CAMP11-407b

NormativeSource: PLAN-09 (section: ContentSpecification)

Target: A Plan file.

Predicate: in the Plan file, every `ArtifactSpecification` item (`artifacts` array) has a `content` node that has either an `href` node, or a `data` node but not both.

PrescriptionLevel: mandatory

Tag: conformance= Plan

4 Conformance

4.1 Implementations

Implementations of this specification are **test suites** that are derived from (or exercise) all or parts of the test assertions defined in this document. Such Implementations are called here CAMP test suites.

Note: Implementations of this CAMP Test Assertions specification are NOT to be confused with implementations of the CAMP specification, which are programs or parts of a system implementing the CAMP Provider functions, or the CAMP Consumer functions, or yet data artifacts such as a CAMP PDP or a CAMP Plan. However, these four types of CAMP implementations (CAMP Provider, CAMP Consumer, PDP, Plan) are useful to partition the related Test assertions, and to define related conformance profiles for test suites.

A test suite is typically composed of test cases. According to the definition in [TAG] adopted here, a test case is:

> "*A set of a test tools, software or files (data, programs, scripts, or instructions for manual operations) that verifies the adherence of a test assertion target to one or more normative statements in the specification.*"

A test suite is said to implement a test assertion if it contains one or more test cases that exercise this test assertion, with the ability to assess whether an actual test target (i.e. an instance of the "target" of the test assertion) fulfills the associated normative CAMP requirement (as identified by the "normativeSource" of the test assertion). This means that the test case(s) related to a test assertion, must be able to either "pass" or "fail" most test targets, under expected usage conditions.

Identification of test restrictions:

- A conforming CAMP test suite that is unable to assess a subset of test targets for a particular test assertion, MUST clearly identify or characterize this subset, regardless whether this limitation is intentional or not.
- A conforming CAMP test suite that is unable to assess test targets under specific conditions or within a specific context for a particular test assertion, MUST clearly identify these conditions and/or context, regardless whether this limitation is intentional or not.

Four **conformance profiles** are defined for CAMP test suites:

1. CAMP test suite for CAMP Provider (or "CAMP Provider test suite")
2. CAMP test suite for CAMP Consumer (or "CAMP Consumer test suite")
3. CAMP test suite for PDP (or "CAMP PDP test suite")
4. CAMP test suite for Plan (or "CAMP Plan test suite")

A CAMP test suite may claim conformance to more than one of the above conformance profiles.

4.2 CAMP Provider Test Suite Conformance Profile

In order to claim conformance to the CAMP Provider test suite conformance profile, a CAMP test suite MUST do both of the following:

- Satisfy the above "Identification of test restrictions" requirements,
- Implement all test assertions (in the sense defined in Section 2) that are tagged with a "conformance" tag of value "Provider".

Such a test suite is qualified for verifying conformance of a CAMP implementation to the CAMP Provider profile.

4.3 CAMP Consumer Test Suite Conformance Profile

In order to claim conformance to the CAMP Consumer test suite conformance profile, a CAMP test suite MUST do both of the following:

- Satisfy the above "Identification of test restrictions" requirements,
- Implement all test assertions (in the sense defined in Section 2) that are tagged with a "conformance" tag of value "Consumer".

Such a test suite is qualified for verifying conformance of a CAMP implementation to the CAMP Consumer profile.

4.4 CAMP PDP Test Suite Conformance Profile

In order to claim conformance to the CAMP PDP test suite conformance profile, a CAMP test suite MUST do both of the following:

- Satisfy the above "Identification of test restrictions" requirements,
- Implement all test assertions (in the sense defined in Section 2) that are tagged with a "conformance" tag of value "PDP".

Such a test suite is qualified for verifying conformance of a CAMP implementation to the CAMP PDP profile.

4.5 CAMP Plan Test Suite Conformance Profile

In order to claim conformance to the CAMP Plan test suite conformance profile, a CAMP test suite MUST do both of the following:

- Satisfy the above "Identification of test restrictions" requirements,
- Implement all test assertions (in the sense defined in Section 2) that are tagged with a "conformance" tag of value "Plan".

Such a test suite is qualified for verifying conformance of a CAMP implementation to the CAMP Plan profile.

Appendix A. Acknowledgments

This section is informative. The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Roshan Agrawal	Rackspace Hosting, Inc.
Michael Behrens	US Department of Defense (DoD)
Bhaskar Reddy Byreddy	Software AG, Inc.
Mark Carlson	Oracle
Martin Chapman	Oracle
Francesco D'Andria	Cloud4SOA
Jacques Durand	Fujitsu Limited
Panagiotis Gouvas	Cloud4SOA
Keith Grange	JumpSoft
Alex Heneveld	Cloudsoft Corporation Limited
Gershon Janssen	Individual Member
David Jilk	Standing Cloud, Inc.
Duncan Johnston-Watt	Cloudsoft Corporation Limited
Anish Karmarkar	Oracle
Tobias Kunze	Red Hat
Eugene Luster	US Department of Defense (DoD)
Ashok Malhotra	Oracle
Alex McDonald	NetApp
Rich Miller	Cloudsoft Corporation Limited
Jeff Mischkinsky	Oracle
Adrian Otto	Rackspace Hosting, Inc.
Derek Palma	Vnomic
Gilbert Pilz	Oracle
Krishna Raman	Red Hat
Tom Rutt	Fujitsu Limited
Zhexuan Song	Huawei Technologies Co., Ltd.
Charles Tupitza	JumpSoft
Jeffrey West	Oracle
Prasad Yendluri	Software AG, Inc.

Appendix B. Revision History

Revision	Date	Editor	Changes Made
Wd18	Feb 4, 2014	Jacques Durand	First complete set of test assertions matching CAMP WD37
Wd19	Feb 7, 2014	Jacques Durand	Several edits from Wd18 (see diffs) - OASIS references set to the right citation format for OASIS refs. - Missing hyperlinks for cross-references (bookmarks) - Various minor edits in Section 2. - Improved section 2.3 (avoid redundancies, add more complete explanations.) - Improved on several PUT test assertions (more appropriate set of resources, for their variables) - Various minor edits and typos in TAs fixed. - Added missing TA (CAMP11-105) - Improved on the Conformance clause.
WD20	Feb 10, 2014	Gilbert Pilz	Editorial cleanups and formatting.
WD21	Feb 11, 2014	Jacques Durand	Editorial fixes.