PSI Tasks and Operations Dictionary

PSI-TOD

Version:	1.2.2
Date:	2025-02-03
Reference:	PSI-TOD
Total Pages:	191

Table of Contents

List of Figures

List of Tables

1 Document Meta Information

1.1 Document Signature Table

	Name	Function	Company
Author	Dafinka Srezoska	PSI Project Team	CGI
Author	Christian Grubert	PSI Project Team	CGI
Author	Norbert Czeranka	PSI Project Team	CGI
Author	Hendrik Oppenberg	Technical Officer	CGI
Approval	Rui Goncalves	Project Manager	SES
Approval	Wolfgang Robben	Project Manager	CGI
Checked	Pepijn Witte	Quality Assurance Manager	CGI

Table 1.1: Signature Table.

1.1.1 Document Change Record

1.1.1.1 Changes

Date	Version	author	message
2022-08-24	MS1	Christian Grubert	Initial version. Process definition. Building blocks for the party management
2022-09-30	MS2	Dafinka Srezoska	First API drafts and implementations: Resource, service and product catalogue management, Product offering management. Finalization of the party management. Customer inquiry initial version
2022-12-31	MS3	Dafinka Srezoska	Added inquiry, template, event, attachment and billing tasks. Various additional updates
2023-04-19	MS4	Norbert Czeranka	Added interaction diagrams, user mission tasks and geometries for beams. Clarified on ressource templates

Date	Version	author	message
2023-07-27	MS5	Christian Grubert	Added trouble ticket management, feature examples, priority, quality management figures and rephrased inquiry based on ESA comments
2023-10-06	MS6	Christian Grubert	Added access probability, defragmentation and updates required to encompass the JSON Schema approach
2024-01-25	MS7	Christian Grubert	Added product templates, improved product characteristics, advanced billing.
2024-09-11	MS8 [1.2.0]	Thomas Schulz	Public release adjustments.
2024-12-09	MS9 [1.2.1]	Christian Grubert	Definition of mission API.
2025-02-03	MS10 [1.2.2]	Christian Grubert	Relative time model in mission API, mission asset management.

Table 1.2: DCR Table.

1.1.1.2 Source Control

Changes to this document are tracked electronically. No signature is required by the authors. The following information can prove the integrity of the document and reveal any change.

Repo	Date	Author	Branch	Hash

Table 1.3: GIT Changelog Table.

Figure 1.1: DCR QR-Code.

1.2 Documents

1.2.1 Reference Documents

Acronym	Reference	Title	Version
PSI-DL	PSI-DL	PSI CGI Document List	current MS (doc version)
PSI-CST	PSI-CST	PSI Case Study	see before
PSI-ICD	PSI-ICD	PSI Interface Control Document	see before
PSI-MADR	PSI-MADR	PSI Markdown Administrative Decision Records	see before
PSI-REQ	PSI-REQ	PSI Requirements	see before
PSI-TAD	PSI-TAD	PSI Terms, Abbreviations and Definitions	see before
PSI-TOD	PSI-TOD	PSI Tasks and Operations Dictionary	see before

Table 1.4: Reference Documents.

2 Introduction

The Pooling & Sharing Interfaces Definitions (PSID) project is an ESA co-funded effort to define a common standard for the interfaces of Pooling & Sharing Systems (PSS) for Satellite Communication (SatCom) services. A PSS is a digital platform for matchmaking (Gov)SatCom users' demands (both commercial and institutional) with (Gov)SatCom providers' offers. Bringing together multiple (Gov)SatCom providers in one platform makes the market transparent, thus allowing users to get an overview of the market and to compare different offers efficiently. Additionally, a PSS assists users with little knowledge about the (Gov)SatCom domain defining their requirements on the (Gov)SatCom services. Those two aspects combined allow for fast access to the services and an efficient usage of the available capacities. To accomplish this, a PSS steps in between the usual processes of finding a provider/supplier, requesting an offer, and ordering the desired products or services, either as a service broker or by pooling products and services from different providers and offering them as an intermediary or distributor. Subsequently, the PSS can be used to monitor the services and manage multiple missions in a single application.

Eventually, a PSS can also be used as (or manage) a community hub, i.e., a number of end users or customers with similar interest that *share* their common resources and utilize a commonly obtained *pool* of (Gov)SatCom capacities. This strategy increases the efficient usage of scarce resources further.

There are already different approaches on PSSs, that might lead to an unnecessary fragmentation of the market. Therefore, a common standard for the interfaces of a PSS is required to allow the interaction between those different PSSs and reduce the effort of (Gov)SatCom providers to offer their product and services via multiple PSSs to maximize their reach.

Such a standard needs to take care of the different interfaces involved in the aforementioned processes, i.e.,

1. an interface between PSS and resource providers (satellite operators, service providers, or other PSSs),
2. an interface between the PSS and users, and
3. an interface between PSS and its own governance.

The goal of this project is to mainly define aspect 1 and to develop a software mock-up as needed to validate the various interfaces being developed.

The PSI standard derives from the existing industry-standard "Open Digital Framework" of **TM Forum** alliance[1]. The "Open Digital Framework" is a reference framework for delivering online Information, Communications and Entertainment services to the telecom world. It empowers market participants to compete and cooperate. One of PSI's goals is to make this existing standard fit for the world of satellite communication.

The consortium for this project consists of the service & technology providers SES Techcom and CGI, as well as of the (Gov)SatCom operators SES, Hellas Sat, Hispasat, Hisdesat, and LuxGovSat, and Inmarsat being both a service & technology provider and a (Gov)SatCom operator.

[1] See https://www.tmforum.org/resources/reference/gb991-tm-forums-core-concepts-and-principles-v22-0-0/

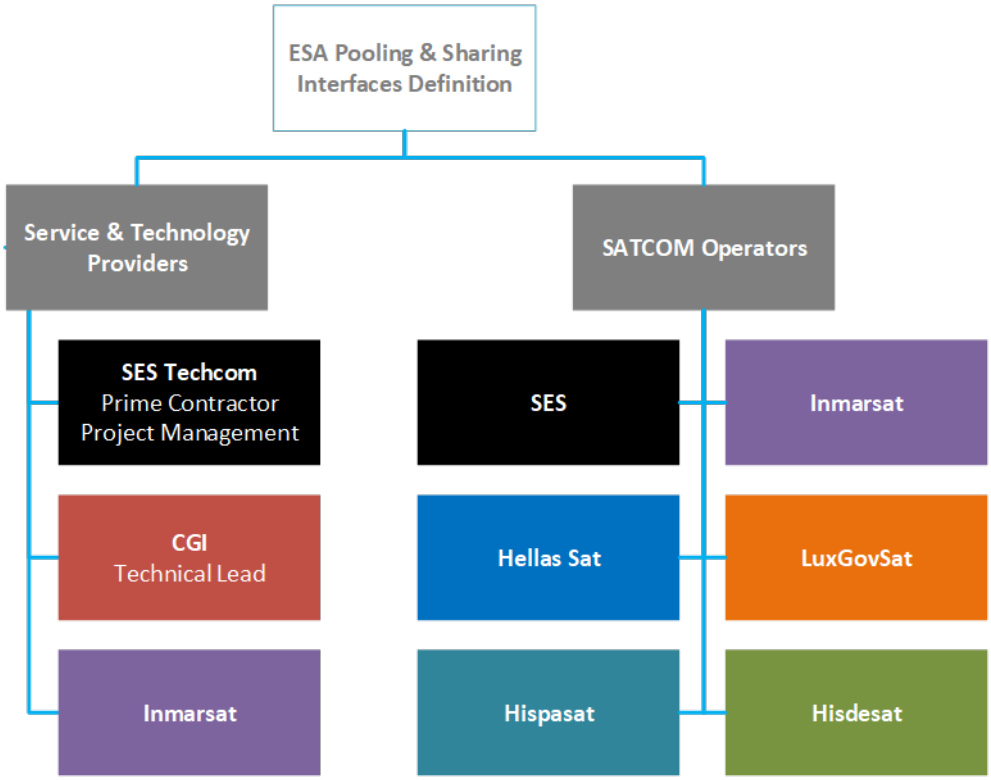

Figure 2.1: The PSI consortium.

2.1 Document Scope

This document contains all explanations of tasks and operations supported by the Pooling And Sharing Interfaces Definition Project (PSID) and how a PSS/Provider can integrate these using the interfaces described in the [PSI-ICD].

The following sections heavily refer to terms, abbreviations and definitions defined in the [PSI-TAD].

> Note: The [PSI-TOD] does not contain workflows as a guideline for concluding certain business processes. It is rather a technical description of all business tasks and operations that are covered by the PSI project, and how they are realized through the standardized interfaces. For the accommodation of case studies as compilations of business cases inspired by business processes collected from the business partners, cf. [PSI-CST].

2.1.1 Compiled Document

NOTE: THIS IS A COMPILED DOCUMENT [2]

This document has been compiled/generated from external sources and is not being written as-is. Therefore, any changes made within this compiled version of the document will be lost upon recompilation!

To make (permanent) changes, edit the respective sources directly or contact the PSID team.

[2] Document compiled on 2025-02-12 11:46.

2.1.2 Signature

Changes to this document are tracked electronically. No signature is required by the authors. The information in the “Source Control” chapter can prove the integrity of the document and reveal any change.

2.1.3 Development State

Current document version is `1.2.2`.
Next version is targeted for `2025-04-01`.

2.1.4 PSI Release Notes

2.1.4.1 Introduction

Welcome to the third release of the Pooling and Sharing Interface API! Below, you’ll find details about the features, enhancements, and other important aspects of this release.

2.1.4.2 Key Highlights

The focus of this release lies on **mission management**, to facilitate a common understanding of user requirement towards communication. It aims to complement the Inquiry API by providing workflows and *understanding* to service requirements. This is mainly a user-oriented API, but it also enables exchange of mission data between PSS systems and therefore cross-platform-market places. This could become a future focal point. Such data exchange would include actual user requirements (expressed as missions), as well as templates for such missions. By the use of templates, user mission creation is streamlined and allows a governance to safeguard, streamline or ease the process of user requirement gathering.

Together with the APIs, we are working on a Plug&Play component for P&S systems (Hubs, Brokers, Marketplaces...), based on ODA. This will be a standalone Micro-Frontend open to be integrated into existing OSS/BSS/PSS systems. A first draft is included in this release.

It will come with different views:

- Time based (e.g. mission timeline, Gantt-Chart, to express that is needed *when*)
- Geography based (e.g. mission zones or network nodes on a map to express what is needed *where*)
- Logical View (e.g. communication interdependency graph to express *how* the requirements will look like)

Another area of improvement is the **performance management API**.

A new API has been added that allows to request performance reports to an ongoing mission from the provider. That is: the report itself is generated on provider’s systems. The API handles the request and exchange of the report. The report has to be in line with the product’s SLA and allows monitoring of compliance. It allows also to define alarm thresholds and receive a push of threshold violations by the provider, avoiding a constant pull.

We also added the technical considerations and resulting decisions to the document set. This allows easier future evolution and maintenance of the standard.

2.1.4.3 What's New

- [PSI-GID] now contains descriptions about the ODA component for mission management
- [PSI-ICD] now contains new and updated APIs - see below!
- [PSI-ADR] first release of our decision records
- [PSI-TAD] now contains descriptions of concepts for user missions, as well as performance and alarm management
- [PSI-TOD] now contains new tasks and operations for user missions, performance and alarm management

2.1.4.3.1 Newly added APIs

- PSID002 Mission Management
- This customer-facing API allows them to manage missions and assign products, services and resources to them.
- It can also serve as an entry point for the Customer Inquiry API to find matching products for their requirements.
- PSID143 Performance Monitoring
- Based on MEF143 - Performance Monitoring API (Version 2.0.0-RC).
- The performance monitoring allows a PSS or customer to request performance reports from a provider.
- PSID642 Alarm
- Based on TMF642 - Alarm API (Version 4.1.0).
- Allows the provider to notify a PSS or customer about detected problems with their products.

2.1.4.3.2 Updates APIs

- PSID001 Customer Inquiry
- Improved handling of places by adapting TMF Geography types.
- PSID620 Product Catalog
- Based on TMF620 - Product Catalog Management API (Version 4.1.0).
- Changed `SLARef` to `ServiceLevelSpecificationRef`
- Streamlined GeoJSON types
- PSID633 Service Catalog
- Based on TMF633 - Service Catalog Management API (Version 4.1.0).
- Changed `SLARef` to `ServiceLevelSpecificationRef`
- Streamlined GeoJSON types
- PSID634 Resource Catalog

- Based on TMF634 - Resource Catalog Management API (Version 4.1.0).
- Changed `SLARef` to `ServiceLevelSpecificationRef`
- Streamlined GeoJSON types
- PSID657 Service Quality Management
- Based on TMF657 - Service Quality Management API (Version 4.1.0).
- Add endpoints to manage KPIs that are supported by the PSS.

2.1.4.3.3 Added Requirements

- MISSION requirement category
- REQ-06-03 Key Indicator Management
- REQ-06-04 Performance Monitoring Job Management
- REQ-06-05 Performance Monitoring Report Management
- REQ-06-06 Alarm Management

2.1.4.4 Known Limitations

1. The Service Quality Management is rather basic. There is an ongoing effort to align this set of APIs with the results of a TM Forum Catalyst project. More information will follow in one of the next releases.
2. The Mission Management Service is at an early state. However, the available API implements basic mission management services, import and export. A full set of APIs to implement such a service are subject to an upcoming release. Refer also the [PSI-GID] to learn about the available API use cases.

2.1.5 Outlook

Currently, we are working on the next release with the following focal points:

- Finalize the mission management component
- Update the API baseline to TM Forum 5
- Converge with MEF schema for some selected APIs

2.1.6 Feedback and Contributions

We value your feedback! If you encounter any issues or have suggestions, please reach out. Additionally, we welcome contributions from the community.

3 Preamble

The Pooling and Sharing Interfaces Definition (PSID) project is based on the Open Digital Framework of the TM Forum alliance[3]. It defines processes and business entities that are commonly used by telecommunication providers to achieve the best possible compatibility between them towards rapidly transforming their business operations into multipartner digital services. Although most of the work is built around terrestrial communication services, satellite providers are interested to adapt it, too. PSID follows the same domain structure, including but not necessarily restricted to:

- Common
- Customer
- Product
- Service
- Resource

The actual specification of the standardized APIs is based on the Open APIs of TM Forum, which are closely related to all other TM Forum frameworks. They are entity-centric and manipulate resources that are found in the Information Framework (SID) and they play part in a number of business processes that are defined in the Business Process Framework (also known as eTOM).

The Business Process Framework (eTOM) categorizes business activities in a structured manner starting from high-level core processes that are further decomposed by lower-level activities. There is a wide range of business processes which are adopted in the different domains, but this document outlines only relevant ones for the PSID project, excluding all that are not required.

One-to-one mapping between the business processes of the eTOM and the API specification is rather challenging. Therefore, the descriptions of tasks and operations for the PSID are tailored in such a way as to match the interactions between entities providing/consuming satellite communication services via standardised interfaces, while the related process identifiers from the eTOM are referenced where applicable.

The criteria used to build the list of tasks and operations are:

- The candidate requirements, which define the functionalities that should be supported by the interfaces.
- The identified needs of the consortium partners via direct communication with them.

[3] See https://www.tmforum.org/resources/standard/gb991-odf-concepts-and-principles-v22-0-0/

4 How to Read this Document

This document is structured on three hierarchical levels:

- Categories,
- Tasks, and
- Operations.

Categories follow the different business processes a PSS can implement to group the tasks by their general topic. For example, the category TOD-02-Product-Publishing collects all tasks that are related to registering and publishing resources, services, and products. The first category TOD-01-Miscellaneous is an exception. Here, some general topics are collectively discussed, e.g., the Event API that is used both for inquiries and ordering.

A task is more specific and handles only a couple of activities that are tightly connected. For example, the task TOD-02-03-Product_Catalog_Management collects all the operations that are necessary to manage the product catalog, e.g., creating, updating, viewing or deleting product specifications from it.

Operations are the low-level activities required to perform a certain task. An operation is connected to a REST endpoint of the API and therefore involves two entities:

1. The server (e.g. a PSS) that *implements* the endpoint of the operation.
2. The client (e.g. a provider) that *uses* the endpoint.

For example, the operation TOD-02-03-02-Update_Product_Specification outlines how the PATCH endpoint `/psi-api/productCatalog/v1/productSpecification/{id}` described in the [PSI-ICD] is to be used to update a product specification in a PSS's product catalog. Additionally, each operation references the corresponding PSI requirement(s). The description of the listed PSI requirements can be found in the [PSI-REQ] document by searching for the requirement ID. They define the functionality that should be covered by the implementation of the operation endpoint(s). Thus, it is easy to make a connection between the PSI requirements and the endpoints.

The naming of an operation follows this hierarchy: The first two digits enumerate the category, the second two the task, and finally the third two digits the operation.

Tasks and operations each follow a standardized structure.

4.1 Template for a Task

The description of a task starts with the section title stating its ID for reference with four digits (*TOD-XX-XX*) and its name (*Task_Template*).

It is followed by a descriptive text on the task summarizing the business process it covers and giving some hints on the low-level operations that need to be performed to conclude it. In general, all operations collected under a common task are implemented as REST endpoints either by a provider system or a PSS, that take the role of a server.

The REST endpoints are then invoked by users, the PSS governance, and other PSS-s or providers, that take the role of clients. The respective interfaces can be deduced based on the involved entities in the service-client relationship. For example, suppose the PSS is the server that exposes the resource registration operation as a

REST endpoint and the provider is the client that uses it to register its resources. In that case, the “PSS-Provider interface (I/F)” is involved.

The connection between the task and its subsidiary operations is displayed graphically below, while the matrix represents which clients can perform which operation. Checkmarks in parentheses indicate that these operations are carried out via events (see TOD-01-02). The interfaces covered by the task are graphically represented in the respective sections of the related operations.

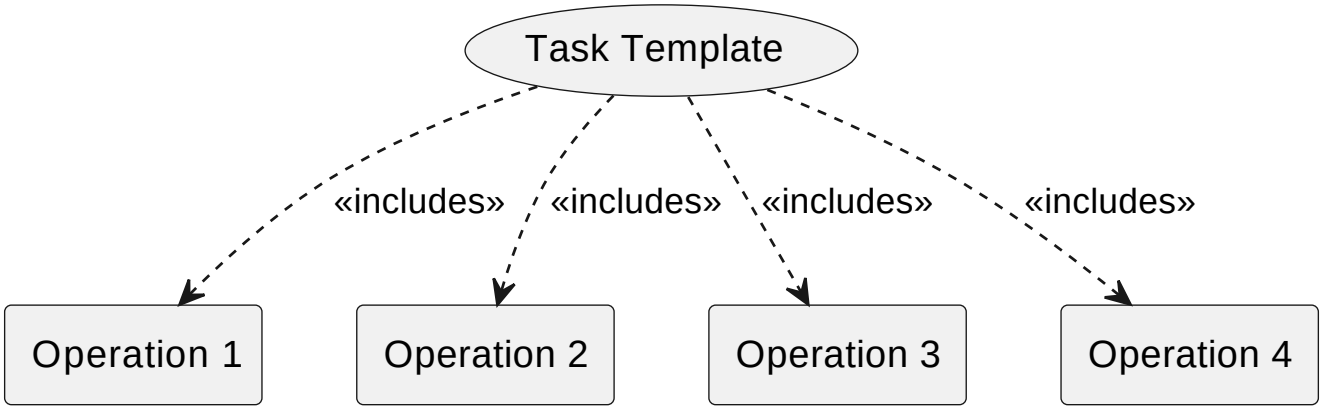

Figure 4.1: TOD-XX-XX: Task Template

	Customer	Provider	Other PSS	Governance
Operation 1	✓	✓		
Operation 2	✓	✓		
Operation 3	✓		(✓)	
Operation 4	✓		(✓)	

Table 4.1: Task Template Matrix.

Applicable Requirements

Some tasks have overarching requirements that apply to all operations. For details, see [PSI-REQ], which follows the same structure as this document for easy reference.

- PSI-XX-XX-00-01
- PSI-XX-XX-00-02
- ...
- PSI-XX-XX-00-NN

eTOM Reference

Most tasks can be mapped to TM Forum’s eTOM in one way or another. Identified processes of eTOM v22.0.0 are linked by the eTOM *Process Identifier*. For more details about the eTOM processes, please check the eTOM Process Decomposition L3[4] and L4[5].

[4] See https://www.tmforum.org/resources/reference/gb921d-l3-process-decompositions-v22-0-0
[5] See https://www.tmforum.org/resources/reference/gb921dx-l4-process-decompositions-v22-0-0

4.2 Template for an Operation

Just like for the tasks, the section title gives a general overview of the operation. It states its ID for reference with six digits (*TOD-XX-XX-XX*) and its name (*Operation_Template*).

Below, the operation is shown graphically. Here, the PSS takes the role of the server and implements the operation as REST endpoint(s), while the actors above are the clients that invoke the endpoint(s) to perform the operation. The interfaces involved are depicted in the PSI layer of the diagram. For example, the customer uses the operation implemented by the PSS via the PSS-USR interface.

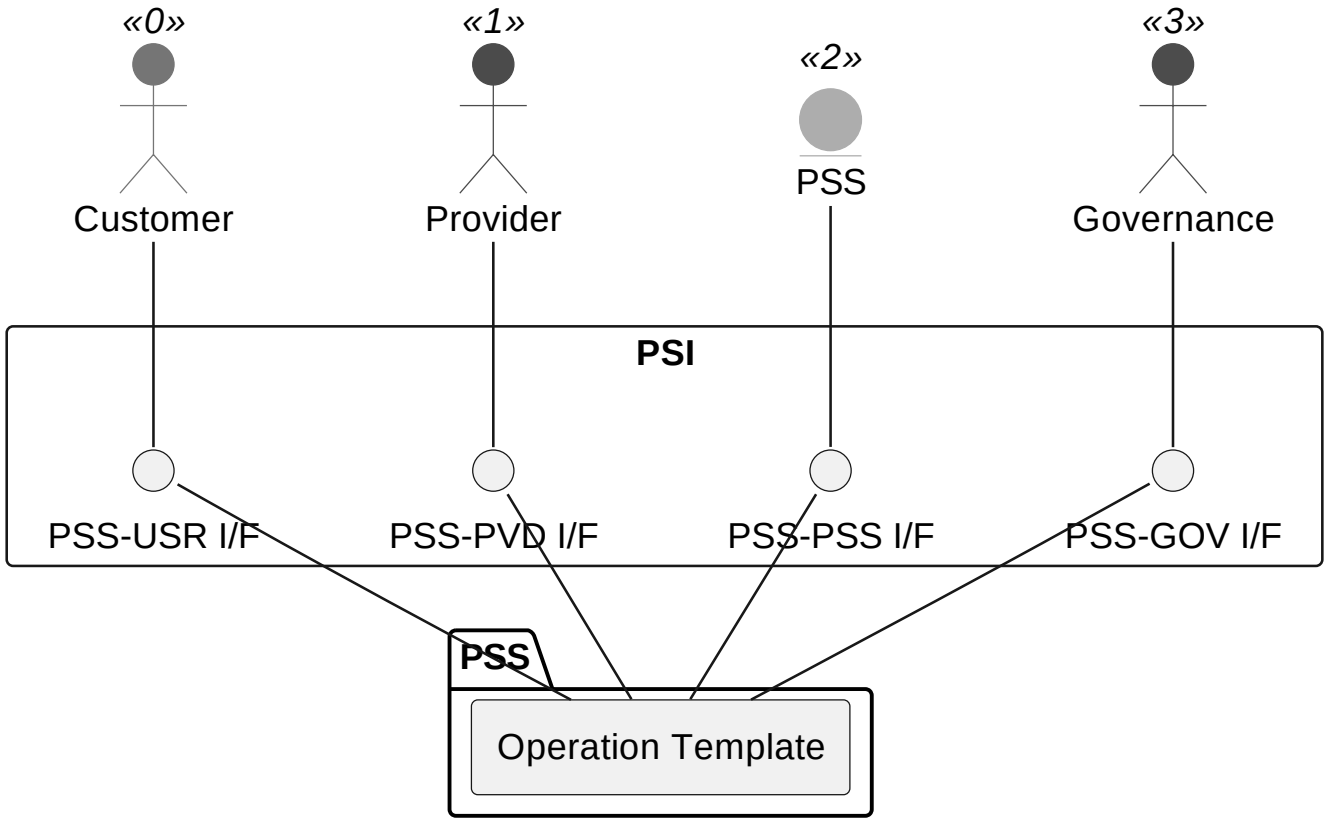

Figure 4.2: TOD-XX-XX-XX: Operation Template

Prerequisites

Some operations might have prerequisites to be performed. For example, a party that already is registered to the PSS should not be allowed to register their profile again.

Main operation

Creates a profile for a party with basic party data, identification data, contact data and additional attributes, via a standard interface specification.

The party can be an individual or an organization.

REST Endpoints

The REST endpoints implementing the operation are listed here. Commonly, an operation has one or more associated endpoints with a common HTTP method like `POST`, `PATCH`, `DELETE`, or `GET`. They may look like listed below and can be referenced in the [PSI-ICD].

- POST /psi-api/foo/v1/bar

- PATCH /psi-api/foo/v1/bar/{id}
- PATCH /psi-api/foo/v1/baz/{id}
- DELETE /psi-api/foo/v1/bar/{id}
- GET /psi-api/foo/v1/bar/{id}

Post Conditions

Most operations have a post condition, i.e., there is a system change that can be checked. For example, if the operation TOD-01-01-01-Create_Party_Profile is performed successfully, the profile for the party is successfully created in the PSS datastore.

Applicable Requirements

Each operation is defined by one or more requirement listed below. For details, see [PSI-REQ], which follows the same structure as this document for easy reference.

- PSI-XX-XX-XX-01
- PSI-XX-XX-XX-02
- ...
- PSI-XX-XX-XX-NN

eTOM Reference

Most tasks can be mapped to TM Forum's eTOM in one way or another. Identified processes of eTOM v22.0.0 are linked by the eTOM *Process Identifier*. For more details about the eTOM processes, please check the eTOM Process Decomposition L3[6] and L4[7].

[6] See https://www.tmforum.org/resources/reference/gb921d-l3-process-decompositions-v22-0-0
[7] See https://www.tmforum.org/resources/reference/gb921dx-l4-process-decompositions-v22-0-0

5 Tasks and Operations

This chapter describes the main tasks and operations that are covered as part of the Pooling and Sharing Interface Definition (PSID) project. They are logically grouped in categories for a cleaner hierarchical structure. Tasks and operations that belong to the same category, usually relate to a common higher level business process.

Consequently, the hierarchy consists of three levels, each using a two-digit numbering.

- Level 1 - Categories
- Level 2 - Tasks
- Level 3 - Operations

The security aspects to be taken into account when implementing these are described in the **Security Considerations** section of the [PSI-MADR] document.

5.1 TOD-01-Miscellaneous

The Miscellaneous category contains tasks and operations that can't be grouped to a common business process.

5.1.1 TOD-01-01-Party_Management

The Party Management task takes care of the lifecycles of parties with whom the PSS has a relationship. A party can be a natural person (individual) or an organization.

A customer wants to utilize a PSS to find/inquire and then order resources, services or products offered by providers. A provider wants to pool its resources, services, products and offerings to a PSS, so that they can be found and ordered by customers. Another PSS needs to be able to view the resources, service, products and offerings as well.

Therefore, a new customer/provider or another PSS, needs to register themselves as a party to a PSS via a standard interface to start the business interaction. Additionally, they need to view the information stored in the PSS, update the party profile, or delete it from the PSS. The governance of the PSS needs also to be able to manage the different party profiles (e.g. create/update/remove/view a party profile or view all party profiles).

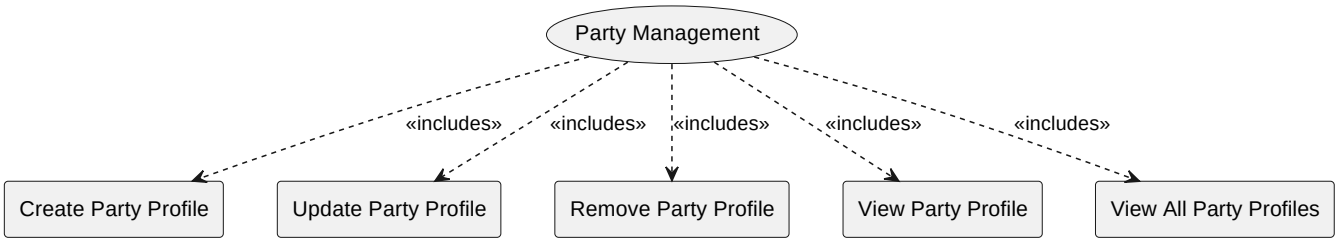

Figure 5.1: TOD-01-01: Party Management

	Customer	Provider	Other PSS	Governance
Create Party Profile	✓	✓		✓
Update Party Profile	✓	✓		✓
Remove Party Profile	✓	✓		✓

	Customer	Provider	Other PSS	Governance
View Party Profile	✓	✓	✓	✓
View All Party Profiles	✓	✓	✓	✓

Table 5.1: Party Management Matrix.

eTOM Reference

The task is based on the 1.3.6 and 1.6.21 process identifiers from the eTOM.

5.1.1.1 TOD-01-01-01-Create_Party_Profile

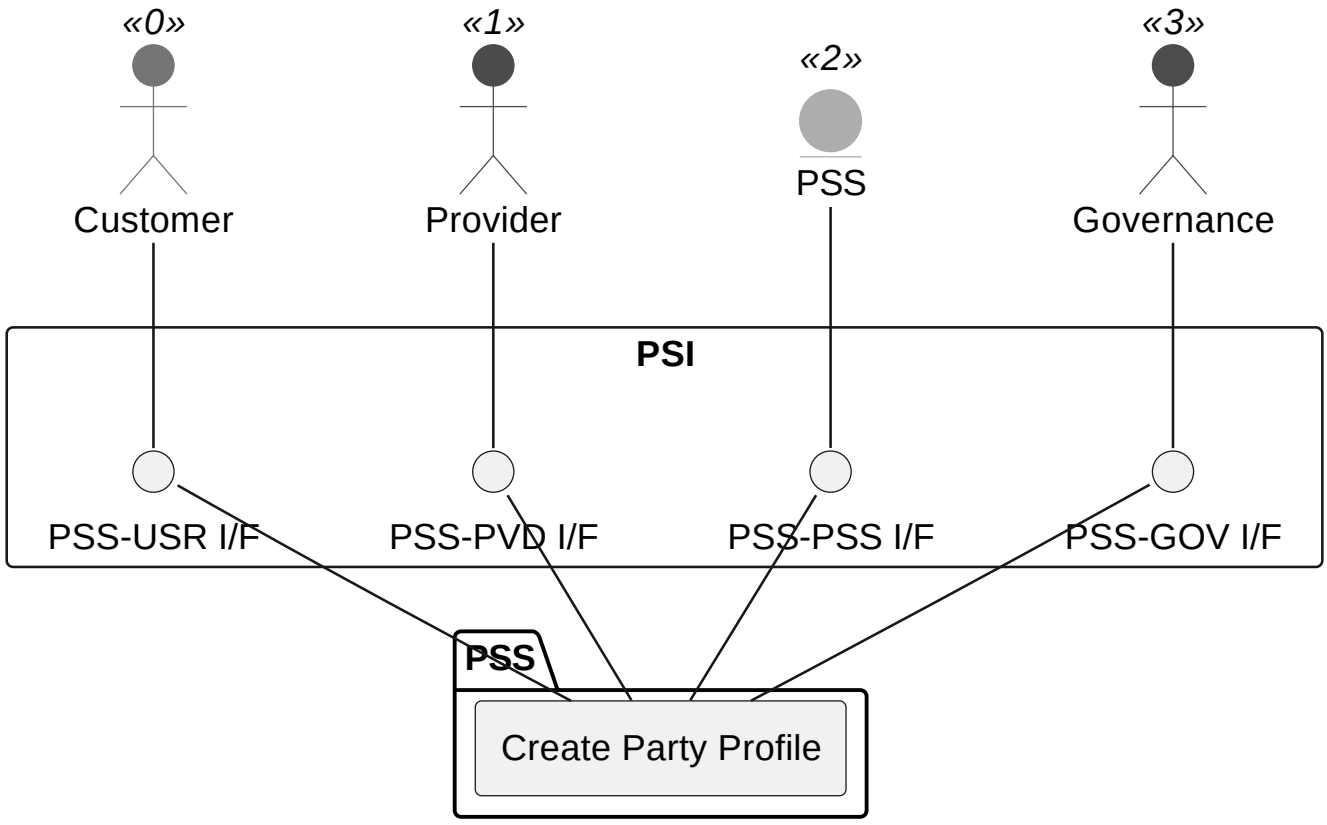

Figure 5.2: TOD-01-01-01: Create Party Profile

Prerequisites

The party has no profile.

Main operation

Creates a profile for a party with basic party data, identification data, contact data and additional attributes, via a standard interface specification.

The party can be an individual or an organization.

REST Endpoints

- POST /partyManagement/v1/individual
- POST /partyManagement/v1/organization

Post Conditions

The profile for the party is successfully created in the PSS datastore.

Applicable Requirements

- PSI-01-01-01-01
- PSI-01-01-01-02
- PSI-01-01-01-03
- PSI-01-01-01-04
- PSI-01-01-01-05
- PSI-01-01-01-06

eTOM Reference

The operation is based on 1.3.6.1 and 1.6.21.2 process identifiers from the eTOM.

5.1.1.2 TOD-01-01-02-Update_Party_Profile

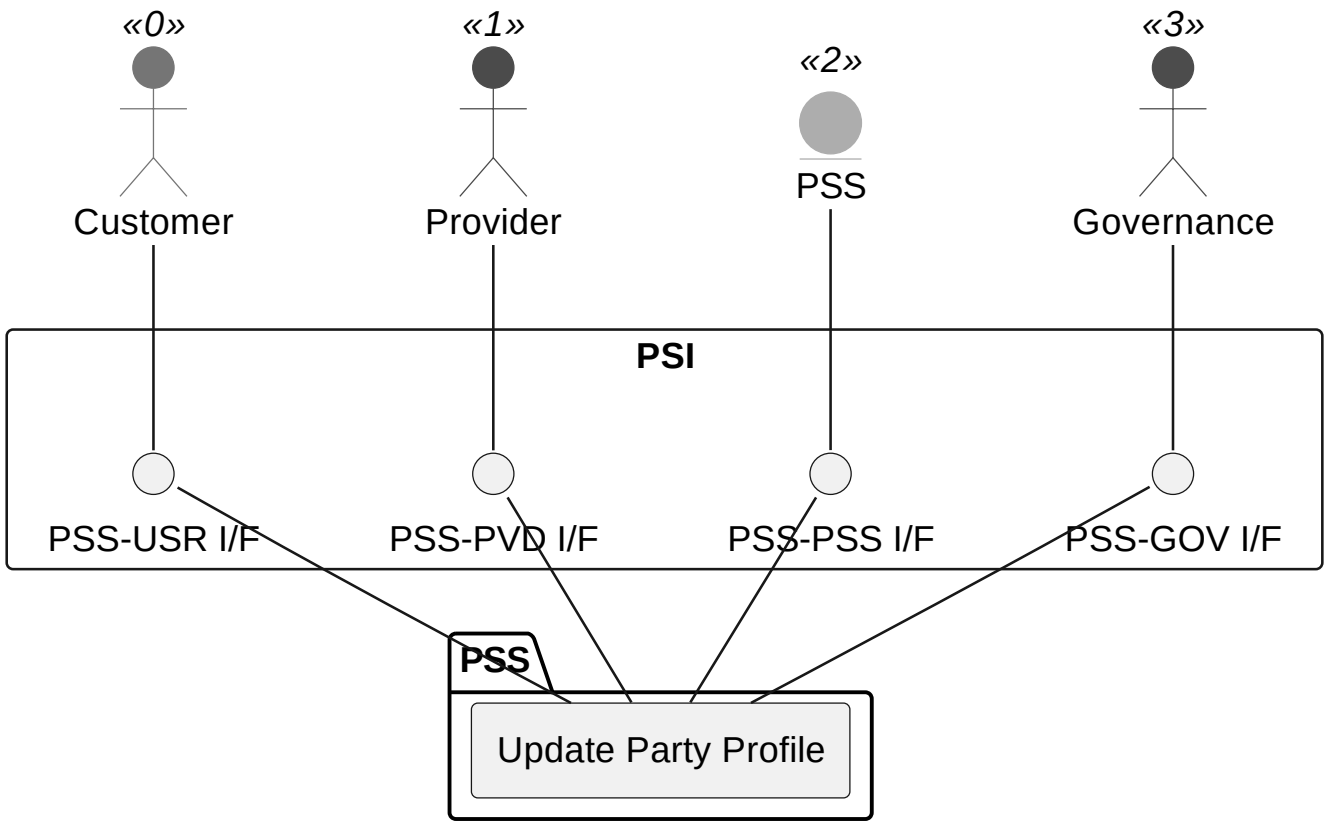

Figure 5.3: TOD-01-01-02: Update Party Profile

Prerequisites

The party has a profile.

Main operation

Updates an existing party's profile via a standard interface specification.

In addition to the standard party profile properties, the governance needs to be able to update additional characteristics as part of the accreditation process:

- *responseTime*: Specifies the time the provider is given to respond to orders or inquiries to avoid long delays on the customer side.
- *maxPriority*: Specifies the maximum priority of a customer when requesting access to resources.

REST Endpoints

- PATCH /partyManagement/v1/individual/{id}
- PATCH /partyManagement/v1/organization/{id}

Post Conditions

The profile for the party is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-01-01-02-01
- PSI-01-01-02-02
- PSI-01-01-02-03
- PSI-01-01-02-04
- PSI-01-01-02-05
- PSI-01-01-02-06
- PSI-01-01-02-07
- PSI-01-01-02-08
- PSI-01-01-02-09

eTOM Reference

The operation is based on 1.3.6.2 and 1.6.21.2 process identifiers from the eTOM.

5.1.1.3 TOD-01-01-03-Remove_Party_Profile

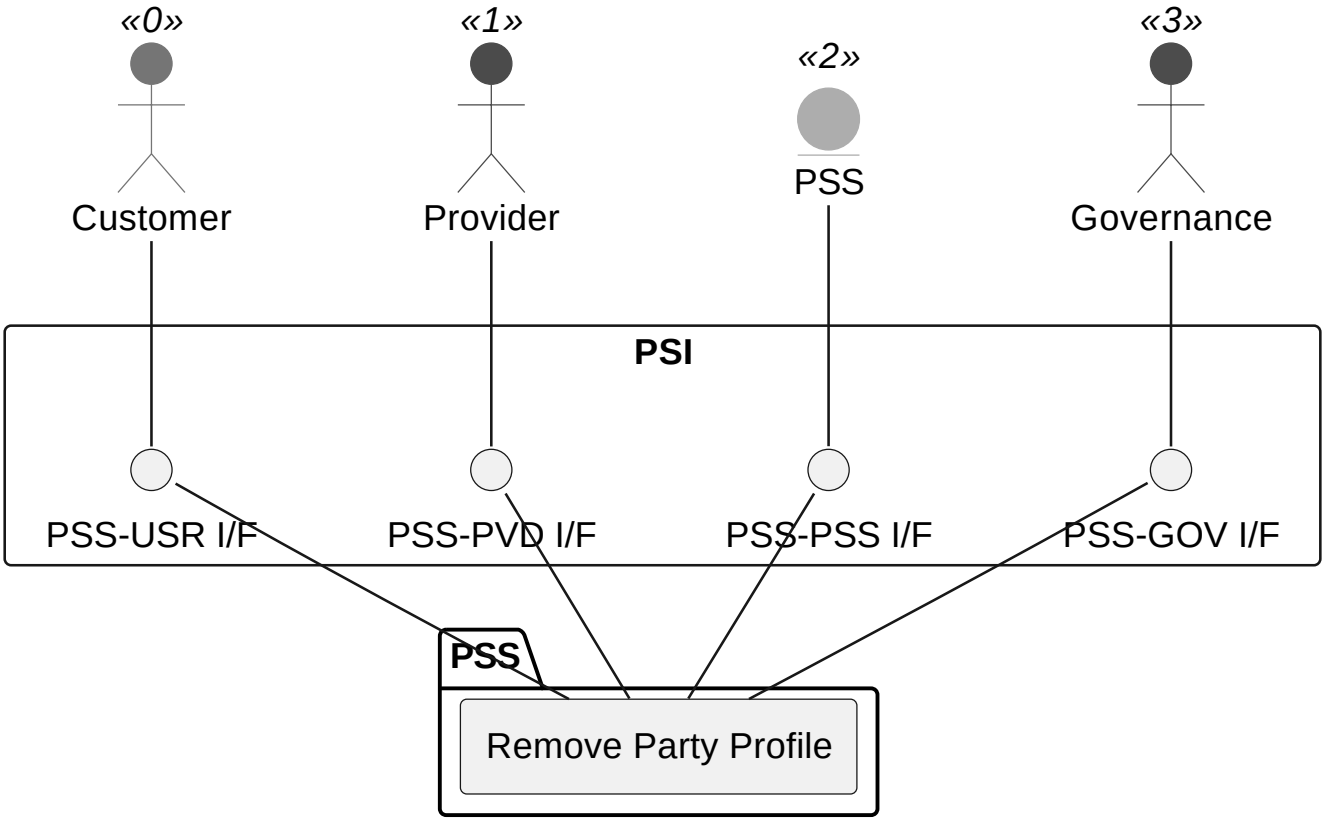

Figure 5.4: TOD-01-01-03: Remove Party Profile

Prerequisites

The party has a profile.

Main operation

Removes a party profile either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /partyManagement/v1/individual/{id}
- DELETE /partyManagement/v1/organization/{id}

Post Conditions

The profile for the party is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-01-01-03-01
- PSI-01-01-03-02
- PSI-01-01-03-03
- PSI-01-01-03-04

- PSI-01-01-03-05

eTOM Reference

The operation is based on 1.3.6.4 and 1.6.21.2 process identifiers from the eTOM.

5.1.1.4 TOD-01-01-04-View_Party_Profile

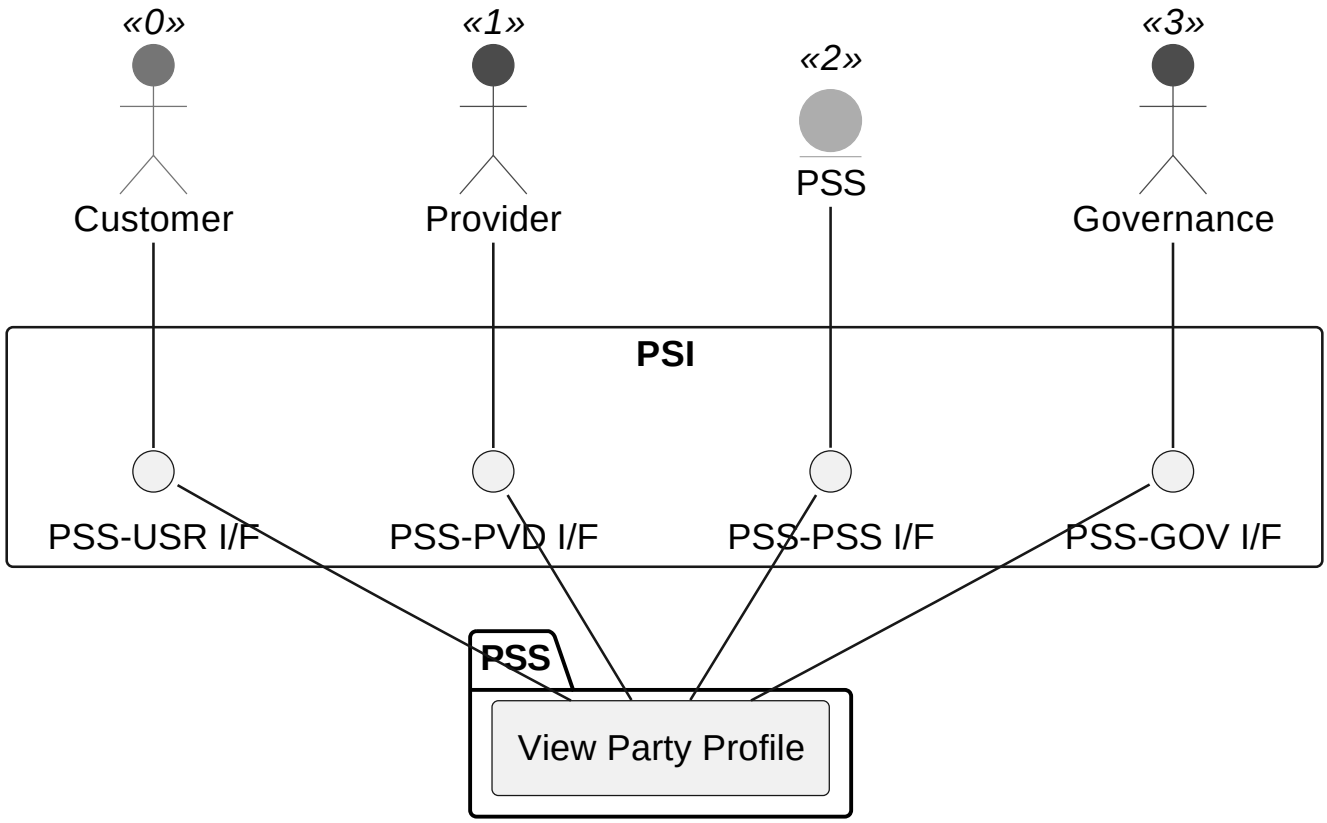

Figure 5.5: TOD-01-01-04: View Party Profile

Prerequisites

The party has a profile.

Main operation

Gets a party profile via a standard interface specification.

REST Endpoints

- GET /partyManagement/v1/individual/{id}
- GET /partyManagement/v1/organization/{id}

Post Conditions

The profile for the specified party is successfully returned to be viewed.

Applicable Requirements

- PSI-01-01-04-01
- PSI-01-01-04-02
- PSI-01-01-04-03
- PSI-01-01-04-04
- PSI-01-01-04-05

eTOM Reference

The operation is based on 1.3.6.5 and 1.6.21.2 process identifiers from the eTOM.

5.1.1.5 TOD-01-01-05-View_All_Party_Profiles

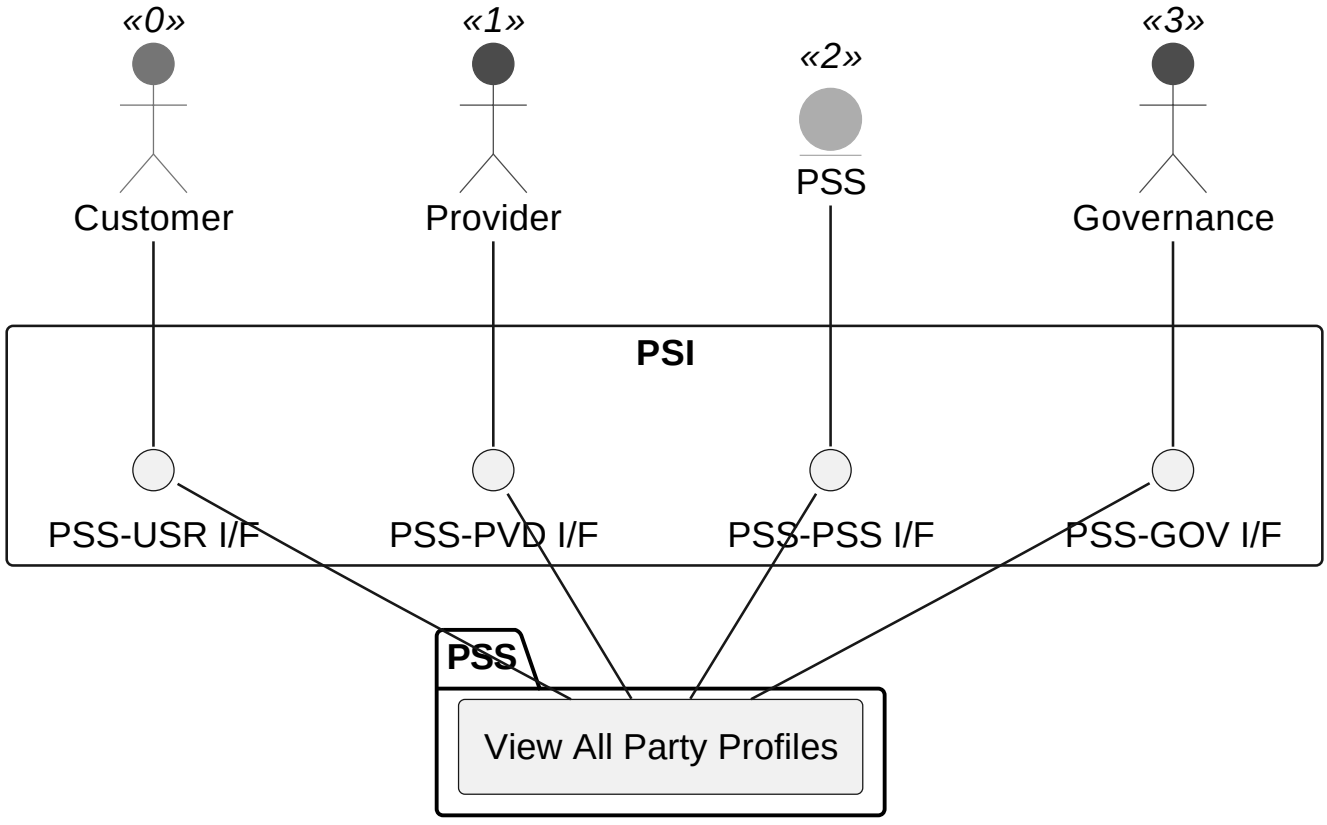

Figure 5.6: TOD-01-01-05: View All Party Profiles

Prerequisites

Party profiles exist in the PSS datastore.

Main operation

Retrieves associated party profiles. Access and visibility restrictions are the responsibility of the implementing PSS.

The Governance, Customer, Provider, and PSS can retrieve all party profiles via this interface.

REST Endpoints

- GET /partyManagement/v1/individual
- GET /partyManagement/v1/organization

Post Conditions

All visible party profiles are successfully returned to be viewed.

Applicable Requirements

- PSI-01-01-05-01
- PSI-01-01-05-02

eTOM Reference

The operation is based on 1.3.6.5 and 1.6.21.2 process identifiers from the eTOM.

5.1.2 TOD-01-02-Event_Management

Some processes between a PSS and a provider (or PSS and PSS), such as customer inquiries and orders, can take longer time to complete. For example, when a customer inquiry is created, the provider may require significant time to process and respond with an adequate product offering. Or, when a product order is placed by a customer, it can take hours to days for its state to change, e.g. from ‘inProgress’ to ‘completed’.

Inside a PSS (or a sophisticated provider system) the anticipated approach to propagate such state changes are message queues. A direct connection between these, although possible, would result in a strong coupling of the systems and major implications by the interface definition on the internal implementations. In order to avoid this, the Event Management defines how to exchange the information using REST.

> Note that this does not **enforce** the use of message queues. All named operations and endpoints can also be implemented in a monolithic application.

The Event Management task is based on the exchange of events between two systems. *Topics* are target containers for events which exist to store different events separated into domains. A PSS must have at least three topics: *order*, *inquiry* and *alarm*. Hence, all events related to orders are collected in the *order* topic, while the events of the inquiries are stored in the *inquiry* topic. Alarms related to breaching service level specification are stored in the *alarm* topic. However, PSS and provider systems are allowed to define additional topics, if needed, to organise the events in their interface implementation.

The following diagram illustrates a usual execution sequence, using the “order” topic as an example:

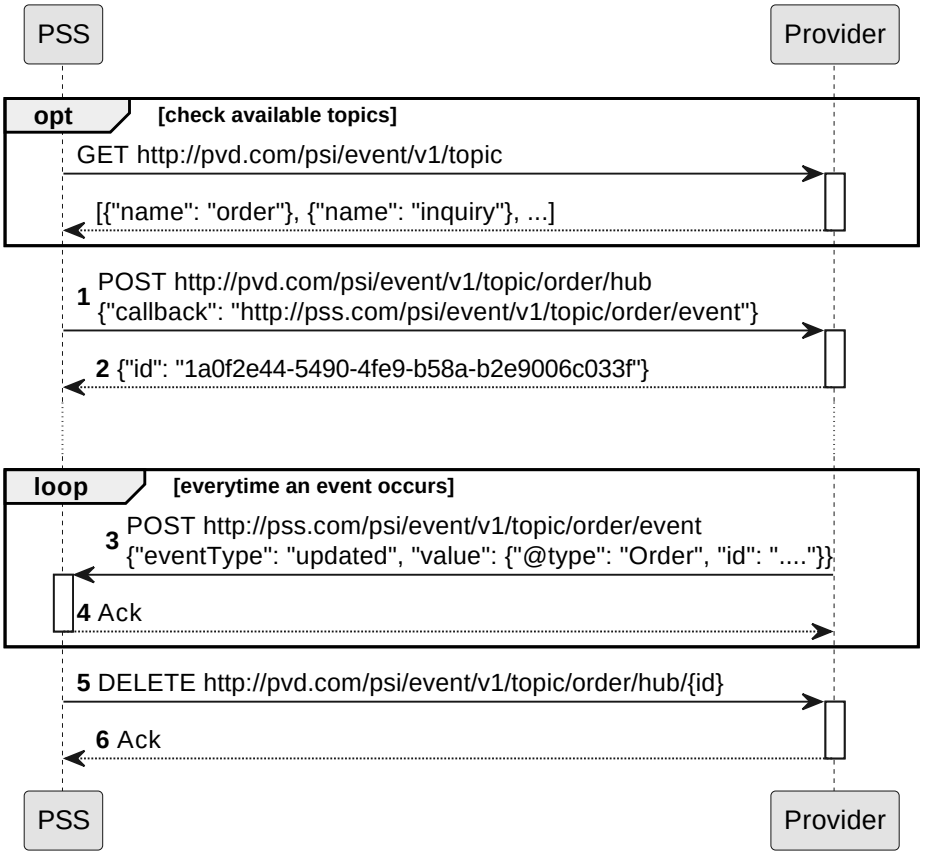

Figure 5.7: TOD-01-02: Event Management Sequence

The shown steps are further described in the following operations:

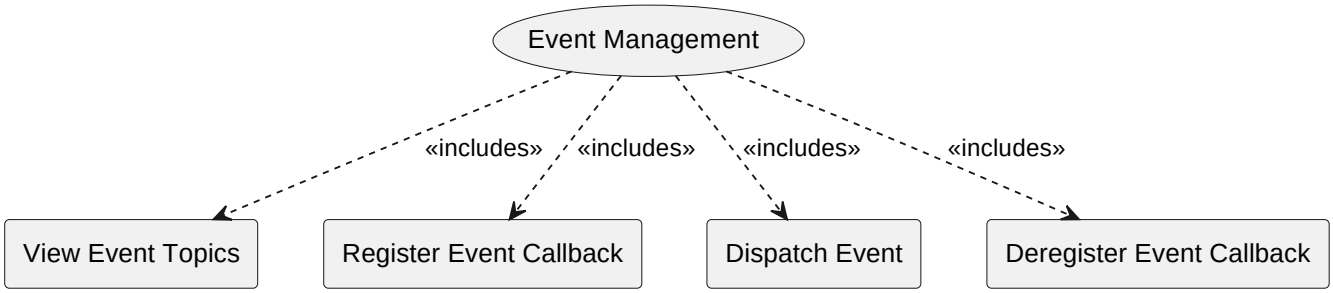

Figure 5.8: TOD-01-02: Event Management

	Customer	Provider	Other PSS	Governance
View Event Topics		✓	✓	
Register Event Callback		✓	✓	
Dispatch Event		✓	✓	
Deregister Event Callback		✓	✓	

Table 5.2: Event Management Matrix.

Please note that the Governance does not get direct access to the endpoints. Nevertheless, the Governance usually

has read access to the event data via the monitoring service.

eTOM Reference

None

5.1.2.1 TOD-01-02-01-View_Event_Topics

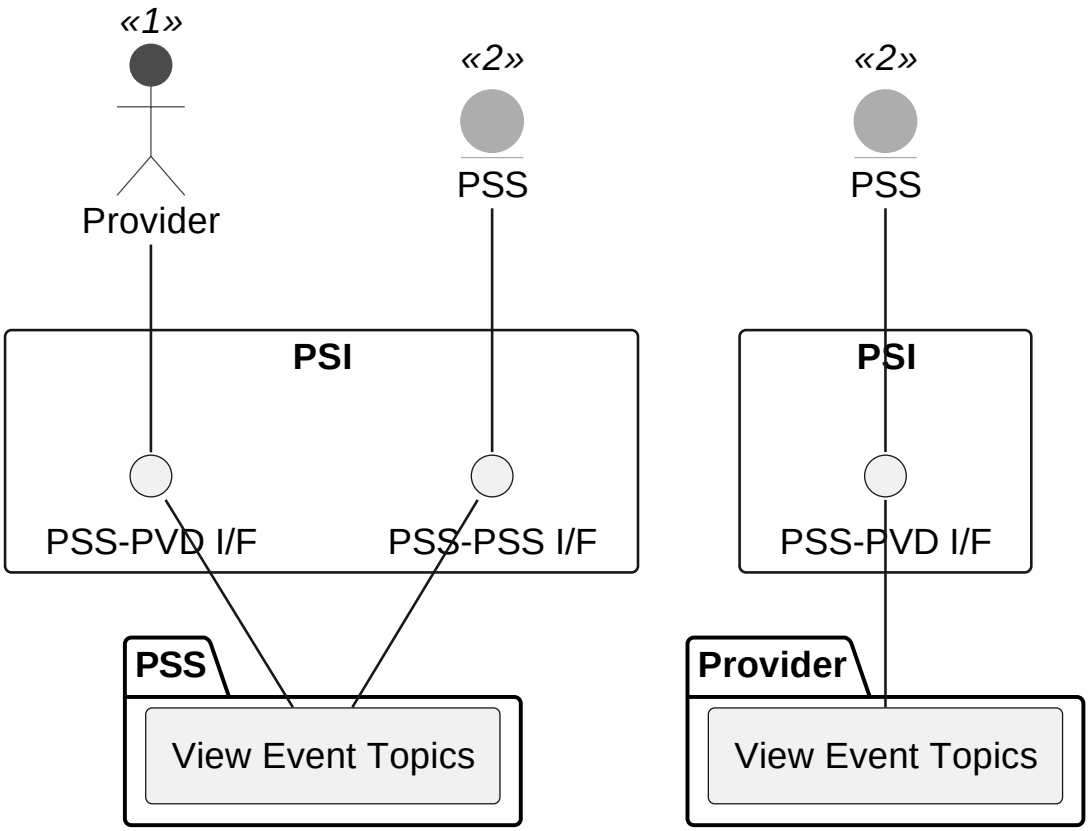

Figure 5.9: TOD-01-02-01: View Event Topics

Prerequisites

The event topics are created on the other system.

Main operation

This operation allows one system to query the available event topics of another system. Since the relevant names for PSID are predefined, doing so is considered optional, though it may reveal additional topics. The topics can then be subscribed to using TOD-01-02-02.

REST Endpoints

- GET /eventManagement/v1/topic

Post Conditions

All available event topics of the queried system (PSS or provider) are successfully returned to be viewed.

Applicable Requirements

- PSI-01-02-01-01
- PSI-01-02-01-02

eTOM Reference

None

5.1.2.2 TOD-01-02-02-Register_Event_Callback

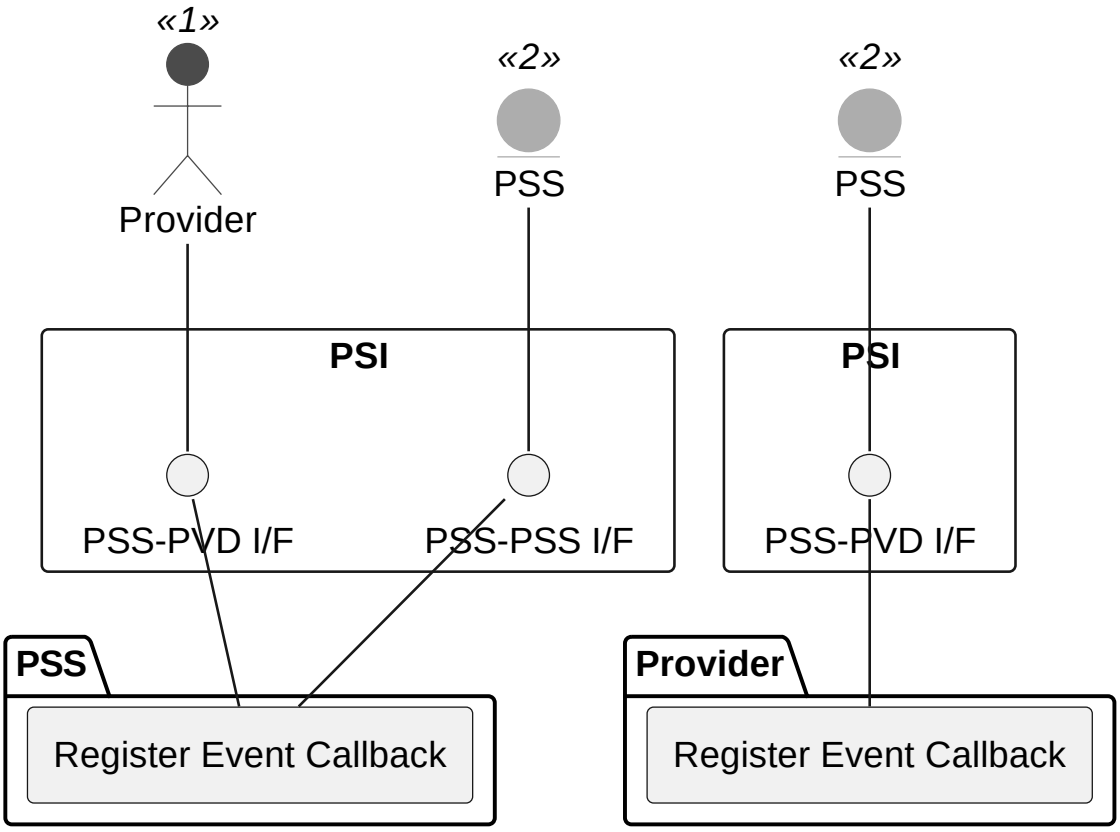

Figure 5.10: TOD-01-02-02: Register Event Callback

Prerequisites

At least one event topic must exist.

Main operation

Each system has to register callbacks for the events they want to receive from one another. Usually, both systems will register for the following topics named after the corresponding entities:

- inquiry
- order
- invoice

The registration must contain the URL of the callback endpoint, which is expected to be the one described in TOD-01-02-03. Note that the implementation may reject callbacks if the host is not whitelisted beforehand (see PSI-ICD).

Events are always filtered by the sending system on a need-to-know basis, e.g. the PSS will send order events only to the parties participating in the interaction. Additionally, the registration may contain a filter query to tailor the scope of the received events. Those can be defined as a conjunction of attributes and expected values. The available attributes depend on the event type and can be nested using dot-notation. When a list of items is queried, only one of them has to match the given value. Implementations may offer additional capabilities as described in the TMF630 REST API Design Guidelines 4.2.0[8]. For example, an order event could be filtered using:

- `event.priority=1` (receive only orders with high priority)
- `event.productOrderItem.productOffering.id=abcd` (receive only orders of a specific product offering)
- `event.category=InternetAccess&state=pending` (receive only orders with category 'InternetAccess' **and** state 'pending')

REST Endpoints

- POST /eventManagement/v1/topic/{topicId}/hub

Post Conditions

The callback is registered in the other system.

Applicable Requirements

- PSI-01-02-02-01
- PSI-01-02-02-02
- PSI-01-02-02-03

eTOM Reference

None

[8] https://www.tmforum.org/resources/specification/tmf630-rest-api-design-guidelines-4-2-0/

5.1.2.3 TOD-01-02-03-Dispatch_Event

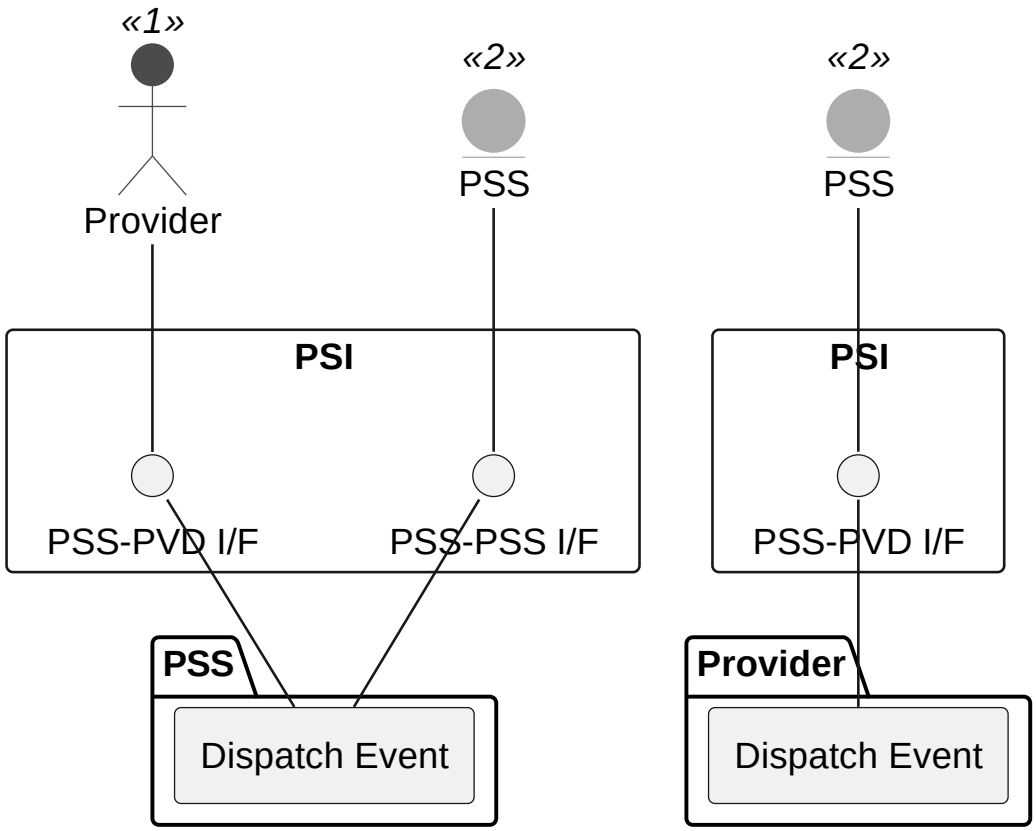

Figure 5.11: TOD-01-02-03: Dispatch Event

Prerequisites

At least one callback is registered in the system.

Main operation

Whenever an event occurs in a system (e.g. an entity is created, updated or deleted), it notifies all qualified callbacks previously registered by TOD-01-02-02. They are selected by:

- Event topic
- Need-to-know (e.g. the PSS will send order events only to the parties participating in the interaction)
- Matching filter query

The message body contains the event type and the whole affected entity, meaning there is no need to query it additionally. The `correlationId` helps to identify the event across the systems, as it will get a different `ID` in each.

The sender has to ensure that every qualified callback is called successfully exactly once. This includes failover mechanisms in particular. When the receiver is not reachable or responding positively, the message must be re-dispatched with exponential backoff. Hereby, the sending attempt is repeated with an exponentially increasing waiting time until a maximum number of retries is reached. In the latter case, where the delivery was completely stopped, only a manual intervention could restart the sending.

REST Endpoints

- POST /eventManagement/v1/topic/{topicId}/event

Post Conditions

The event is dispatched to all qualified callbacks.

Applicable Requirements

- PSI-01-02-03-01
- PSI-01-02-03-02
- PSI-01-02-03-03
- PSI-01-02-03-04
- PSI-01-02-03-05
- PSI-01-02-03-06
- PSI-01-02-03-07
- PSI-01-02-03-08
- PSI-01-02-03-09

eTOM Reference

None

5.1.2.4 TOD-01-02-04-Deregister_Event_Callback

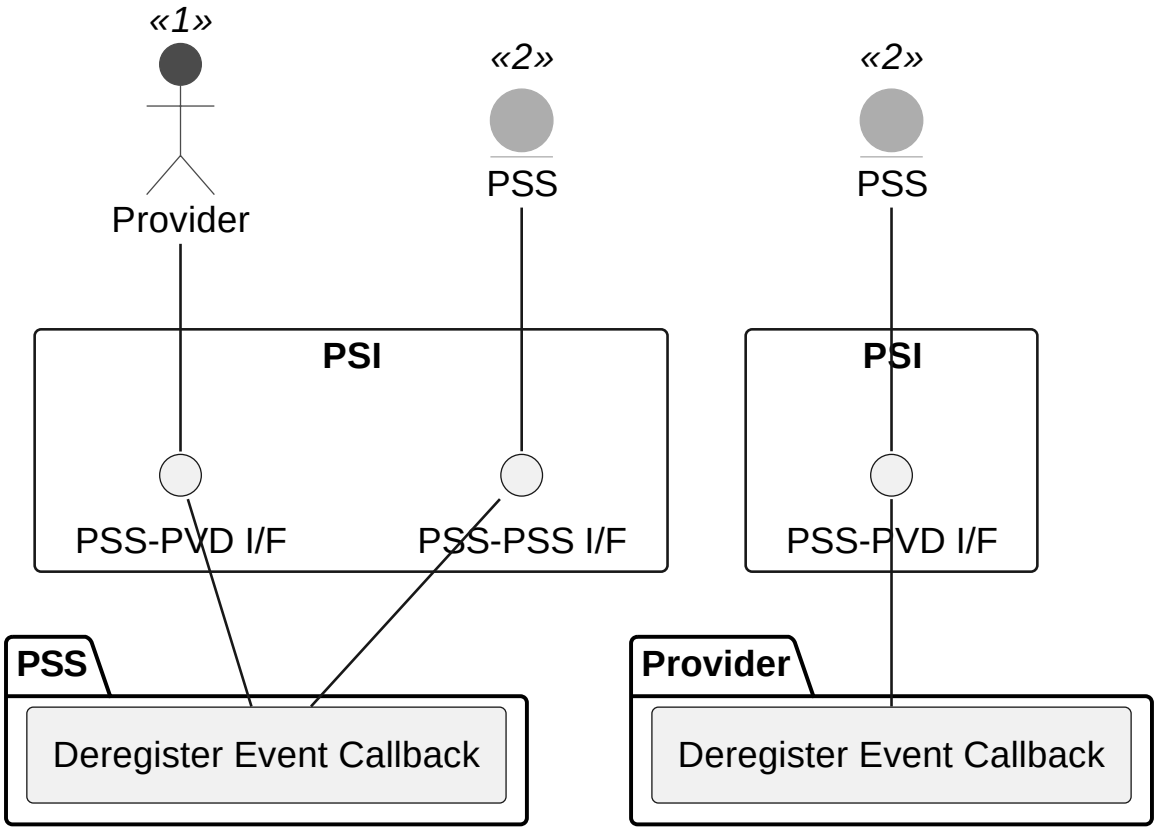

Figure 5.12: TOD-01-02-04: Deregister Event Callback

Prerequisites

The callback is registered in another system.

Main operation

When a system does not need to listen for an event topic any more, it can deregister its callback from other systems. The other system will consequently not dispatch events to the callback afterwards.

REST Endpoints

- DELETE /eventManagement/v1/topic/{topicId}/hub/{id}

Post Conditions

The callback is deregistered in the other system.

Applicable Requirements

- PSI-01-02-04-01

eTOM Reference

None

5.1.3 TOD-01-03-Document_Management

The Document Management task is responsible for managing documents that need to be exchanged between providers, customers and PSS. For example, a product offering of a provider is accompanied by a Service Level Agreement that should be shared with the customer via REST API, or when an order is concluded, an interface is required for sending the invoice.

Any party involved in the PSI processes needs to be able to create, update, remove or view available document(s). Each document can have arbitrary characteristics describing the document itself or the context of it, for example order numbers. It can also be directly linked to other documents or entities of other APIs.

The content of the document is stored in one or more attachments, which can be uploaded either binary or encoded in Base64 format along with the mime type e.g. “application/pdf”, “application/msword” or “image/jpeg”. The party implementing the interface endpoints should take care of performing a malware scan of the created attachments. While the PSID does not define how the content is stored, it enables use of cloud storage like S3 as well as plain filesystem access. In both scenarios, read/write access is handled by the same HTTP endpoint.

> Note that while the Document Management is derived from TM Forum Document Management API v4.0.0, it introduces a lot of (partially incompatible) changes, because the operations of this task are envisioned but not yet defined in TM Forum. Therefore, this task might be subject to change when a new version of TM Forum’s Document Management API will be released.

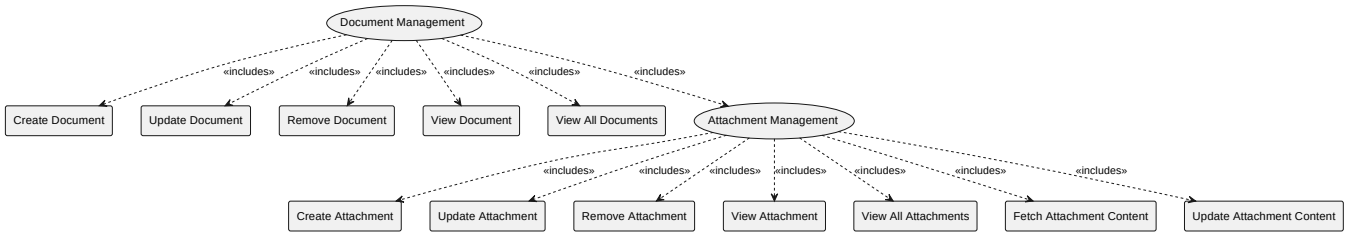

Figure 5.13: TOD-01-03: Document Management

	Customer	Provider	Other PSS	Governance
Create Document	✓	✓		
Update Document	✓	✓		
Remove Document	✓	✓		
View Document	✓	✓	✓	✓
View All Documents	✓	✓	✓	✓
Create Attachment	✓	✓		
Update Attachment	✓	✓		
Remove Attachment	✓	✓		
View Attachment	✓	✓	✓	✓
View All Attachments	✓	✓	✓	✓
Fetch Attachment Content	✓	✓	✓	✓
Update Attachment Content	✓	✓		

Table 5.3: Document Management Matrix.

eTOM Reference

None

5.1.3.1 TOD-01-03-01-Create_Document

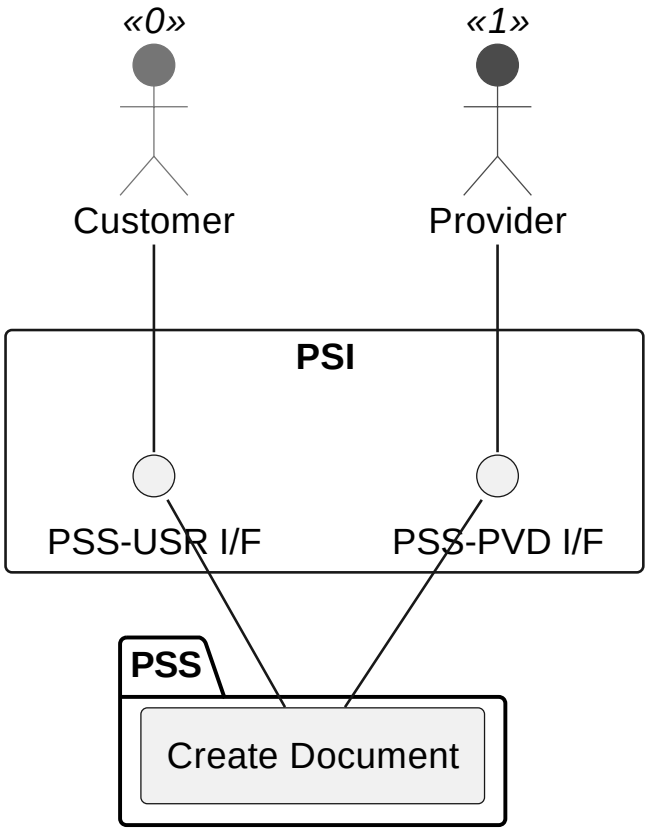

Figure 5.14: TOD-01-03-01: Create Document

Prerequisites

The document does not exist in the PSS datastore.

Main operation

Creates a new document with its characteristics via a standard interface specification.

Some properties of a Document are:

- *name* - Short name of the document (i.e. the title)
- *description* - Description or summary of the document
- *characteristic* - List of characteristics such as order number
- *relatedParty* - References to the parties involved in the document (e.g. provider and customer)
- *relatedEntity* - References to entities in other subsystems (e.g. product or services) which are related to the document

REST Endpoints

- POST /documentManagement/v1/document

Post Conditions

The document is successfully created in the PSS datastore.

Applicable Requirements

- PSI-01-03-01-01

eTOM Reference

None

5.1.3.2 TOD-01-03-02-Update_Document

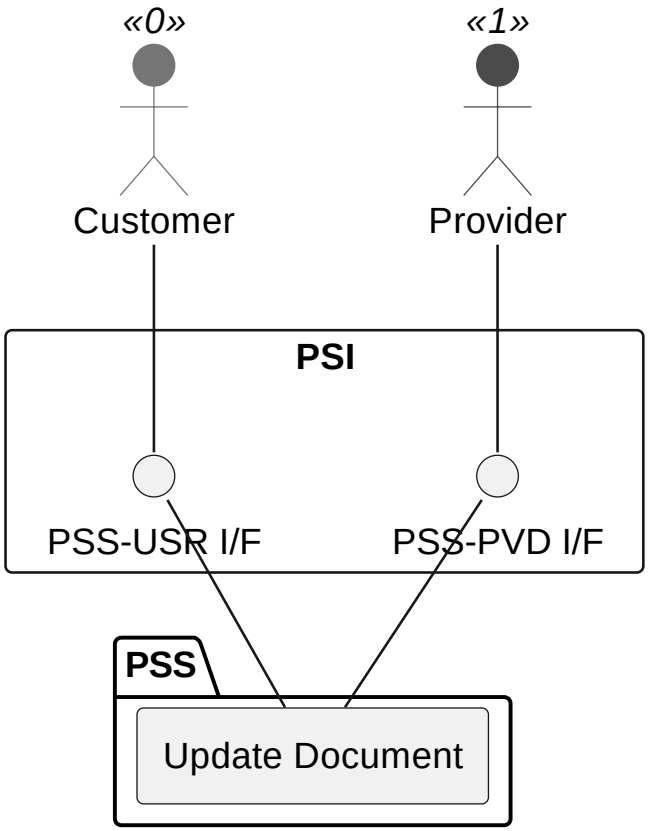

Figure 5.15: TOD-01-03-02: Update Document

Prerequisites

The document exists in the PSS datastore.

Main operation

Updates an existing document via a standard interface specification.

REST Endpoints

- PATCH /documentManagement/v1/document/{id}

Post Conditions

The document is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-01-03-02-01
- PSI-01-03-02-02

eTOM Reference

None

5.1.3.3 TOD-01-03-03-Remove_Document

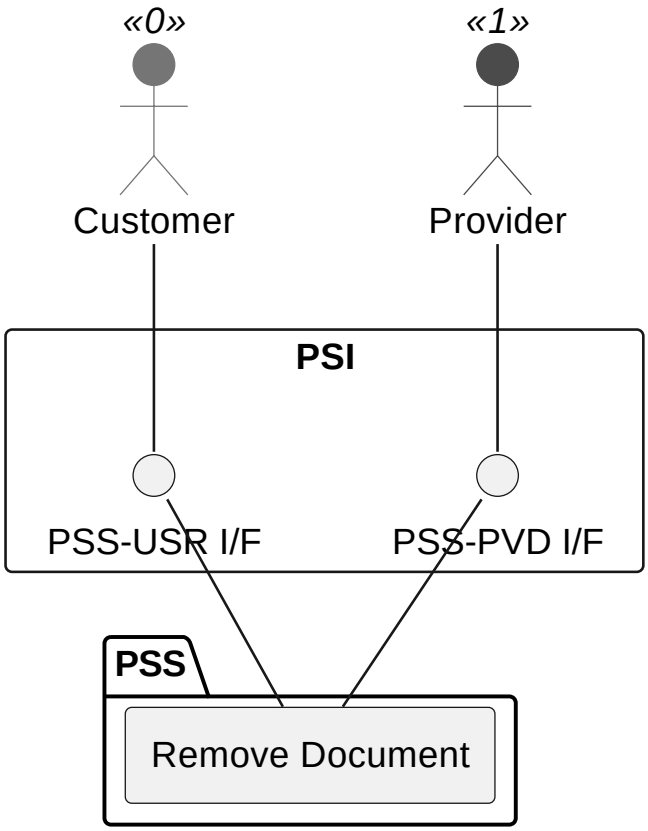

Figure 5.16: TOD-01-03-03: Remove Document

Prerequisites

The document exists in the PSS datastore.

Main operation

Removes a document either by deleting it or indicating it is no longer valid, via a standard interface specification.

Additionally, all associated attachments of the document are deleted or marked as not valid. The system implementing the interface should ensure that the attachments are removed from the physical location or file storage system.

REST Endpoints

- DELETE /documentManagement/v1/document/{id}

Post Conditions

The document is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-01-03-03-01
- PSI-01-03-03-02
- PSI-01-03-03-03

eTOM Reference

None

5.1.3.4 TOD-01-03-04-View_Document

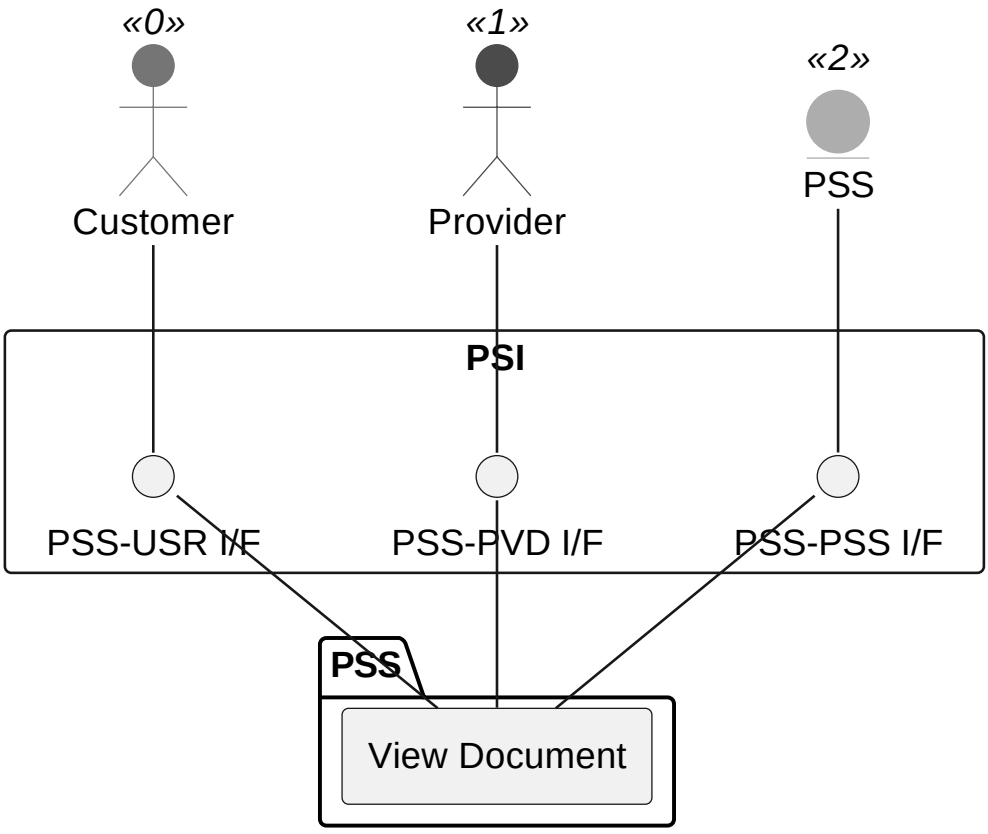

Figure 5.17: TOD-01-03-04: View Document

Prerequisites

The document exists in the PSS datastore.

Main operation

Gets a document with a specific identifier via a standard interface specification.

REST Endpoints

- GET /documentManagement/v1/document/{id}

Post Conditions

The document is successfully returned to be viewed.

Applicable Requirements

- PSI-01-03-04-01
- PSI-01-03-04-02

eTOM Reference

None

5.1.3.5 TOD-01-03-05-View_All_Documents

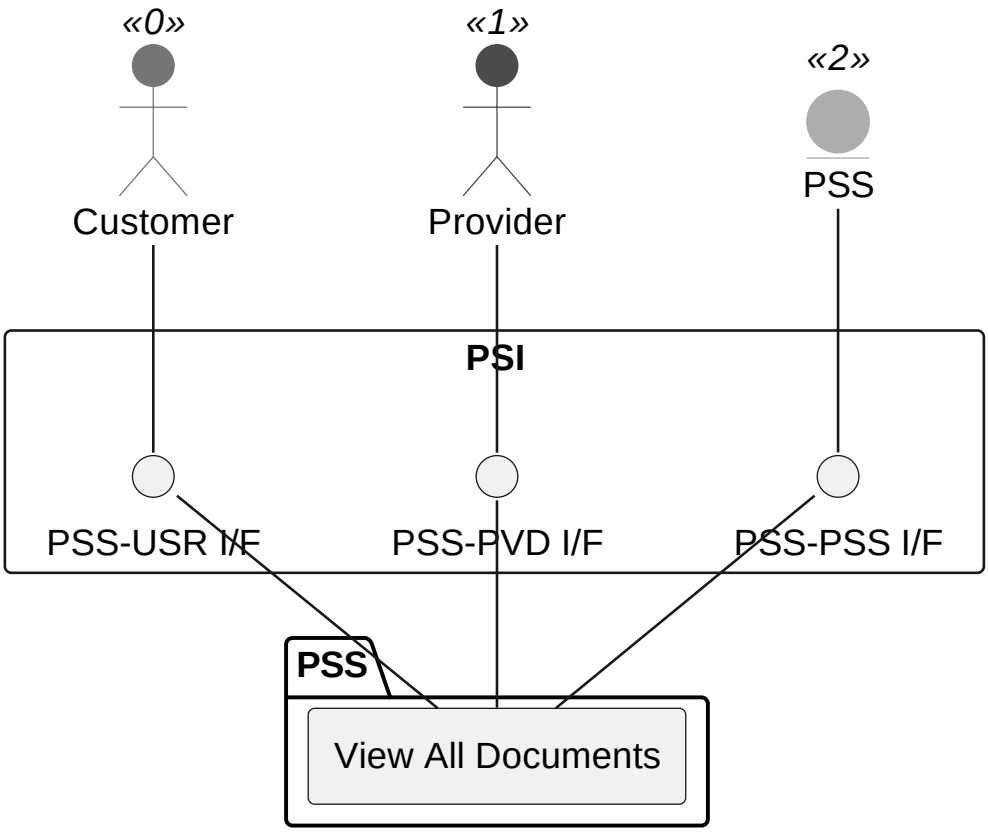

Figure 5.18: TOD-01-03-05: View All Documents

Prerequisites

Documents exist in the PSS datastore.

Main operation

Gets all documents that are applicable to the user requesting them.

REST Endpoints

- GET /documentManagement/v1/document

Post Conditions

All documents of the provider are successfully returned to be viewed.

Applicable Requirements

- PSI-01-03-05-01

eTOM Reference

None

5.1.3.6 TOD-01-03-06-Create_Attachment

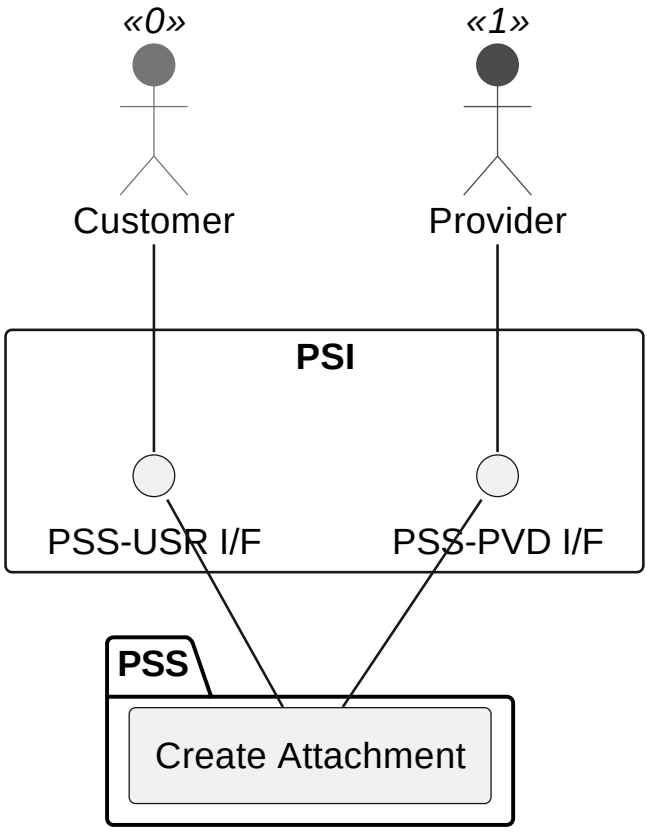

Figure 5.19: TOD-01-03-06: Create Attachment

Prerequisites

A document has to exist in the PSS datastore.

Main operation

This operation allows a party to create an attachment and transfer it to a PSS, which has to take care of storing it in some file storage system. Additionally, the PSS should take care of performing a malware scan of the created attachment.

The attachment must be created as part of an existing document which describes its metadata.

Some properties of the attachment are:

- *name* - The name of the attachment.
- *content* - The actual contents of the attachment object.
- *attachmentType* - The attachment type such as video, picture, document.
- *mimeType* - The attachment mime type such as “application/pdf”, “application/msword” or “image/jpeg”.
- *size* - The size of the attachment.
- *url* - The remote reference to the content if web-addressable.
- *validFor* - The period of time for which the attachment is valid.

REST Endpoints

- POST /documentManagement/v1/document/{documentId}/attachment

Post Conditions

The attachment is successfully created to the PSS.

Applicable Requirements

- PSI-01-03-06-01
- PSI-01-03-06-02
- PSI-01-03-06-03

eTOM Reference

None

5.1.3.7 TOD-01-03-07-Update_Attachment

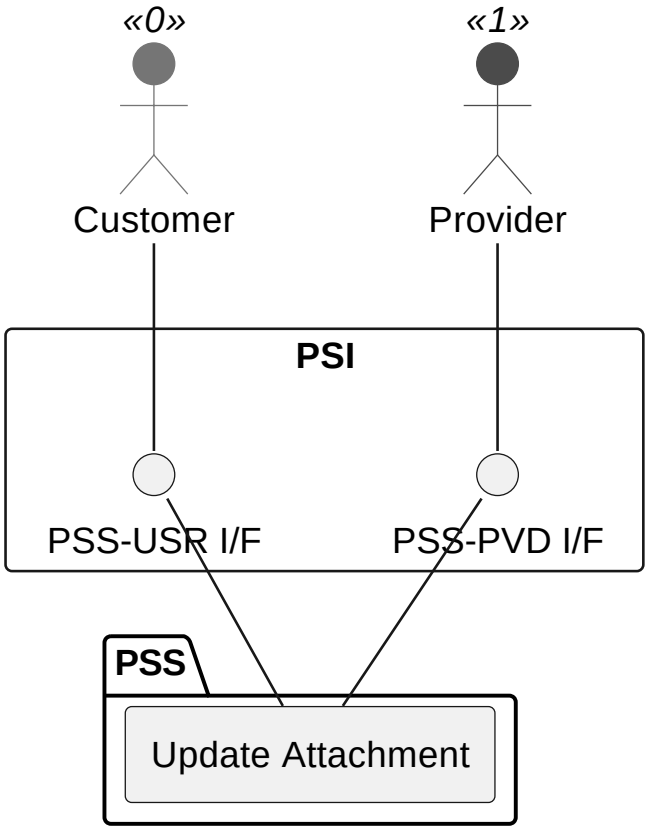

Figure 5.20: TOD-01-03-07: Update Attachment

Prerequisites

The attachment to be updated exists in the PSS.

Main operation

This operation updates the properties of an existing attachment in a PSS via a standard interface specification.

When updating the *content*, the system implementing the interface should ensure that the file is also updated in the physical location or file storage system.

REST Endpoints

- PATCH /documentManagement/v1/document/{documentId}/attachment/{attachmentId}

Post Conditions

The attachment has been successfully updated in the PSS.

Applicable Requirements

- PSI-01-03-07-01
- PSI-01-03-07-02

eTOM Reference

None

5.1.3.8 TOD-01-03-08-Remove_Attachment

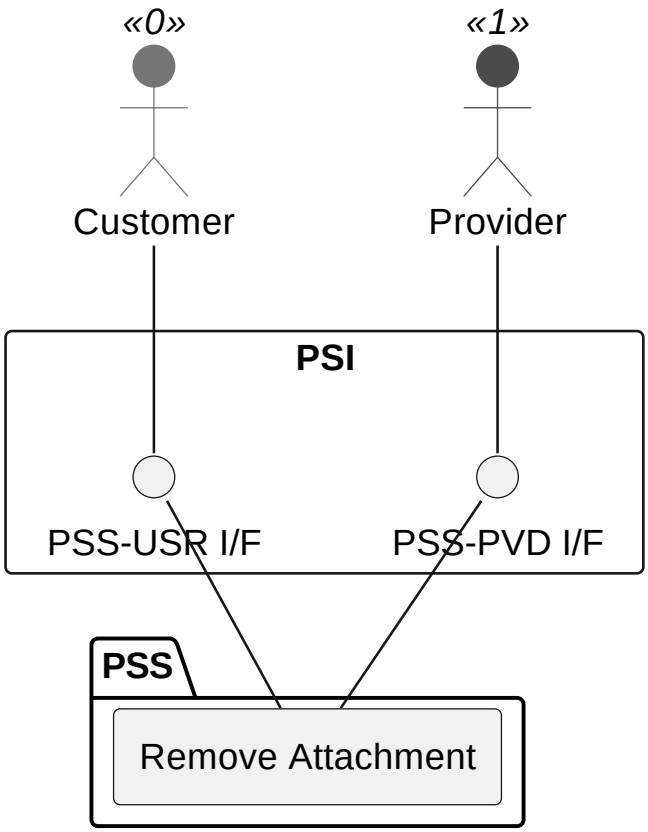

Figure 5.21: TOD-01-03-08: Remove Attachment

Prerequisites

The attachment to be removed exists in the PSS.

Main operation

Removes an attachment either by deleting it or indicating it is no longer valid, via a standard interface specification.

The system implementing the interface should ensure that the attachment is removed also from the physical location or the file storage system.

REST Endpoints

- DELETE /documentManagement/v1/document/{documentId}/attachment/{attachmentId}

Post Conditions

The attachment has been deleted or indicated it is no longer valid in the PSS.

Applicable Requirements

- PSI-01-03-08-01
- PSI-01-03-08-02

eTOM Reference

None

5.1.3.9 TOD-01-03-09-View_Attachment

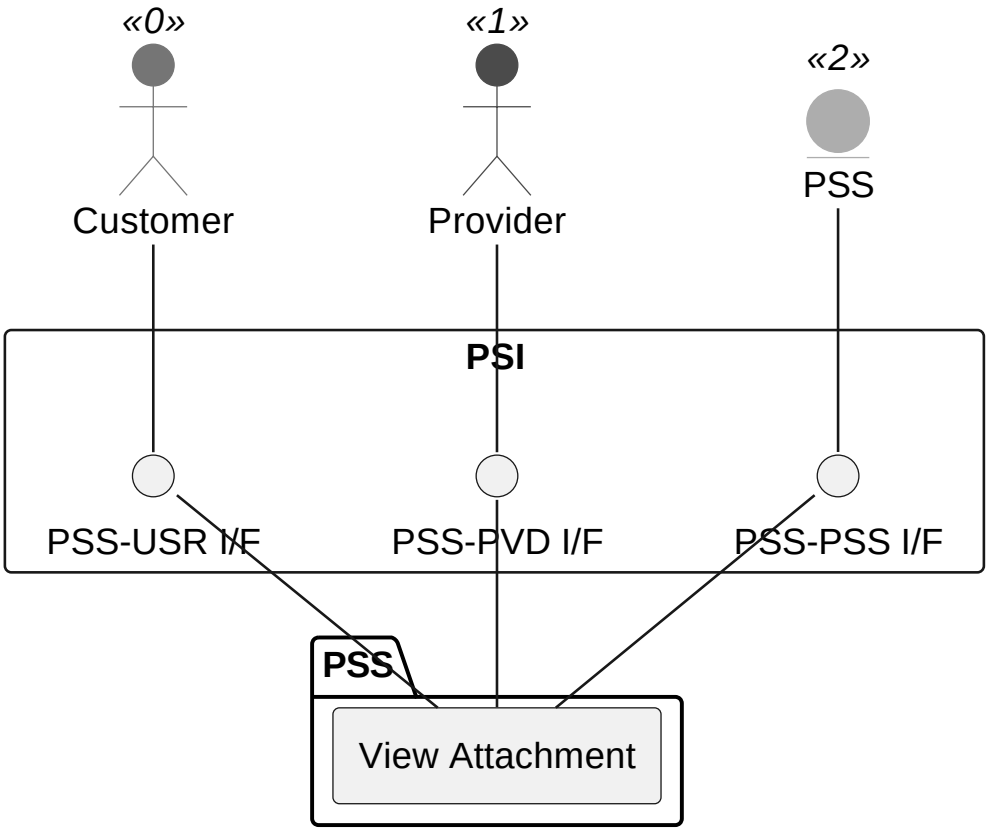

Figure 5.22: TOD-01-03-09: View Attachment

Prerequisites

The attachment object to be viewed exists in the PSS.

Main operation

The operation retrieves an attachment from the PSS in JSON representation.

REST Endpoints

- GET /documentManagement/v1/document/{documentId}/attachment/{attachmentId}

Post Conditions

The attachment is successfully retrieved to be viewed from the PSS.

Applicable Requirements

- PSI-01-03-09-01
- PSI-01-03-09-02

eTOM Reference

None

5.1.3.10 TOD-01-03-10-View_All_Attachments

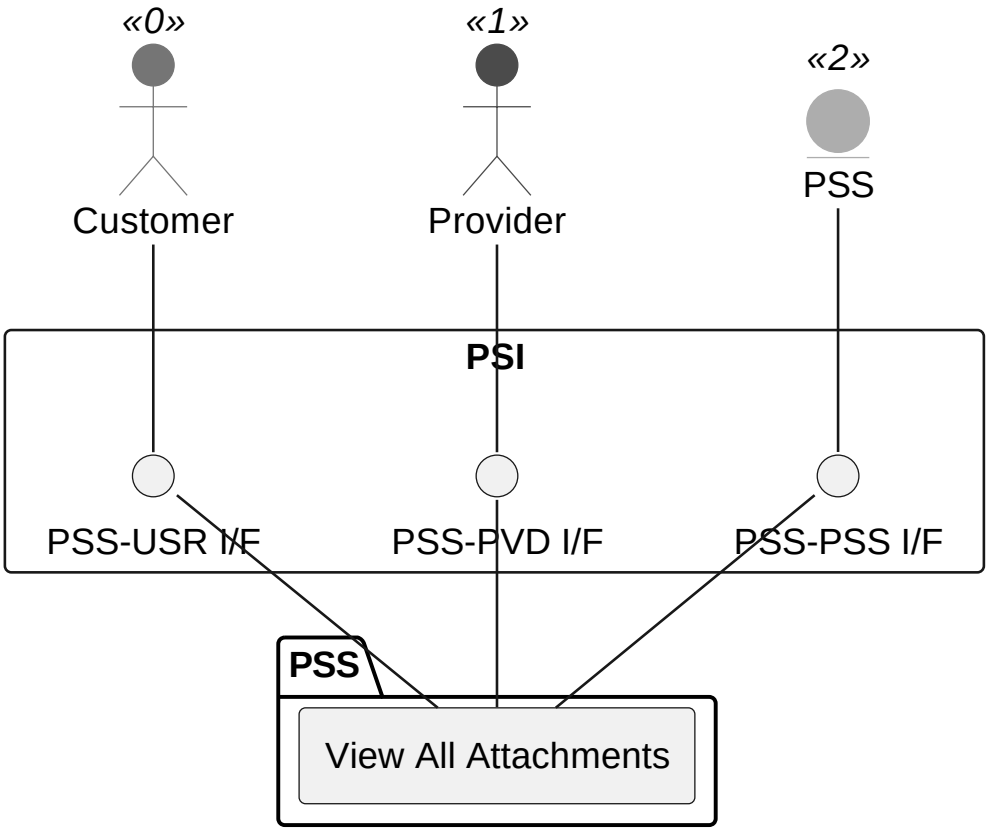

Figure 5.23: TOD-01-03-10: View All Attachments

Prerequisites

Attachments exist in the PSS.

Main operation

Gets a list of all attachments that are applicable to the user requesting them.

REST Endpoints

- GET /documentManagement/v1/document/{documentId}/attachment

Post Conditions

The list of all available attachments is successfully returned.

Applicable Requirements

- PSI-01-03-10-01

eTOM Reference

None

5.1.3.11 TOD-01-03-11-Fetch_Attachment_Content

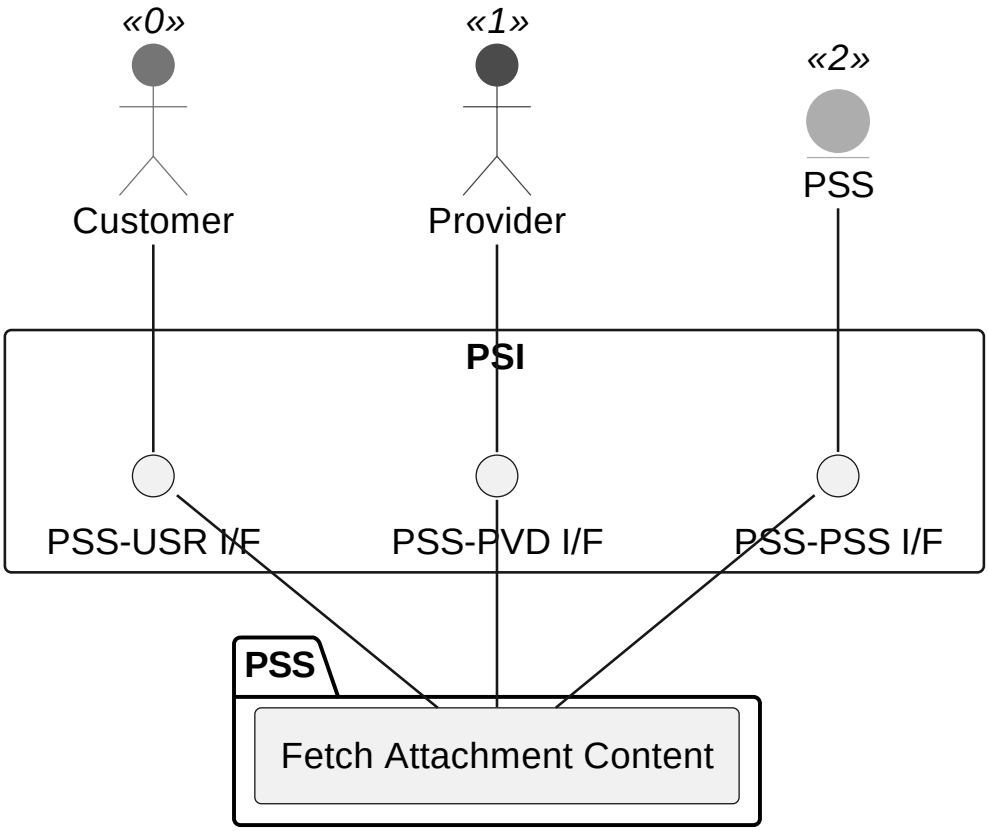

Figure 5.24: TOD-01-03-11: Fetch Attachment Content

Prerequisites

The attachment whose binary content is to be fetched exists in the PSS.

Main operation

The operation fetches the actual binary content of the attachment from the PSS for direct preview.

REST Endpoints

- GET /documentManagement/v1/document/{documentId}/attachment/{attachmentId}/content

Post Conditions

The binary content of the attachment is successfully fetched for preview.

Applicable Requirements

- PSI-01-03-11-01
- PSI-01-03-11-02

eTOM Reference

None

5.1.3.12 TOD-01-03-12-Update_Attachment_Content

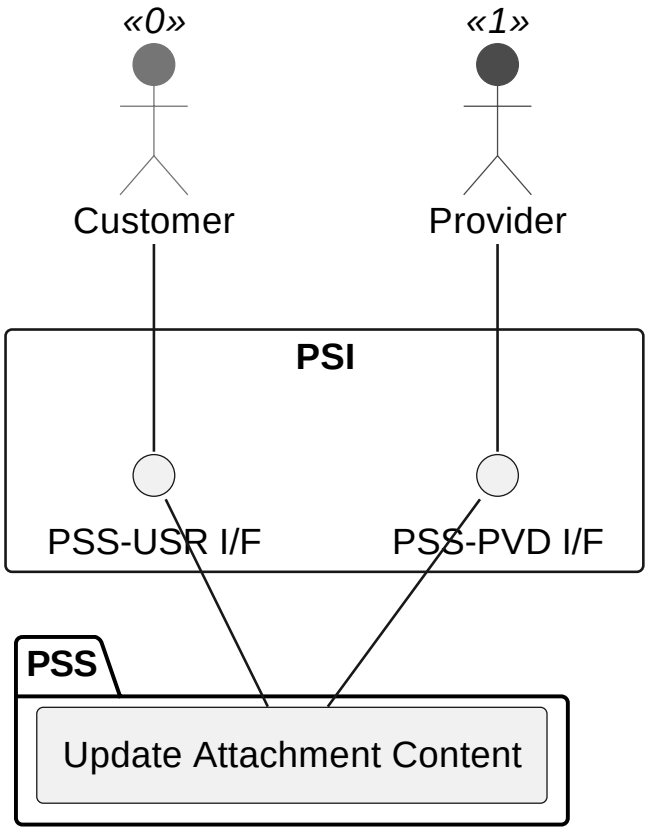

Figure 5.25: TOD-01-03-12: Update Attachment Content

Prerequisites

The attachment whose binary content is to be updated exists in the PSS.

Main operation

This operation updates the actual binary content of an existing attachment in the PSS via a standard interface specification. It means that the system implementing the interface updates the file in the physical location or file storage system.

REST Endpoints

- PUT /documentManagement/v1/document/{documentId}/attachment/{attachmentId}/content

Post Conditions

The binary content of the attachment has been updated.

Applicable Requirements

- PSI-01-03-12-01
- PSI-01-03-12-02

eTOM Reference

None

5.1.4 TOD-01-04-Trouble_Ticket_Management

The Trouble Ticket Management task is responsible for tracking incident reports, complaints and other requests of customers and providers. They can be processed either by a PSS helpdesk operator if they concern the functionality of the PSS itself, or by the provider if they affect a (Gov)SatCom service. Most likely, the actual implementation is outsourced to an existing ticket system or the CRM.

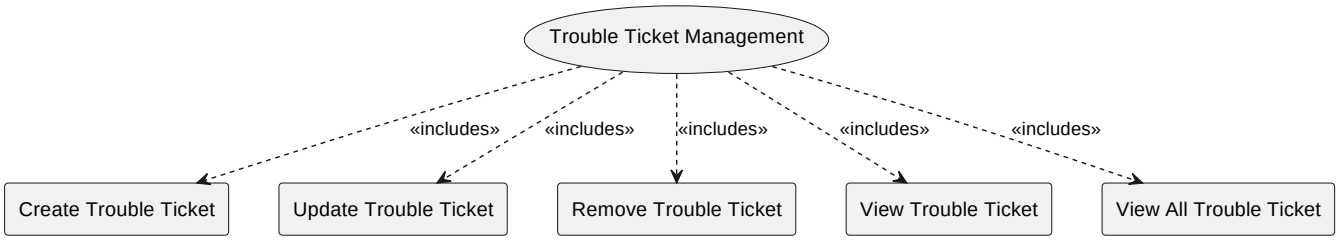

Figure 5.26: TOD-01-04: Trouble Ticket Management

	Customer	Provider	Other PSS	Governance
Create Trouble Ticket	✓	✓		
Update Trouble Ticket	✓	✓		✓
Remove Trouble Ticket	✓	✓		
View Trouble Ticket	✓	✓		✓
View All Trouble Ticket	✓	✓		✓

Table 5.4: Trouble Ticket Management Matrix.

eTOM Reference

The task is based on the 1.4.6 process identifier from the eTOM.

5.1.4.1 TOD-01-04-01-Create_Trouble_Ticket

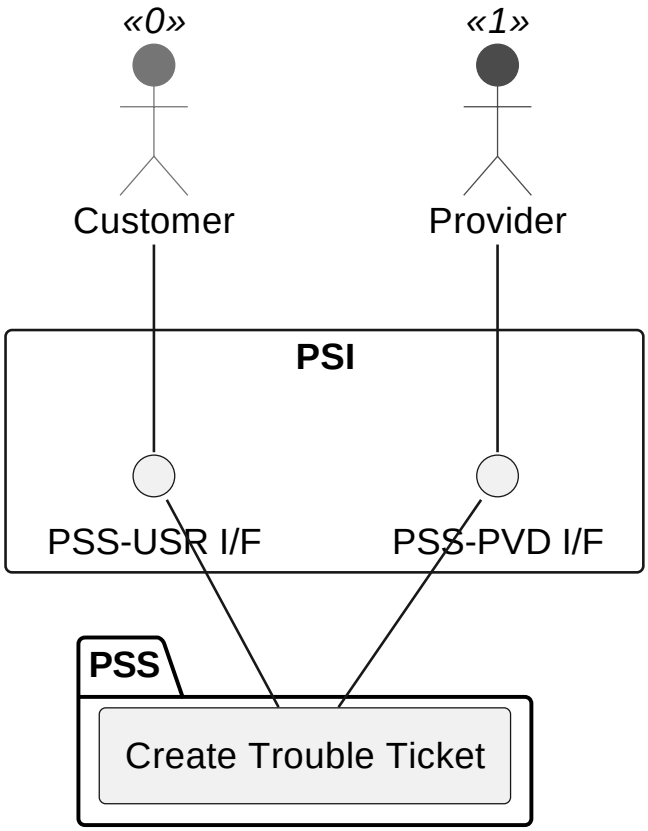

Figure 5.27: TOD-01-04-01: Create Trouble Ticket

Prerequisites

The trouble ticket does not exist in the PSS datastore.

Main operation

Creates a new trouble ticket via a standard interface specification.

Some properties of a trouble ticket are:

- *name* - Short title for the trouble ticket
- *description* - Description of the trouble ticket, e.g. details about the incident.
- *priority* - The priority of the ticket. Can be set by the customer and changed later.
- *requestedResolutionDate* - Optional attribute to complement the priority.
- *ticketType* - Business type of the trouble ticket e.g. incident, complaint, request.
- *attachment* - File(s) attached to the trouble ticket. e.g. picture of broken device.
- *relatedEntity* - The entity against which the ticket is associated, e.g. a service instance.

REST Endpoints

- POST /troubleTicket/v1/troubleTicket

Post Conditions

The trouble ticket is successfully created in the PSS datastore.

Applicable Requirements

- PSI-01-04-01-01

eTOM Reference

The operation is based on 1.4.6.1 process identifier from the eTOM.

5.1.4.2 TOD-01-04-02-Update_Trouble_Ticket

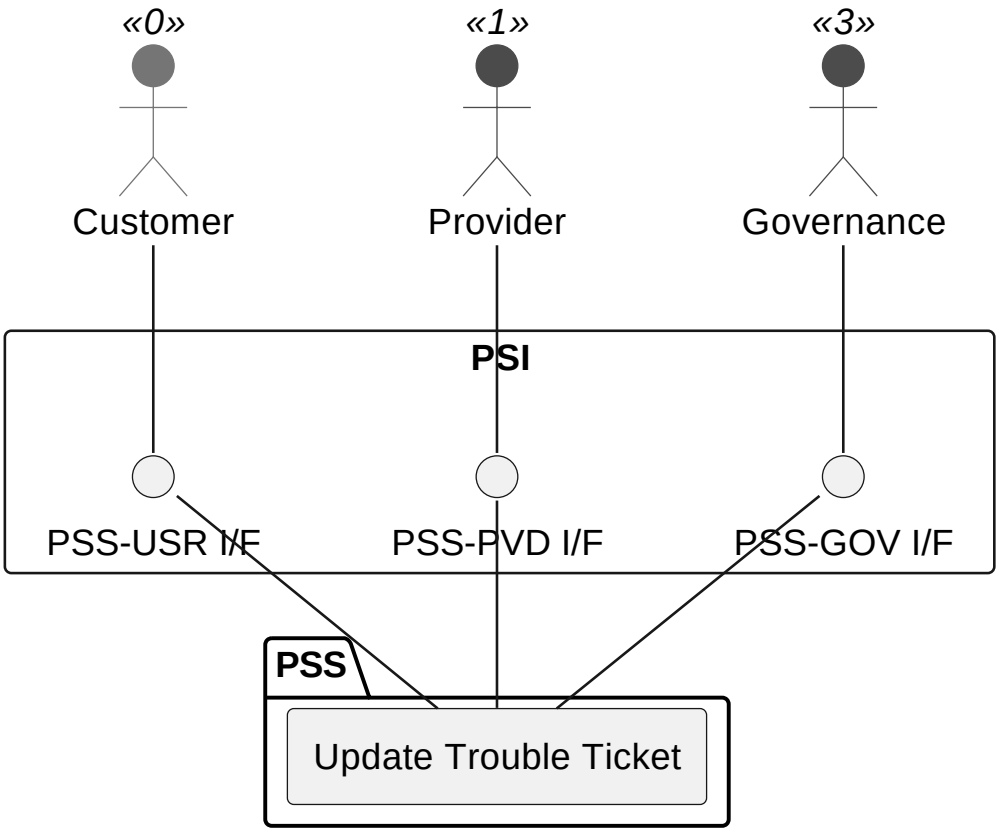

Figure 5.28: TOD-01-04-02: Update Trouble Ticket

Prerequisites

The trouble ticket exists in the PSS datastore. The following properties are available additional to the creation:

- *expectedResolutionDate* - The expected resolution date by the provider.
- *resolutionDate* - The actual resolution date, set when closing the ticket.
- *status* - The current status of the ticket (e.g. pending, in progress, resolved).

- *statusChangeReason* - To be set when the status is changed. Will be stored in the status history by the server.
- *note* - Additional comments by the customer or provider.

Main operation

Updates an existing trouble ticket via a standard interface specification.

REST Endpoints

- PATCH /troubleTicket/v1/troubleTicket/{id}

Post Conditions

The trouble ticket is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-01-04-02-01
- PSI-01-04-02-02

eTOM Reference

The operation is based on 1.4.6.4 and 1.4.6.6 process identifiers from the eTOM.

5.1.4.3 TOD-01-04-03-Remove_Trouble_Ticket

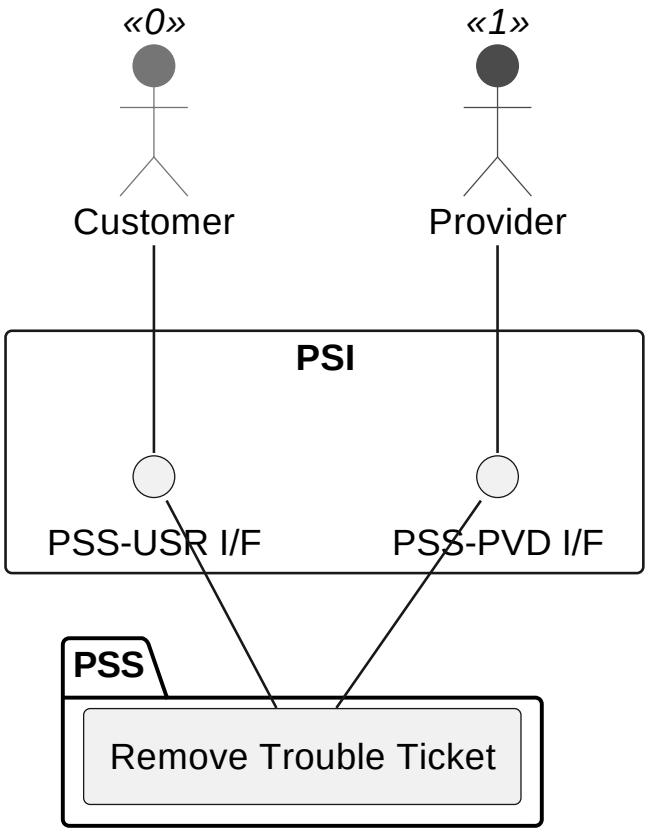

Figure 5.29: TOD-01-04-03: Remove Trouble Ticket

Prerequisites

The trouble ticket exists in the PSS datastore.

Main operation

Removes a trouble ticket either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /troubleTicket/v1/troubleTicket/{id}

Post Conditions

The trouble ticket is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-01-04-03-01
- PSI-01-04-03-02

eTOM Reference

The operation is based on 1.4.6.4 process identifier from the eTOM.

5.1.4.4 TOD-01-04-04-View_Trouble_Ticket

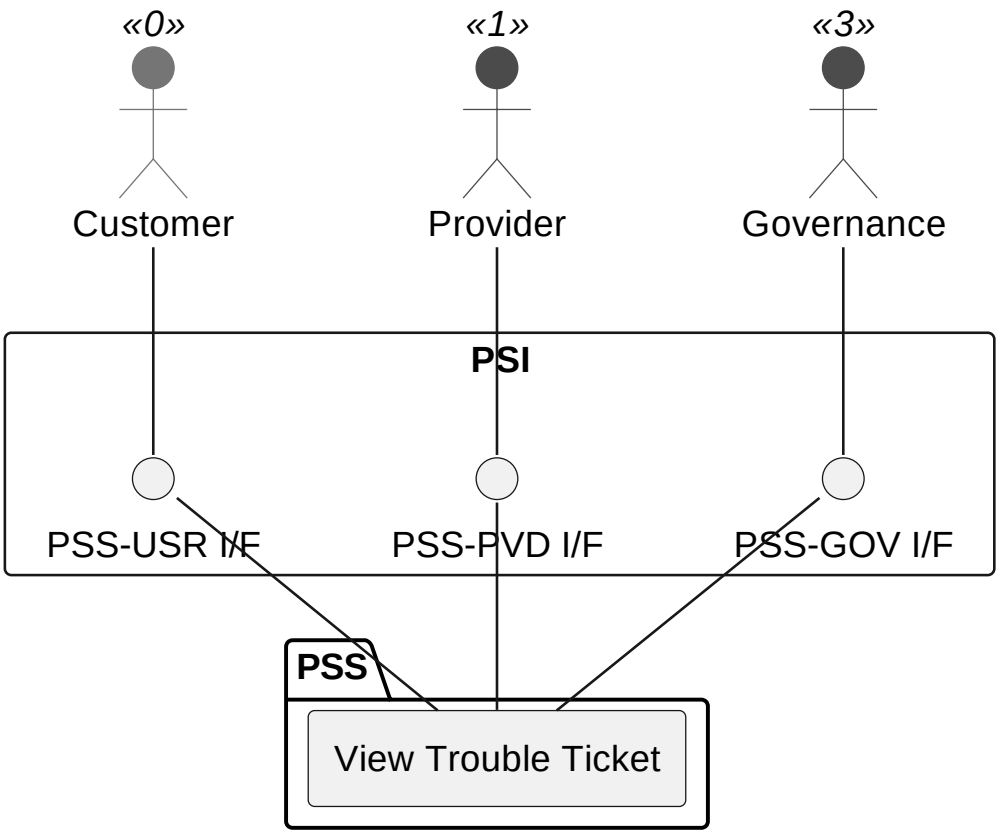

Figure 5.30: TOD-01-04-04: View Trouble Ticket

Prerequisites

The trouble ticket exists in the PSS datastore.

Main operation

Gets a trouble ticket with a specific identifier via a standard interface specification.

REST Endpoints

- GET /troubleTicket/v1/troubleTicket/{id}

Post Conditions

The trouble ticket is successfully returned to be viewed.

Applicable Requirements

- PSI-01-04-04-01
- PSI-01-04-04-02

eTOM Reference

The operation is based on 1.4.6.4 process identifier from the eTOM.

5.1.4.5 TOD-01-04-05-View_All_Trouble_Tickets

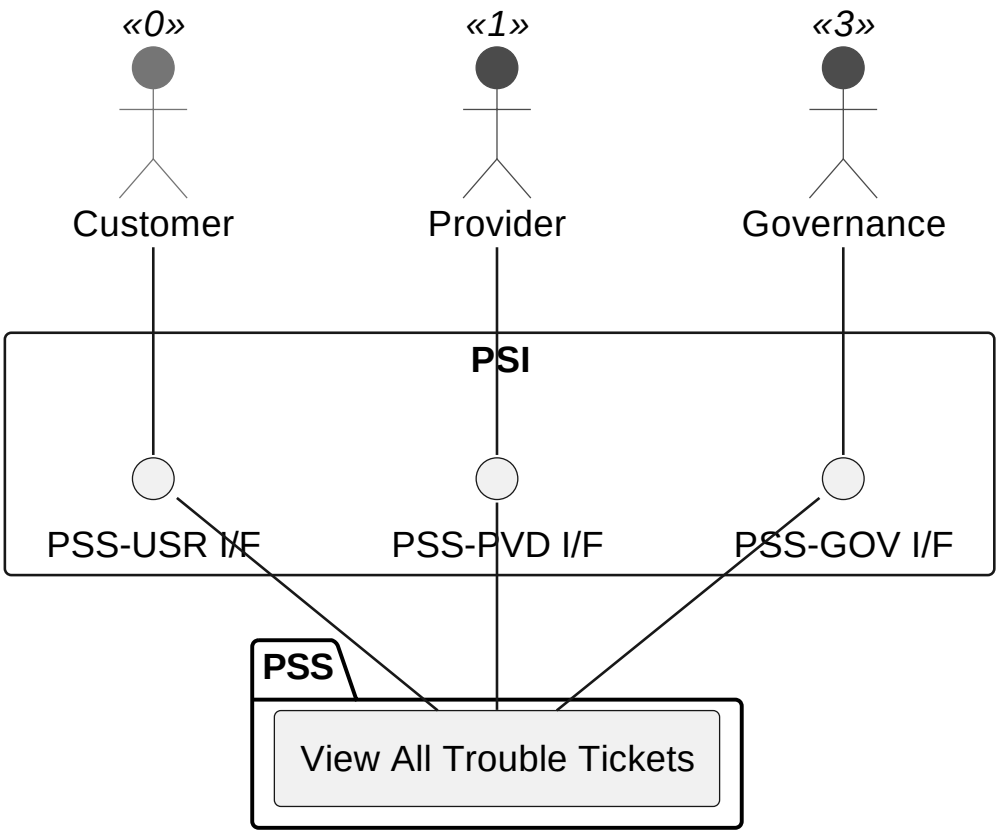

Figure 5.31: TOD-01-04-05: View All Trouble Tickets

Prerequisites

Trouble Tickets exist in the PSS datastore.

Main operation

Gets all trouble tickets visible to the caller via a standard interface specification. These can be filtered at least by type and status.

REST Endpoints

- GET /troubleTicket/v1/troubleTicket

Post Conditions

All visible trouble tickets are successfully returned to be viewed.

Applicable Requirements

- PSI-01-04-05-01

eTOM Reference

The operation is based on 1.4.6.4 process identifier from the eTOM.

5.2 TOD-02-Product-Publishing

This category consists of tasks and operations related to publishing a product. This involves the management of resources, services and products towards a final product offering with a price that could be ordered by a customer.

5.2.1 TOD-02-01-Resource_Catalog_Management

The Resource Catalog Management task takes care of the maintenance of resource specifications available in the PSS, brought in by providers.

A provider wants to utilize a PSS to offer their resources to the users of the PSS. The resources implement a resource specification (describing general characteristics of the resource), and they can be **physical** (e.g. antennas, BUCs and other hardware, satellites, etc.), **logical** (e.g. IP addresses, software) or **compound** (e.g. router consists of different cards/ports and runs software). These are inputs to the PSS which are further constructed/marketed/brokered as products.

Therefore, a provider needs to be able to register(create) resource specifications to the PSS, modify, remove or view them. Another PSS needs to be able to view the resources specifications as well.

Additionally, a customer needs to be able to declare its own resources (e.g. terminal/teleport) into the PSS and later use them as part of a customer inquiry. Details of the matchmaking, such as resource sharing with other customers, are up to the respective PSS implementation.

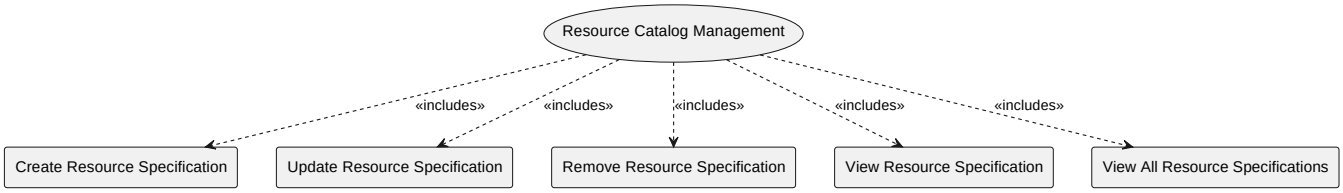

Figure 5.32: TOD-02-01: Resource Catalog Management

	Customer	Provider	Other PSS	Governance
Create Resource Specification	✓	✓		
Update Resource Specification	✓	✓		
Remove Resource Specification	✓	✓		
View Resource Specification	✓	✓	✓	✓
View All Resource Specifications	✓	✓	✓	✓

Table 5.5: Resource Catalog Management Matrix.

eTOM Reference

The task is based on the 1.5.17 and 1.5.3 process identifiers from the eTOM.

5.2.1.1 TOD-02-01-01-Create_Resource_Specification

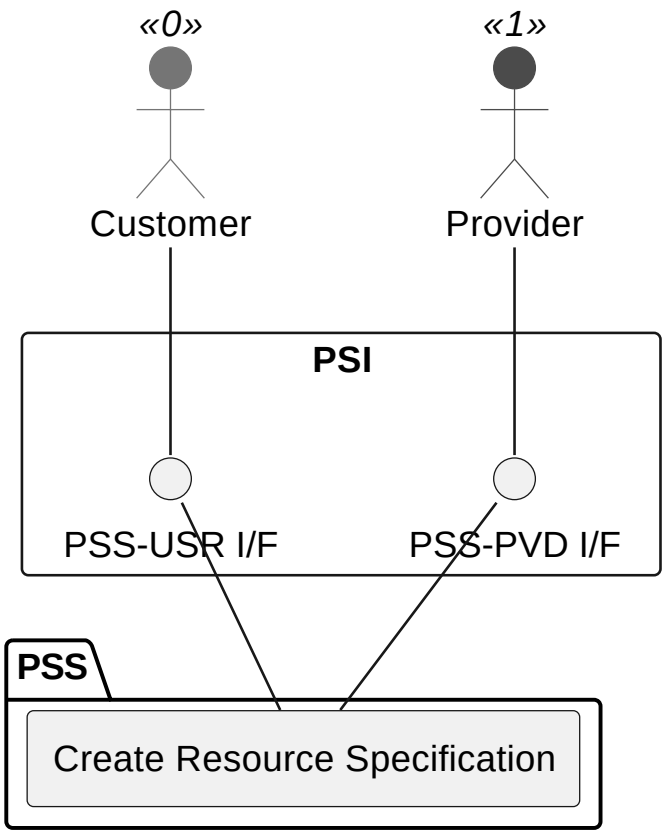

Figure 5.33: TOD-02-01-01: Create Resource Specification

Prerequisites

The resource specification does not exist in the PSS datastore.

Main operation

Creates a new resource specification with its characteristics via a standard interface specification.

Note: It is possible to create a resource specification which will be available in the future by setting the *validFor* property with a future time reference.

Some properties of a resource specification are:

- *name* - Short name of the target resource
- *description* - Description of the target resource
- *category* - Category (resource type) of the target resource like terminals, bandwidth, etc.
- *targetResourceSchema* - Name and reference to the JSON Schema defining the type of resource described by this specification.
- *resourceSpecCharacteristic* - List of characteristics i.e. technical specifications of the resource such as frequency band, Tx/Rx frequency, etc. Beam footprints are described as a characteristic of a *GeoJSON FeatureCollection* that contains one or more beam geometries in the GeoJSON data format (e.g. as a polygon) with corresponding properties, like the EIRP value.
- *relatedParty* - Usually reference to the provider that offers the resource
- *lifecycleStatus* - Current lifecycle status of the resource specification (e.g. active, draft, etc.)
- *validFor* - Time period of validity of the resource specification

Before creating a new resource specification, the provider or customer can request available resource templates from the PSS via the TOD-04-01-05-View_All_Resource_Templates operation. The templates are prepared by the governance of the PSS, and they contain default values for the characteristics of a resource specification. For instance, if the provider wants to register a modem to the PSS, they can request available resource templates for modems. Once they identify the target template for the resource specification they want to create, they need to replace the default values with specific ones and invoke the endpoint.

The templates are generic and therefore can be used by any provider. This way, providers are given the flexibility to reuse from the template what is relevant for their resource specification, but also enhance it to fully match the characteristics of their resource. This significantly shortens the time they require to prepare them for registration to the resource catalog of the PSS.

REST Endpoints

- POST /resourceCatalog/v1/resourceSpecification

Post Conditions

The resource specification is successfully created in the PSS datastore.

Applicable Requirements

- PSI-02-01-01-01
- PSI-02-01-01-02
- PSI-02-01-01-03
- PSI-02-01-01-04

eTOM Reference

The operation is based on 1.5.17.1 and 1.5.19.2 process identifiers from the eTOM.

5.2.1.2 TOD-02-01-02-Update_Resource_Specification

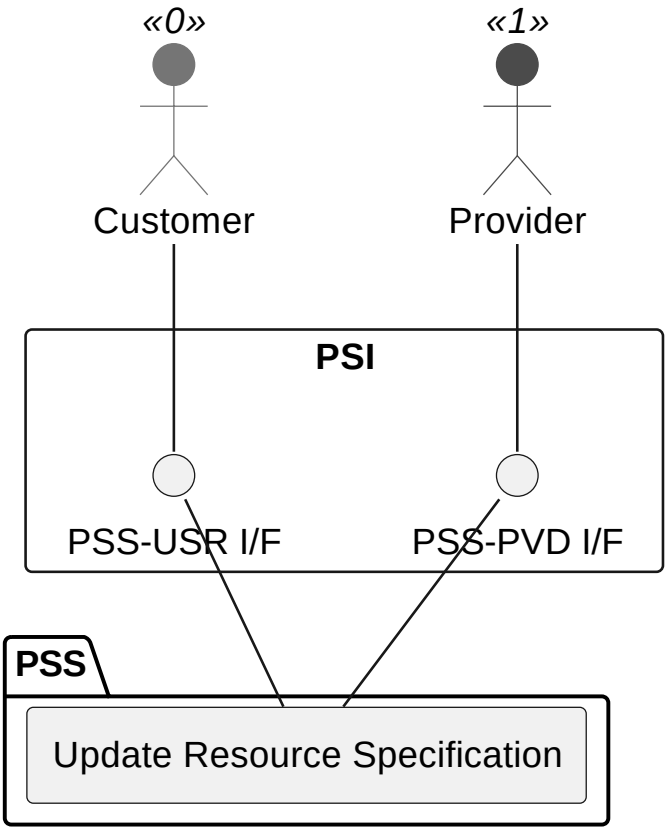

Figure 5.34: TOD-02-01-02: Update Resource Specification

Prerequisites

The resource specification exists in the PSS datastore.

Main operation

Updates an existing resource specification via a standard interface specification. A PSS might reject the update of essential characteristics like the frequency band. In that case, the client shall set an expiration date for the current specification and create a new one that is valid thereafter.

REST Endpoints

- PATCH /resourceCatalog/v1/resourceSpecification/{id}

Post Conditions

The resource specification is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-02-01-02-01
- PSI-02-01-02-02

eTOM Reference

The operation is based on 1.5.17.1 and 1.5.19.2 process identifiers from the eTOM.

5.2.1.3 TOD-02-01-03-Remove_Resource_Specification

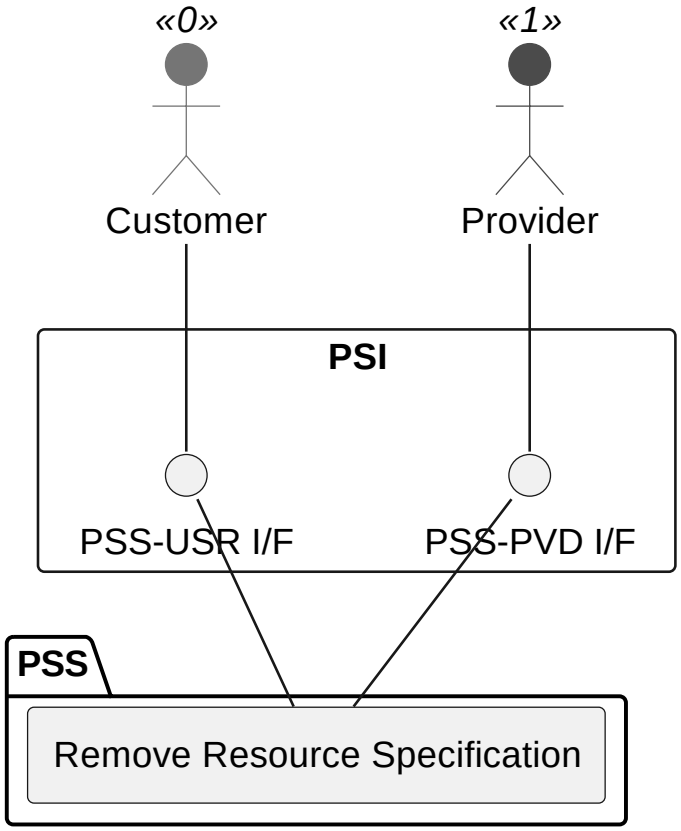

Figure 5.35: TOD-02-01-03: Remove Resource Specification

Prerequisites

The resource specification exists in the PSS datastore.

Main operation

Removes a resource specification either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /resourceCatalog/v1/resourceSpecification/{id}

Post Conditions

The resource specification is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-02-01-03-01
- PSI-02-01-03-02

eTOM Reference

The operation is based on 1.5.17.1 and 1.5.19.2 process identifiers from the eTOM.

5.2.1.4 TOD-02-01-04-View_Resource_Specification

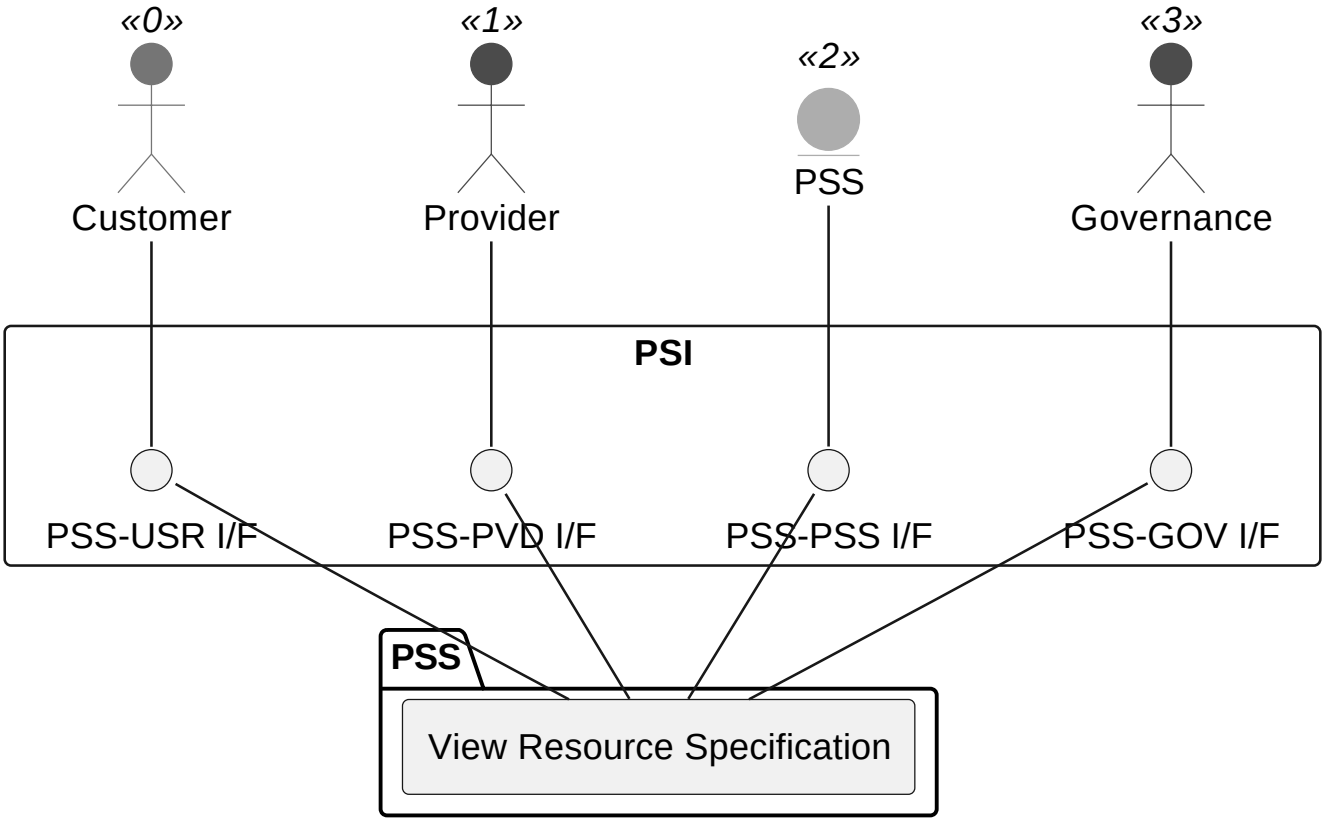

Figure 5.36: TOD-02-01-04: View Resource Specification

Prerequisites

The resource specification exists in the PSS datastore.

Main operation

Gets a resource specification of the provider with a specific identifier via a standard interface specification. Customers can view their own declared resource specification.

REST Endpoints

- GET /resourceCatalog/v1/resourceSpecification/{id}

Post Conditions

The resource specification is successfully returned to be viewed.

Applicable Requirements

- PSI-02-01-04-01
- PSI-02-01-04-02

eTOM Reference

The operation is based on 1.5.17.1 and 1.5.19.2 process identifiers from the eTOM.

5.2.1.5 TOD-02-01-05-View_All_Resource_Specifications

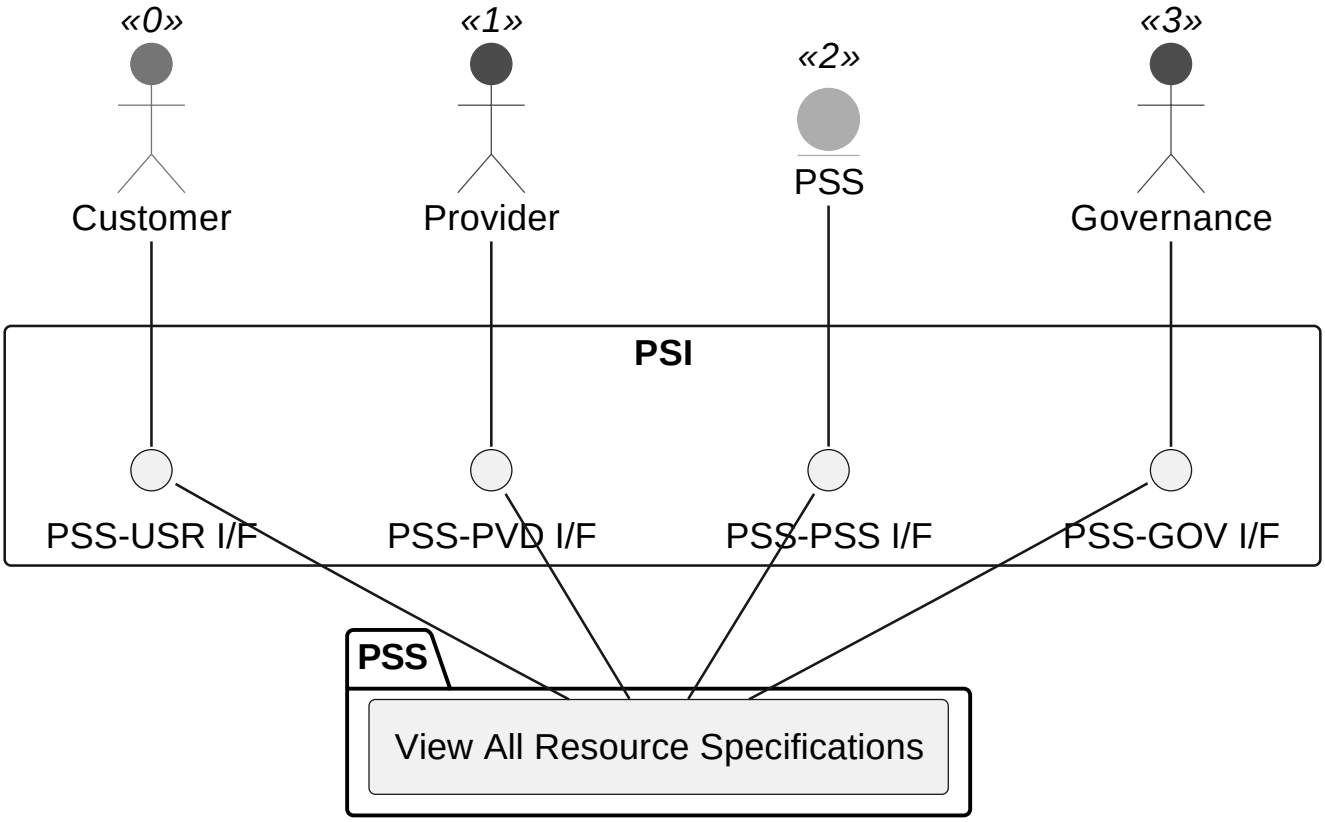

Figure 5.37: TOD-02-01-05: View All Resource Specifications

Prerequisites

Resource specifications of the provider exist in the PSS datastore.

Main operation

Gets all resource specifications of the provider via a standard interface specification. These can be filtered at least by resource type. Customers can view their own declared resource specifications.

REST Endpoints

- GET /resourceCatalog/v1/resourceSpecification

Post Conditions

All resource specifications of the provider are successfully returned to be viewed.

Applicable Requirements

- PSI-02-01-05-01
- PSI-02-01-05-02

eTOM Reference

The operation is based on 1.5.17.1 and 1.5.19.2 process identifiers from the eTOM.

5.2.2 TOD-02-02-Service_Catalog_Management

The Service Catalog Management task takes care of the maintenance of service specifications available in the PSS, brought in by providers.

A provider wants to utilize a PSS to offer their services to the users of the PSS. The services implement a service specification (describing general characteristics of the service), and they represent a communication service such as internet access, telephony, site-to-site IP-Trunk, etc. which require utilization of different resources. These are inputs to the PSS which are further constructed/marketed/brokered as products.

Therefore, a provider needs to be able to register(create) service specifications to the PSS, modify, remove or view them. Another PSS needs to be able to view the disclosed service specifications as well.

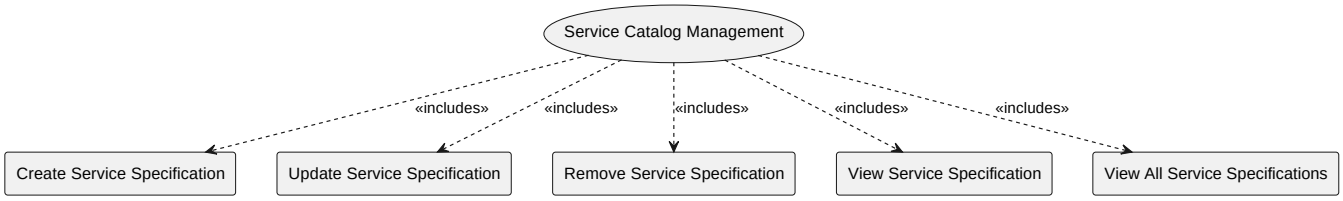

Figure 5.38: TOD-02-02: Service Catalog Management

	Customer	Provider	Other PSS	Governance
Create Service Specification		✓		
Update Service Specification		✓		
Remove Service Specification		✓		
View Service Specification	✓	✓	✓	✓
View All Service Specifications	✓	✓	✓	✓

Table 5.6: Service Catalog Management Matrix.

eTOM Reference

The task is based on the 1.4.15 and 1.4.3 process identifiers from the eTOM.

5.2.2.1 TOD-02-02-01-Create_Service_Specification

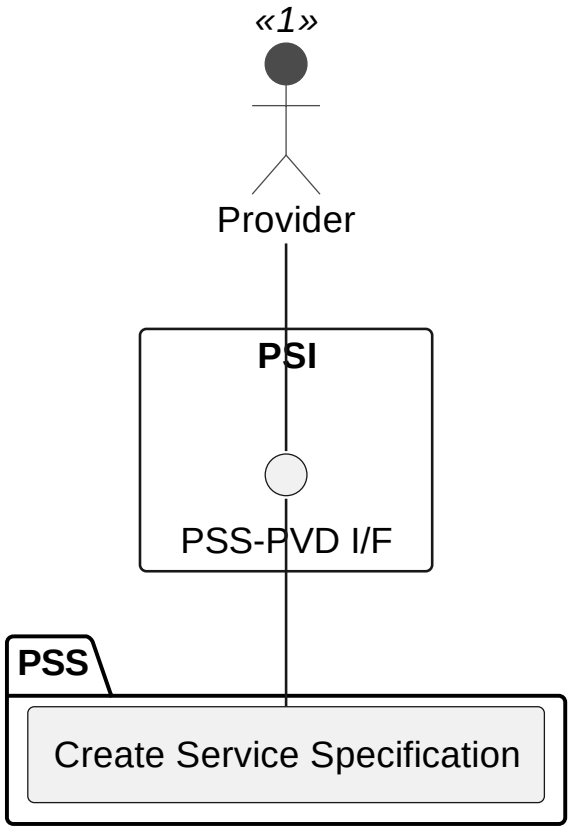

Figure 5.39: TOD-02-02-01: Create Service Specification

Prerequisites

The service specification does not exist in the PSS datastore.

Main operation

Creates a new service specification with its characteristics and references to resource specifications via a standard interface specification.

> Note: It is possible to create a service specification which will be available in the future by setting the *validFor* property with a future time reference.

Some properties of a service specification are:

- *name* - Short name of the target service
- *description* - Description of the target service
- *category* - Category (service type) of the target service like internet access, telephony, IP-Trunk, etc.
- *resourceSpecification* - List of resources that are required to realise the target service
- *targetEntitySchema* - Name and reference to the JSON Schema defining the type of service described by this specification.

- *specCharacteristic* - List of specification characteristics of the target service such as forwardCIR, returnCIR, etc. The footprint of a service is described as a characteristic of type *GeoJSON FeatureCollection*, which contains one or more GeoJSON entries (e.g. as a polygon) with the option for assigned properties, such as the EIRP value.
- *relatedParty* - Usually reference to the provider that offers the service
- *lifecycleStatus* - Current lifecycle status of the service specification (e.g. active, draft, etc.)
- *validFor* - Time period of validity of the service specification

Before creating a new service specification, a provider can request available service templates from the PSS via the TOD-04-02-05-View_All_Service_Templates operation. The templates are prepared by the governance of the PSS, and they contain default values for the characteristics of a service specification. For instance, if the provider wants to register a service specification for internet access to the PSS, they can request available service templates for internet access, replace the default values with specific ones and invoke the endpoint for creating a service specification.

The templates are generic and therefore can be used by any provider. This way, providers are given the flexibility to reuse from the template what is relevant for their service specification, but also enhance it to fully match the characteristics of their service. This shortens the time a provider requires for registration of their service specifications to the service catalog of the PSS while still allowing them to showcase their unique selling points.

REST Endpoints

- POST /serviceCatalog/v1/serviceSpecification

Post Conditions

The service specification is successfully created in the PSS datastore.

Applicable Requirements

- PSI-02-02-01-01
- PSI-02-02-01-02
- PSI-02-02-01-03
- PSI-02-02-01-04

eTOM Reference

The operation is based on 1.4.15.1 and 1.4.19.2 process identifiers from the eTOM.

5.2.2.2 TOD-02-02-02-Update_Service_Specification

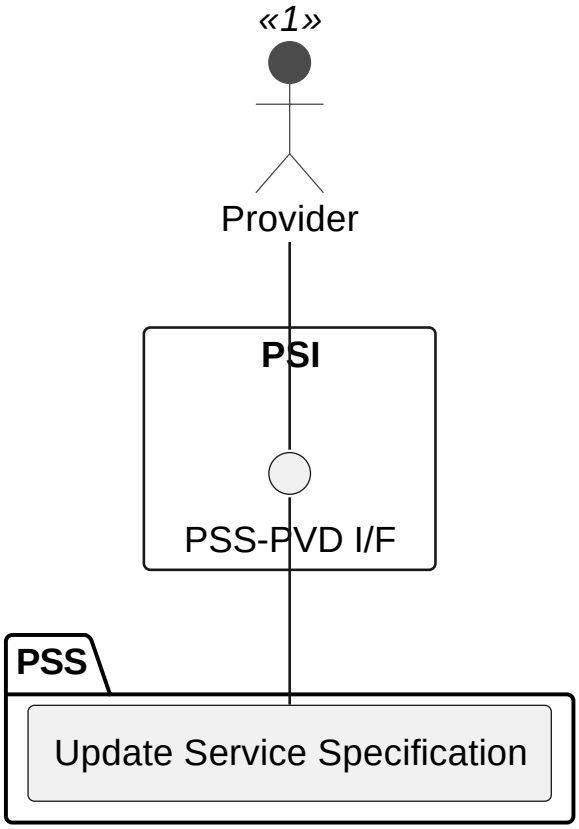

Figure 5.40: TOD-02-02-02: Update Service Specification

Prerequisites

The service specification exists in the PSS datastore. A PSS might reject the update of essential characteristics like the frequency band. In that case, the client shall set an expiration date for the current specification and create a new one that is valid thereafter.

Main operation

Updates an existing service specification via a standard interface specification.

REST Endpoints

- PATCH /serviceCatalog/v1/serviceSpecification/{id}

Post Conditions

The service specification is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-02-02-02-01
- PSI-02-02-02-02

eTOM Reference

The operation is based on 1.4.15.1 and 1.4.19.2 process identifiers from the eTOM.

5.2.2.3 TOD-02-02-03-Remove_Service_Specification

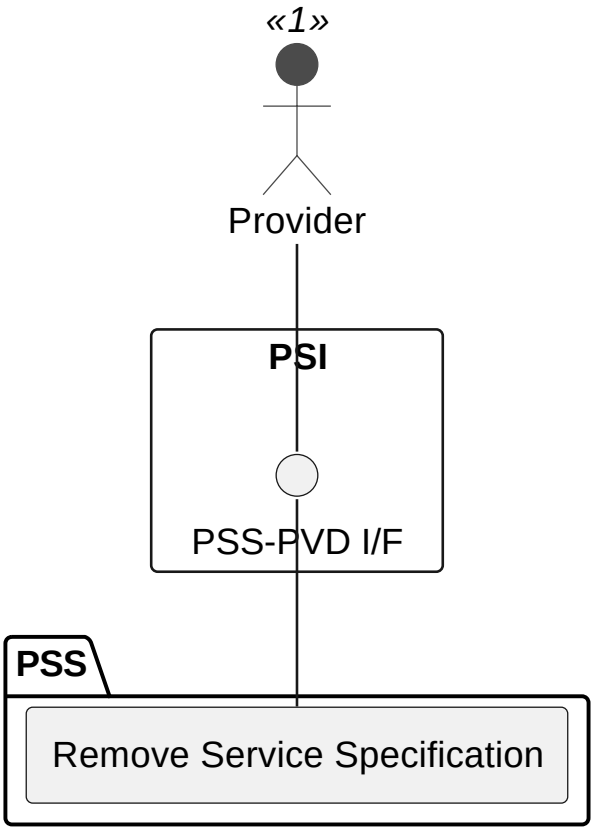

Figure 5.41: TOD-02-02-03: Remove Service Specification

Prerequisites

The service specification exists in the PSS datastore.

Main operation

Removes a service specification either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /serviceCatalog/v1/serviceSpecification/{id}

Post Conditions

The service specification is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-02-02-03-01
- PSI-02-02-03-02

eTOM Reference

The operation is based on 1.4.15.1 and 1.4.19.2 process identifiers from the eTOM.

5.2.2.4 TOD-02-02-04-View_Service_Specification

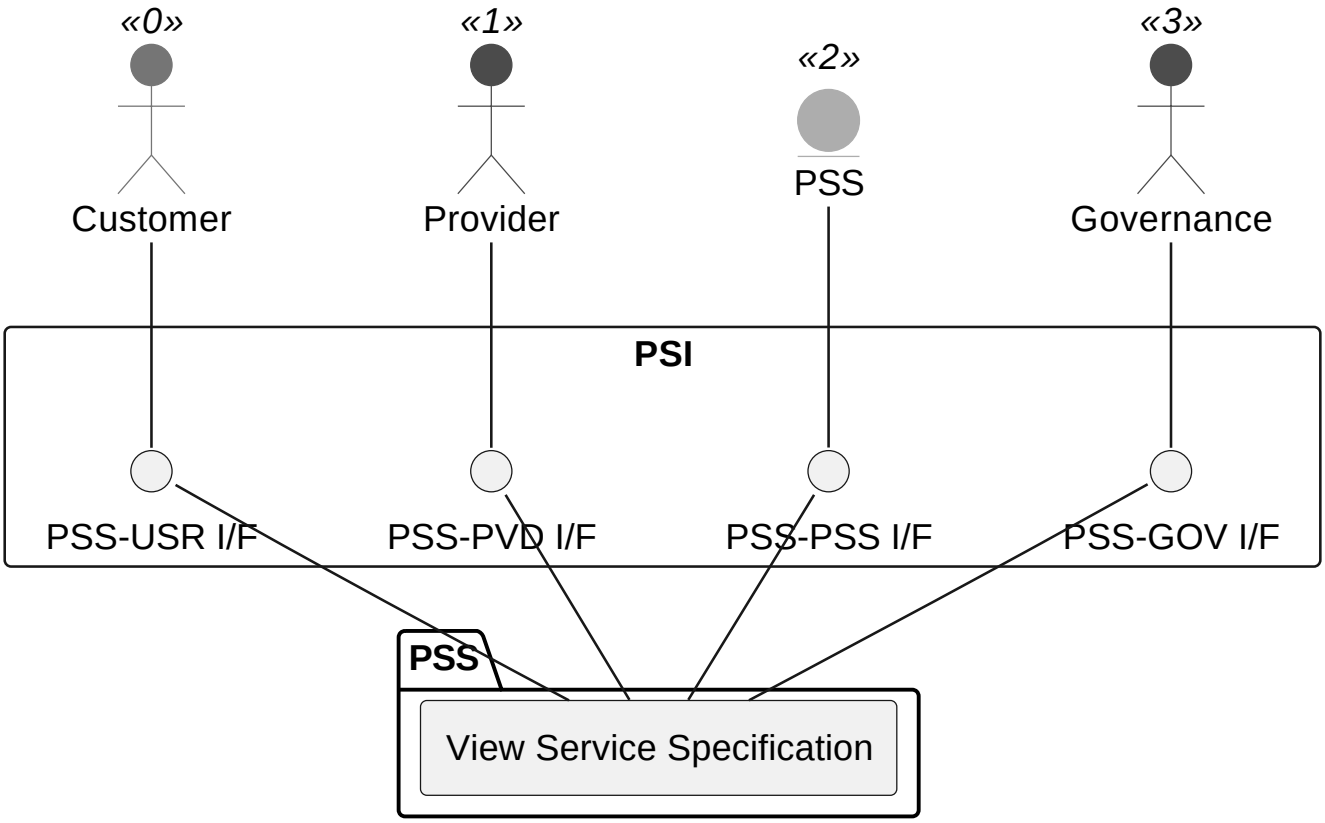

Figure 5.42: TOD-02-02-04: View Service Specification

Prerequisites

The service specification exists in the PSS datastore.

Main operation

Gets a service specification of the provider with a specific identifier via a standard interface specification.

REST Endpoints

- GET /serviceCatalog/v1/serviceSpecification/{id}

Post Conditions

The service specification is successfully returned to be viewed.

Applicable Requirements

- PSI-02-02-04-01
- PSI-02-02-04-02

eTOM Reference

The operation is based on 1.4.15.1 and 1.4.19.2 process identifiers from the eTOM.

5.2.2.5 TOD-02-02-05-View_All_Service_Specifications

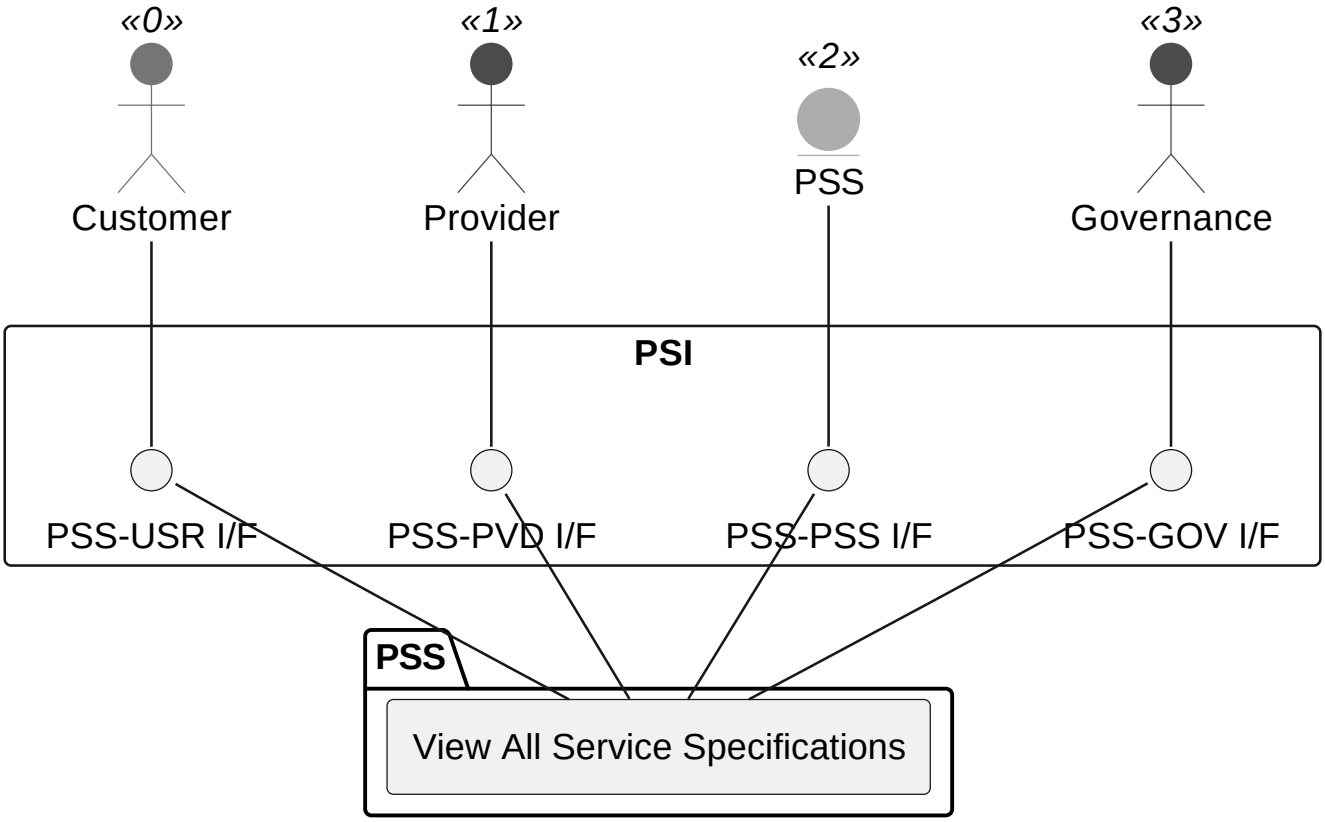

Figure 5.43: TOD-02-02-05: View All Service Specifications

Prerequisites

Service specifications of the provider exist in the PSS datastore.

Main operation

Gets all service specifications of the provider via a standard interface specification. These can be filtered at least by service type.

REST Endpoints

- GET /serviceCatalog/v1/serviceSpecification

Post Conditions

All service specifications of the provider are successfully returned to be viewed.

Applicable Requirements

- PSI-02-02-05-01
- PSI-02-02-05-02

eTOM Reference

The operation is based on 1.4.15.1 and 1.4.19.2 process identifiers from the eTOM.

5.2.3 TOD-02-03-Product_Catalog_Management

The Product Catalog Management task takes care of the maintenance of product specifications available in the PSS, brought in by providers.

A provider wants to utilize a PSS to offer their products to the users of the PSS. The products implement a product specification (describing general characteristics of the product), and they bundle one or more services and/or on-site resources.

Therefore, a provider needs to be able to register(create) product specifications to the PSS, modify, remove or view them. Another PSS needs to be able to view the product specifications as well.

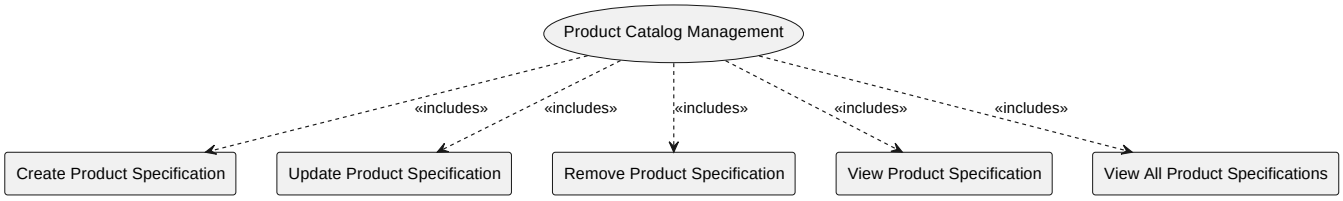

Figure 5.44: TOD-02-03: Product Catalog Management

	Customer	Provider	Other PSS	Governance
Create Product Specification		✓		
Update Product Specification		✓		
Remove Product Specification		✓		
View Product Specification	✓	✓	✓	✓
View All Product Specifications	✓	✓	✓	✓

Table 5.7: Product Catalog Management Matrix.

eTOM Reference

The task is based on the 1.2.22 and 1.2.7 process identifiers from the eTOM.

5.2.3.1 TOD-02-03-01-Create_Product_Specification

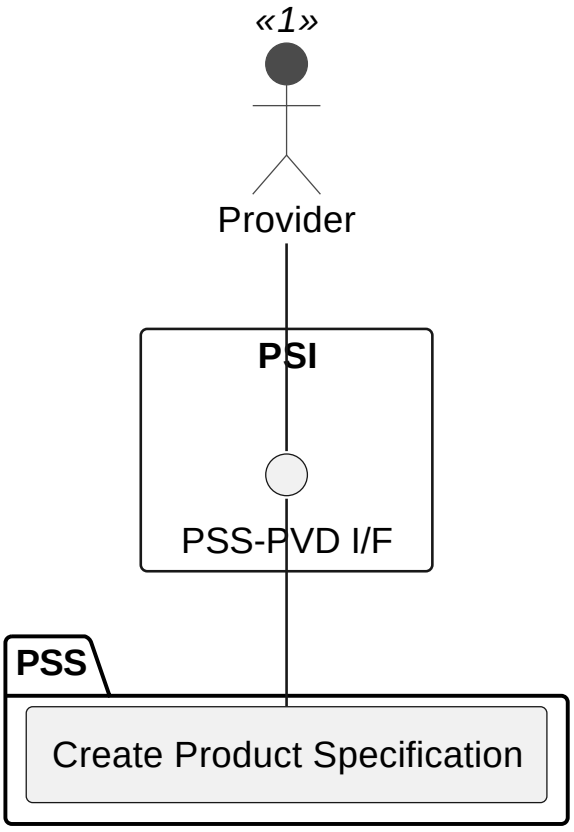

Figure 5.45: TOD-02-03-01: Create Product Specification

Prerequisites

The product specification does not exist in the PSS datastore.

Main operation

Creates a new product specification with its characteristics and references to resource/service specifications via a standard interface specification.

> Note: It is possible to create a product specification which will be available in the future by setting the *validFor* property with a future time reference.

Some properties of a product specification are:

- *name* - Short name of the target product
- *description* - Description of the target product
- *productNumber* - Identification number assigned to uniquely identify the specification
- *bundledProductSpecification* - If the product is a bundle of multiple products, a list of the related product specifications
- *resourceSpecification* - List of resources that are required to realise the target product

- *serviceSpecification* - List of services that are required to realise the target product
- *targetProductSchema* - Name and reference to the JSON Schema defining the type of product described by this specification.
- *productSpecCharacteristic* - List of distinctive features of the target product such as ‘networkUptime’, ‘dataAllowance’, etc.
- *relatedParty* - Usually reference to the provider that offers the product
- *lifecycleStatus* - Current lifecycle status of the product specification (e.g. active, draft, etc.)
- *validFor* - Time period of validity of the product specification

REST Endpoints

- POST /productCatalog/v1/productSpecification

Post Conditions

The product specification is successfully created in the PSS datastore.

Applicable Requirements

- PSI-02-03-01-01
- PSI-02-03-01-02
- PSI-02-03-01-03

eTOM Reference

The operation is based on 1.2.22.1 and 1.2.23.2 process identifiers from the eTOM.

5.2.3.2 TOD-02-03-02-Update_Product_Specification

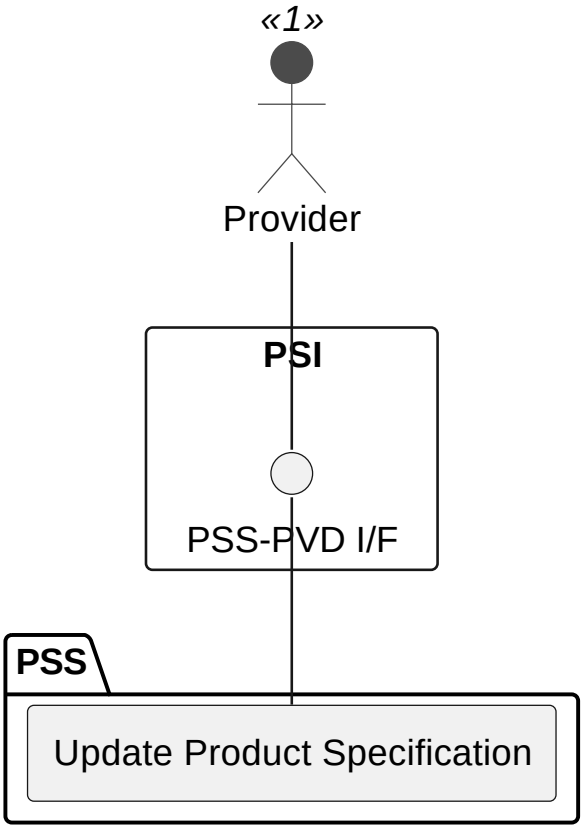

Figure 5.46: TOD-02-03-02: Update Product Specification

Prerequisites

The product specification exists in the PSS datastore. A PSS might reject the update of essential characteristics like the frequency band. In that case, the client shall set an expiration date for the current specification and create a new one that is valid thereafter.

Main operation

Updates an existing product specification via a standard interface specification.

REST Endpoints

- PATCH /productCatalog/v1/productSpecification/{id}

Post Conditions

The product specification is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-02-03-02-01
- PSI-02-03-02-02

eTOM Reference

The operation is based on 1.2.22.1 and 1.2.23.2 process identifiers from the eTOM.

5.2.3.3 TOD-02-03-03-Remove_Product_Specification

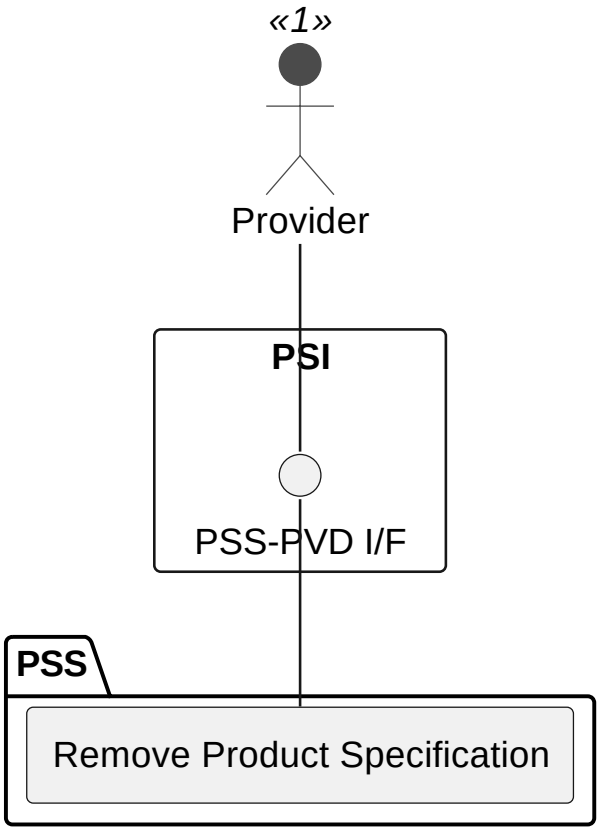

Figure 5.47: TOD-02-03-03: Remove Product Specification

Prerequisites

The product specification exists in the PSS datastore.

Main operation

Removes a product specification either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /productCatalog/v1/productSpecification/{id}

Post Conditions

The product specification is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-02-03-03-01
- PSI-02-03-03-02

eTOM Reference

The operation is based on 1.2.22.1 and 1.2.23.2 process identifiers from the eTOM.

5.2.3.4 TOD-02-03-04-View_Product_Specification

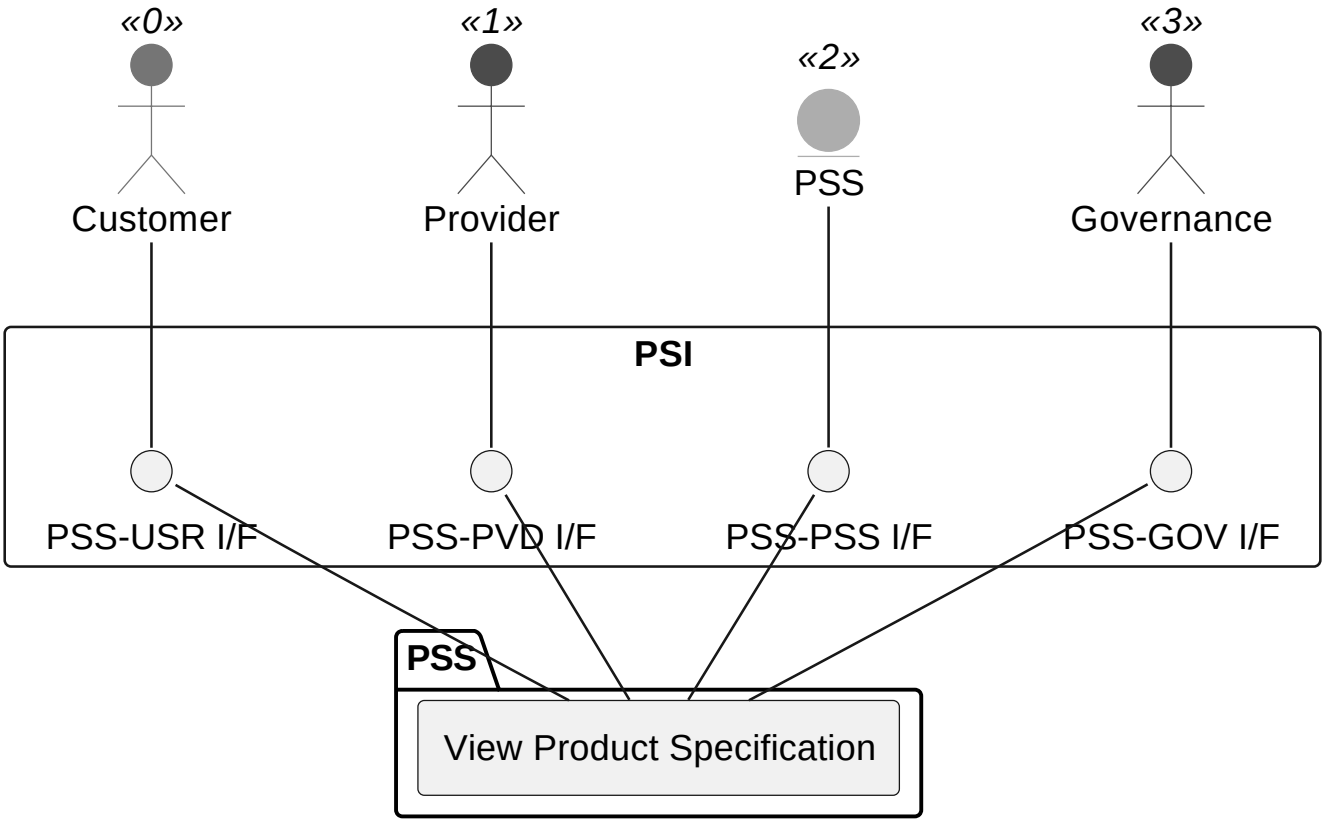

Figure 5.48: TOD-02-03-04: View Product Specification

Prerequisites

The product specification exists in the PSS datastore.

Main operation

Gets a product specification of the provider with a specific identifier via a standard interface specification.

REST Endpoints

- GET /productCatalog/v1/productSpecification/{id}

Post Conditions

The product specification is successfully returned to be viewed.

Applicable Requirements

- PSI-02-03-04-01
- PSI-02-03-04-02
- PSI-02-03-04-03

eTOM Reference

The operation is based on 1.2.22.1 and 1.2.23.2 process identifiers from the eTOM.

5.2.3.5 TOD-02-03-05-View_All_Product_Specifications

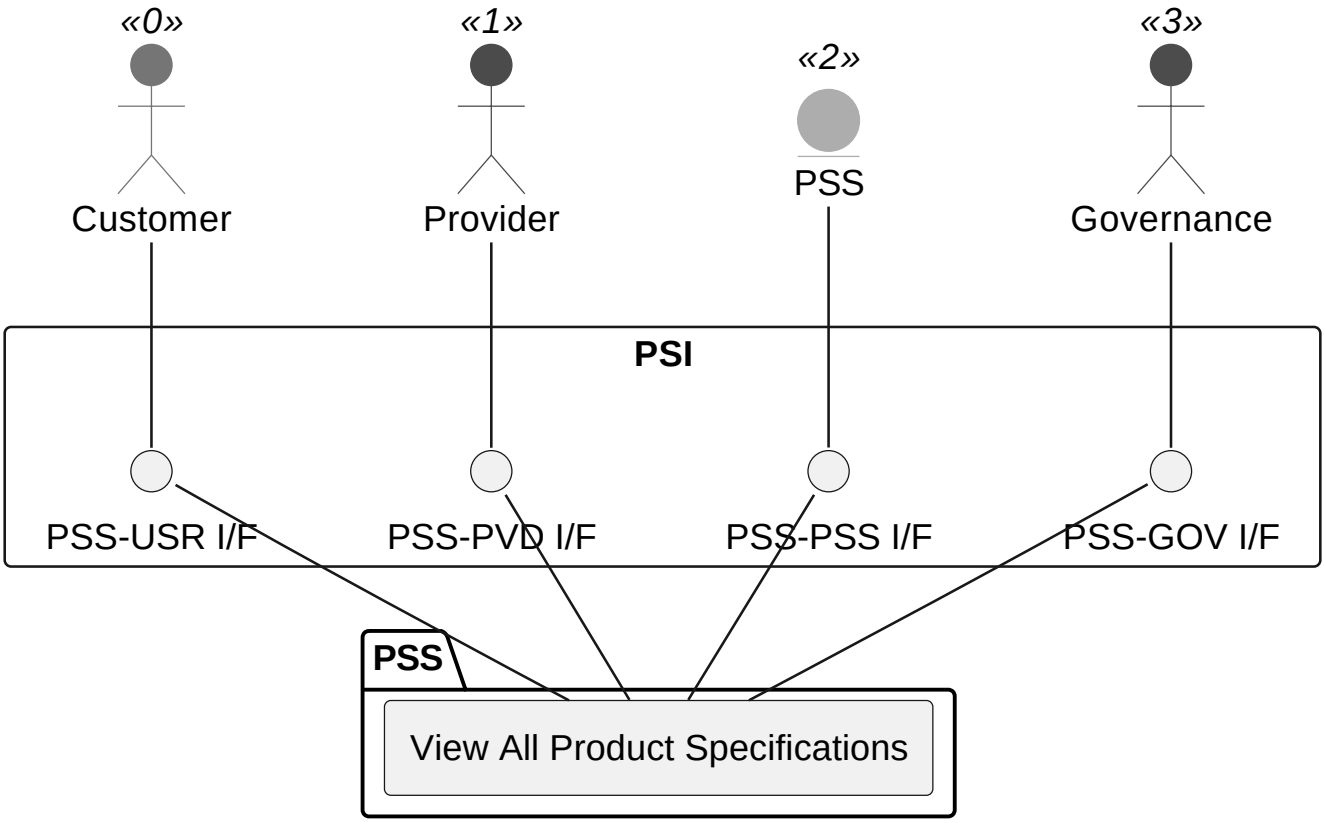

Figure 5.49: TOD-02-03-05: View All Product Specifications

Prerequisites

Product specifications of the provider exist in the PSS datastore.

Main operation

Gets all product specifications of the provider via a standard interface specification.

REST Endpoints

- GET /productCatalog/v1/productSpecification

Post Conditions

All product specifications of the provider are successfully returned to be viewed.

Applicable Requirements

- PSI-02-03-05-01

eTOM Reference

The operation is based on 1.2.22.1 and 1.2.23.2 process identifiers from the eTOM.

5.2.4 TOD-02-04-Product_Offering_Management

The Product Offering Management task takes care of the maintenance of product offerings available in the PSS, brought in by providers.

A provider wants to utilize a PSS to make a concrete offer with a price and SLS of a product specification to the users of the PSS.

Therefore, a provider needs to be able to register(create) product offerings to the PSS, modify, remove or view them. Another PSS needs to be able to view the product offerings as well.

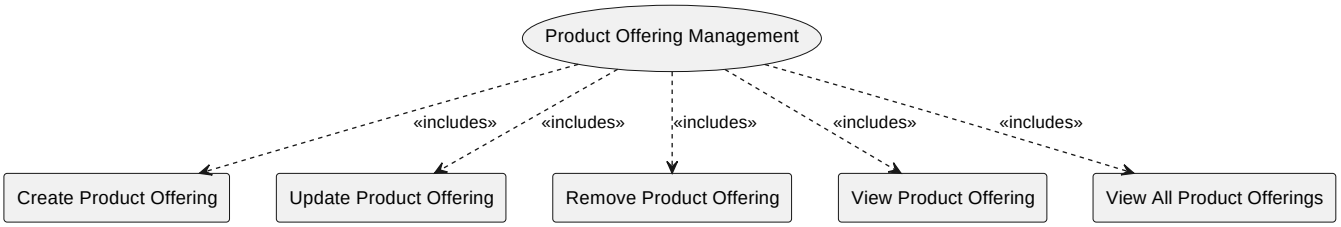

Figure 5.50: TOD-02-04: Product Offering Management

	Customer	Provider	Other PSS	Governance
Create Product Offering		✓		
Update Product Offering		✓		
Remove Product Offering		✓		
View Product Offering	✓	✓	✓	
View All Product Offerings	✓	✓	✓	

Table 5.8: Product Offering Management Matrix.

eTOM Reference

The task is based on the 1.2.7.2 process identifier from the eTOM.

5.2.4.1 TOD-02-04-01-Create_Product_Offering

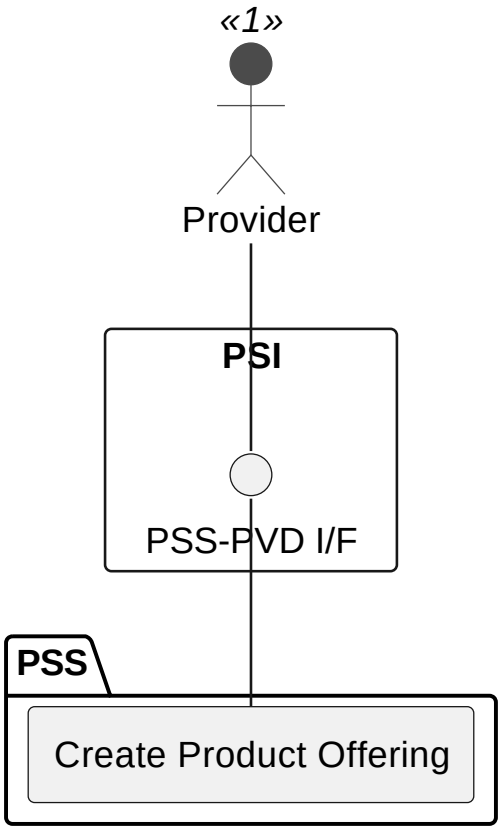

Figure 5.51: TOD-02-04-01: Create Product Offering

Prerequisites

The product offering does not exist in the PSS datastore.

Main operation

Creates a new product offering with product price, SLS, etc. via a standard interface specification.

> Note: It is possible to create a product offering which will be available in the future by setting the *validFor* property with a future time reference.

Some properties of a product offering are:

- *name* - Short name of the target offering
- *description* - Description of the target offering
- *productOfferingTerm* - Condition under which the offering is made available to customers, for example different commitment periods
- *productSpecification* - Reference to the product specification the target offering is about
- *serviceLevelSpecification* - SLS for the product offering
- *prodSpecCharValueUse* - Applied characteristics of the product specification, when there are variants

- *productOfferingPrice* - Price components for the offered product. Combinations of different types or multiple entries of the same type are possible.
- “one-time” prices are related to a singular payment (e.g. for delivered hardware).
- “recurring” prices are paid regularly (e.g. services or leased hardware). The interval can be defined by the sub-properties `recurringChargePeriod` (e.g. “month”, “week” or “day”) and `recurringChargePeriodLength` (defaults to 1).
- “per-use” prices depend on the usage of a product. The sub-property `unitOfMeasure` defines the basis for calculations (e.g. “per Minute” or “per 100 MB”). `recurringChargePeriod` and `recurringChargePeriodLength` are used to define the billing period.
- *accessProbability* - Probability that an order of this offering can be fulfilled
- *lifecycleStatus* - Current lifecycle status of the product offering (e.g. active, draft, etc.)
- *validFor* - Time period of validity of the product offering

REST Endpoints

- POST /productCatalog/v1/productOffering

Post Conditions

The product offering is successfully created in the PSS datastore.

Applicable Requirements

- PSI-02-04-01-01
- PSI-02-04-01-02

eTOM Reference

The operation is based on 1.2.7.2.3 process identifier from the eTOM.

5.2.4.2 TOD-02-04-02-Update_Product_Offering

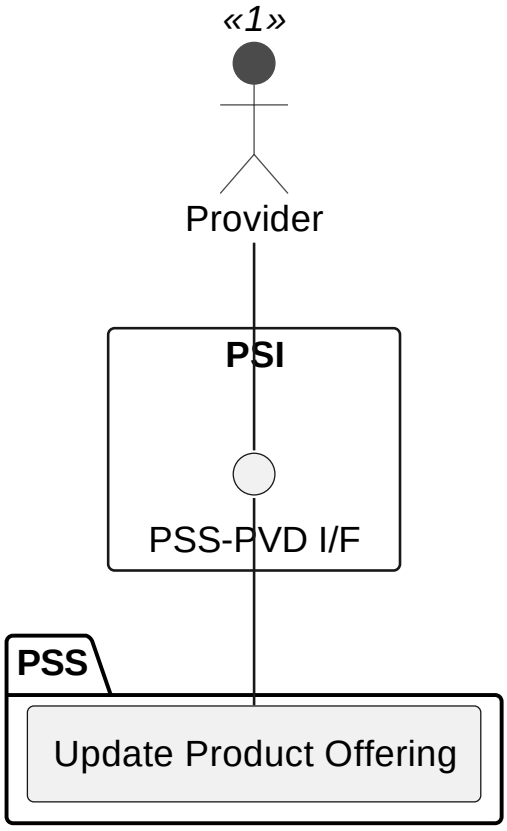

Figure 5.52: TOD-02-04-02: Update Product Offering

Prerequisites

The product offering exists in the PSS datastore.

Main operation

Updates an existing product offering via a standard interface specification.

REST Endpoints

- PATCH /productCatalog/v1/productOffering/{id}

Post Conditions

The product offering is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-02-04-02-01
- PSI-02-04-02-02
- PSI-02-04-02-03
- PSI-02-04-02-04

eTOM Reference

The operation is based on 1.2.7.2.3 process identifier from the eTOM.

5.2.4.3 TOD-02-04-03-Remove_Product_Offering

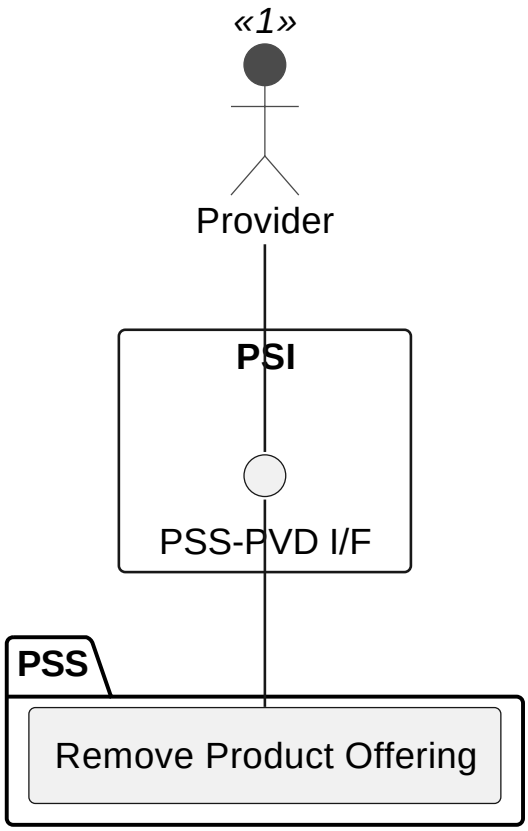

Figure 5.53: TOD-02-04-03: Remove Product Offering

Prerequisites

The product offering exists in the PSS datastore.

Main operation

Removes a product offering either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /productCatalog/v1/productOffering/{id}

Post Conditions

The product offering is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-02-04-03-01
- PSI-02-04-03-02

eTOM Reference

The operation is based on 1.2.7.2.3 process identifier from the eTOM.

5.2.4.4 TOD-02-04-04-View_Product_Offering

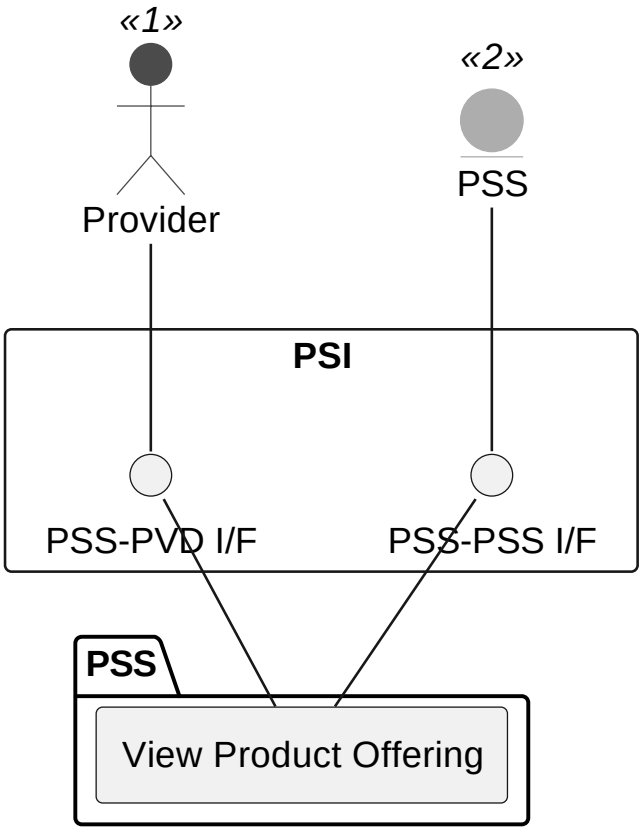

Figure 5.54: TOD-02-04-04: View Product Offering

Prerequisites

The product offering exists in the PSS datastore.

Main operation

Gets a product offering of the provider with a specific identifier via a standard interface specification.

REST Endpoints

- GET /productCatalog/v1/productOffering/{id}

Post Conditions

The product offering is successfully returned to be viewed.

Applicable Requirements

- PSI-02-04-04-01
- PSI-02-04-04-02

eTOM Reference

The operation is based on 1.2.7.2.3 process identifier from the eTOM.

5.2.4.5 TOD-02-04-05-View_All_Product_Offerings

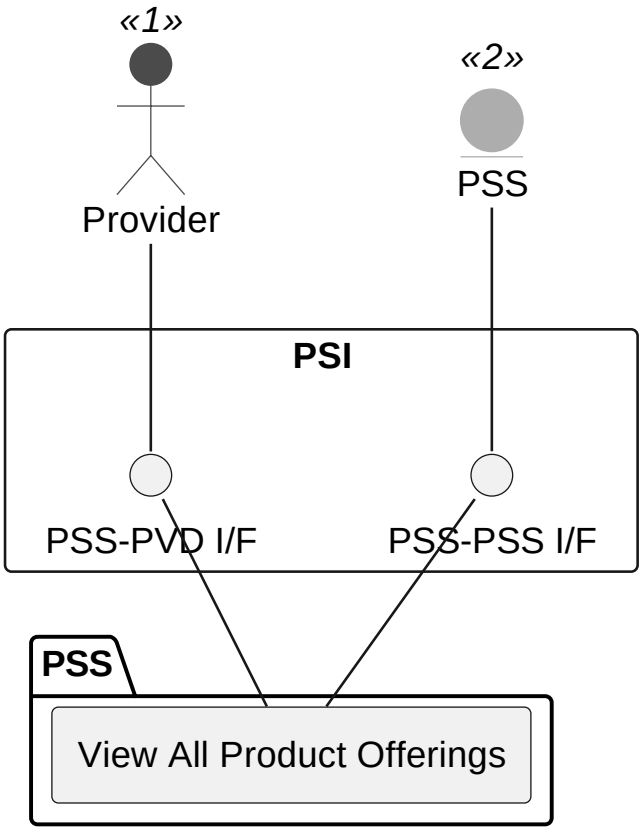

Figure 5.55: TOD-02-04-05: View All Product Offerings

Prerequisites

Product offerings of the provider exist in the PSS datastore.

Main operation

Gets all product offerings of the provider via a standard interface specification.

REST Endpoints

- GET /productCatalog/v1/productOffering

Post Conditions

All product offerings of the provider are successfully returned to be viewed.

Applicable Requirements

- PSI-02-04-05-01

eTOM Reference

The operation is based on 1.2.7.2.3 process identifier from the eTOM.

5.3 TOD-03-Product_Inquiry_And_Ordering

The category consists of tasks and operations related to customer's inquiries and product ordering.

5.3.1 TOD-03-01-Customer_Inquiry_Management

The Customer Inquiry Management task takes care of the handling of inquiries sent by a customer and responded by a PSS or provider.

The PSS may provide different ways for the customer to create an inquiry, depending on the expertise of the user. These can range from just selecting from templates with commonly used product types, optionally customizing the characteristics or even the manual definition of the communication needs.

An inquiry is stateful and can be processed in different ways: Firstly, the PSS can implement a matchmaking algorithm that searches the local database to provide immediate results. This process can also be partially or fully outsourced to other PSSs and sufficiently advanced provider systems that are able to respond in real-time. The PSS will then aggregate, rank and possibly filter the results before making them available to the customer. In the case of a Request-For-Quote or Invitation-To-Tender, human intervention is foreseen to tailor a product offering, which can take some hours or days.

To prevent long delays for the customer, the response time of the providers to an RFQ or ITT should be limited. The governance of the PSS can set the maximum inquiry response time for each provider in their party profiles according to their SLA. The provider is then responsible to respect the response time and provide the inquiry results within the expected time frame. The PSS is responsible to implement mechanisms to monitor the response times of the provider for each customer inquiry and ensure the provider respects the response time provided by the governance of the PSS. In case the provider exceeds the given deadline, its product offering(s) will not be considered for the issued inquiry.

The customer inquiry result may contain

- Concrete product offerings which can be ordered directly (see TOD-03-02)
- Product instances from the inventory that are available to fulfil the order
- Matching product specifications to request a tailored offering for (RFQ or ITT)

If the customer is not satisfied with the result, they may refine their criteria and send a new inquiry. To speed up the access and the review of responses to delegated inquiries, a PSS can download and cache offerings as they become available.

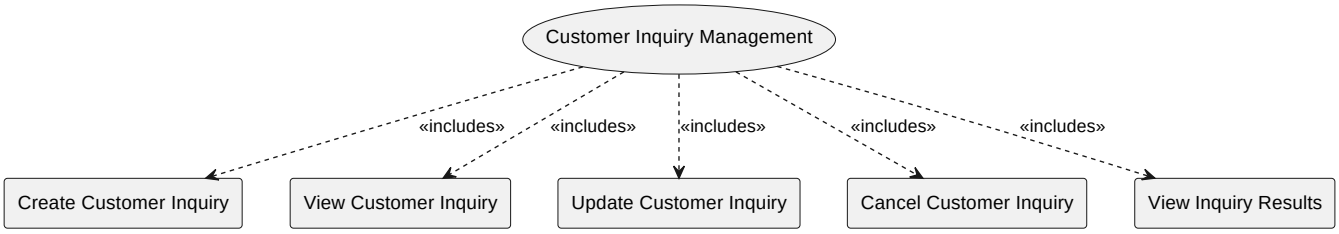

Figure 5.56: TOD-03-01: Customer Inquiry Management

	Customer	Provider	Other PSS	Governance
Create Customer Inquiry	✓		(✓)	
View Customer Inquiry	✓	✓	✓	
Update Customer Inquiry	✓		(✓)	
Cancel Customer Inquiry	✓		(✓)	
View Inquiry Results	✓		✓	

Table 5.9: Customer Inquiry Management Matrix.

Please note, Checkmarks in parentheses indicate that these operations are carried out via events (see TOD-01-02), not via direct call of the REST endpoints.

eTOM Reference

The task is based on the 1.3.5 process identifier from the eTOM.

5.3.1.1 TOD-03-01-01-Create_Customer_Inquiry

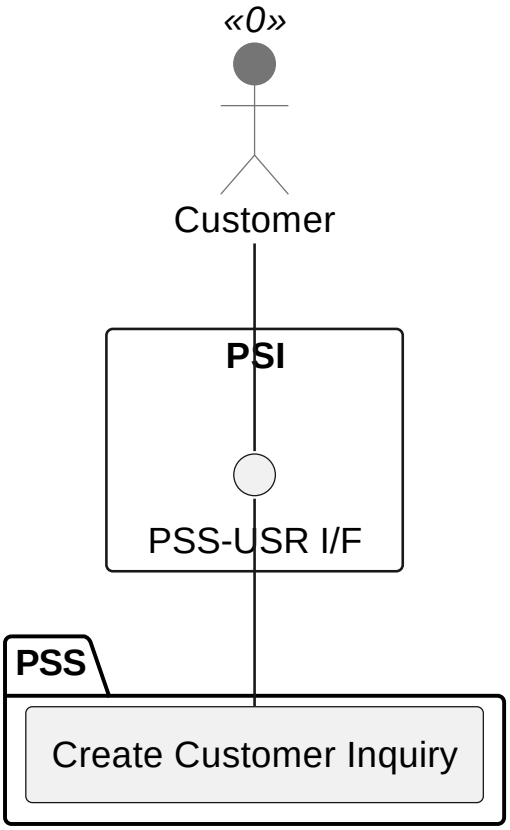

Figure 5.57: TOD-03-01-01: Create Customer Inquiry

Prerequisites

None

Main operation

Creates a new customer inquiry via a standard interface specification. The inquiry will be processed asynchronously, either via an algorithm or manually by a provider. To do that, the PSS may forward the created inquiry to eligible providers via Event Management API, which is also used by them to propagate the update back.

The implementation must provide **at least** the maximum estimated response time to prevent long delays. Optionally, it can also define a minimum and an average response time. These values are either determined by the algorithm run time, the governance of the PSS that can set the response time of each provider in their party profiles, the customer's preference or a default PSS configuration.

Some properties of a customer inquiry are:

- *customerProfile* - Automatically filled with authenticated data from the customer profile by the PSS. The set of characteristics has to be agreed between the involved systems a priori, which enables different levels of anonymization. For example, the governances of both systems may decide to just exchange an anonymized indication of the kind of customer and their country code instead of their real name and address. If the *priority* of a customer is included, there must be a common understanding of what the values mean or one system has to map these accordingly. For example, a "high priority" customer of a commercial PSS may be less important than a customer with the same level denomination in a governmental system. Note that the customer may be allowed to select a *lower* priority for their request, which shall then be indistinguishable from other low priority requests.
- *customerResources* - List of resource specifications, which are already owned by the customer and shall be used for the inquired service
- *inquiredProducts* - Specifications of products the customer needs. These are built from characteristics with minimum, maximum and target values. The customer can prioritize characteristics (e.g. "availability is more important than information rate"). The provider can use this information when they can not meet all target values and have to waive on some. If this inquiry is an RFQ, the customer may specify a specification ID from the catalog to get an offering for that specific item.
- *providers* - A list of third parties to contact. If the list is empty, then the PSS will take into consideration the product offerings of all providers. This is typical for a matchmaking customer inquiry. For an RFQ, for example, the customer might specify a list of preferred providers from which they want to get an offer.

REST Endpoints

- POST /customerInquiry/v1/customerInquiry

Post Conditions

- The customer inquiry is successfully created in the PSS datastore.
- The processor of the inquiry is notified.

Applicable Requirements

- PSI-03-01-01-01
- PSI-03-01-01-02

- PSI-03-01-01-03
- PSI-03-01-01-04
- PSI-03-01-01-05
- PSI-03-01-01-06
- PSI-03-01-01-07
- PSI-03-01-01-08
- PSI-03-01-01-09
- PSI-03-01-01-10
- PSI-03-01-01-11
- PSI-03-01-01-12

eTOM Reference

The operation is based on 1.3.5.1 process identifiers from the eTOM.

5.3.1.2 TOD-03-01-02-View_Customer_Inquiry

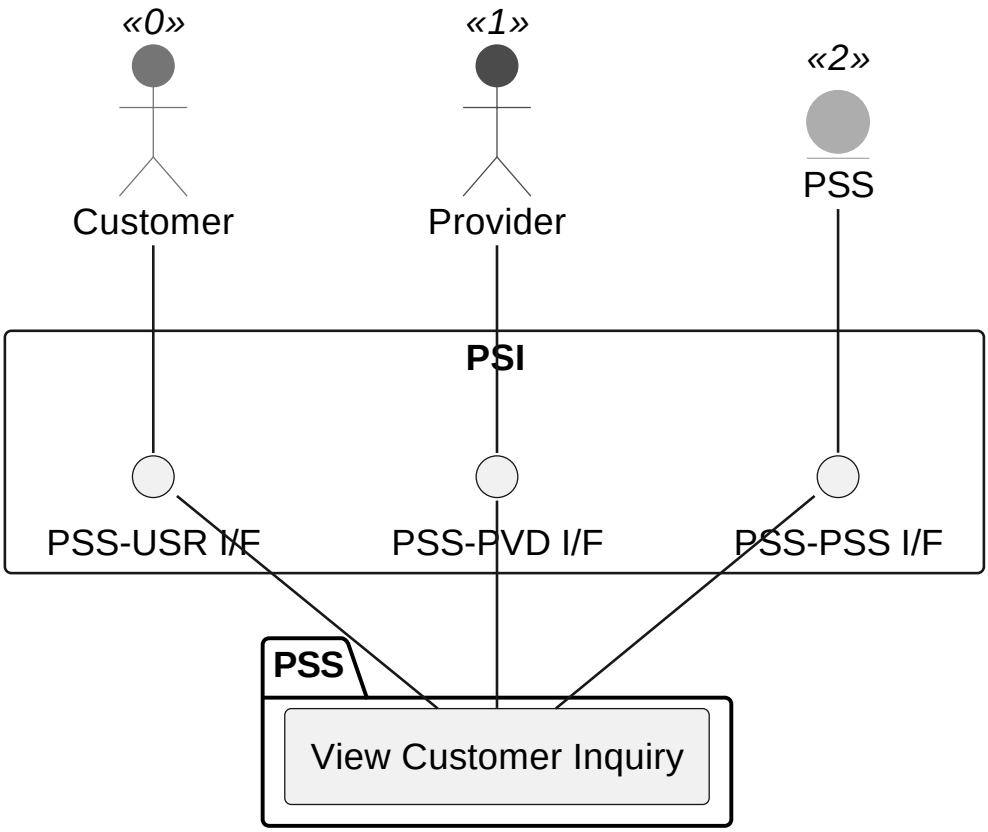

Figure 5.58: TOD-03-01-02: View Customer Inquiry

Prerequisites

The customer inquiry exists in the PSS datastore.

Main operation

Gets a customer inquiry of the customer with a specific identifier via a standard interface specification.

REST Endpoints

- GET /customerInquiry/v1/customerInquiry/{id}

Post Conditions

The customer inquiry is successfully returned to be viewed.

Applicable Requirements

- PSI-03-01-02-01
- PSI-03-01-02-02
- PSI-03-01-02-03
- PSI-03-01-02-04

eTOM Reference

The operation is based on 1.3.5.6 process identifiers from the eTOM.

5.3.1.3 TOD-03-01-03-View_Inquiry_Results

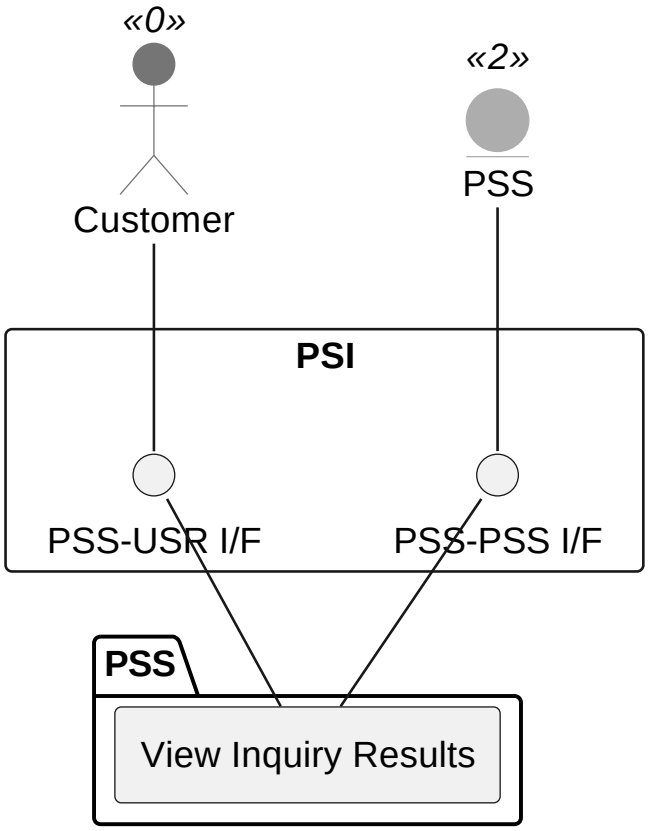

Figure 5.59: TOD-03-01-03: View Inquiry Results

Prerequisites

The customer inquiry was processed and produced results.

Main operation

The customer wants to see the results of their inquiry. These can be of different nature:

- The foundation of all results are product specifications that are within the boundaries of the inquired characteristics. When there is no more data, these may be used to initiate a RFQ or ITT to retrieve a product offering.
- Whenever possible, each product specification should be complemented with a product offering. This allows the customer to order the product immediately without a (possibly lengthy) RFQ process. If more than one offering for the same product is matching the inquiry (e.g. with different conditions), the provider system can either return them as multiple results with different priorities or in a single entry. The PSS must support both variants, but may split an entry depending on the subsequent processing needs. When there is no more data, the actual availability is unknown and may be checked via TOD-05-04-01.
- Lastly, the results can contain available product instances from the inventory. In this case, the PSS can show the availability and no further checks are necessary, which may result in higher ranking. Note that this does not imply a reservation, so the actual order may still be rejected if the resource was booked in the meantime.

In addition, the inquiry result can contain relationship definitions between a single result entity and an inquired product. This is particularly necessary if the inquiry includes more than one inquired product and potentially results

in multiple entities per inquiry result. With the help of this approach a clear assignment is ensured between the inquired products and the results. Furthermore, relationship-dependent details are provided in this way, such as the coverage (full or partial matches), a deviation of the service period or the calculated prices.

All of these can be fetched from the corresponding endpoint listed below. It should be noted that different responses are used if there is no content because the inquiry was not processed or no entities were found. If the inquiry is processed by an algorithm, the endpoints may optionally block for a few seconds to wait for results. It is up to the implementation (or human decision) whether to include partial matches when there are others that fully match the target values.

REST Endpoints

- GET /customerInquiry/v1/customerInquiry/{id}/results

Post Conditions

The inquiry results are returned to be viewed.

Applicable Requirements

- PSI-03-01-03-01
- PSI-03-01-03-02
- PSI-03-01-03-03
- PSI-03-01-03-04
- PSI-03-01-03-05
- PSI-03-01-03-06
- PSI-03-01-03-07
- PSI-03-01-03-08

eTOM Reference

None

5.3.1.4 TOD-03-01-04-Update_Customer_Inquiry

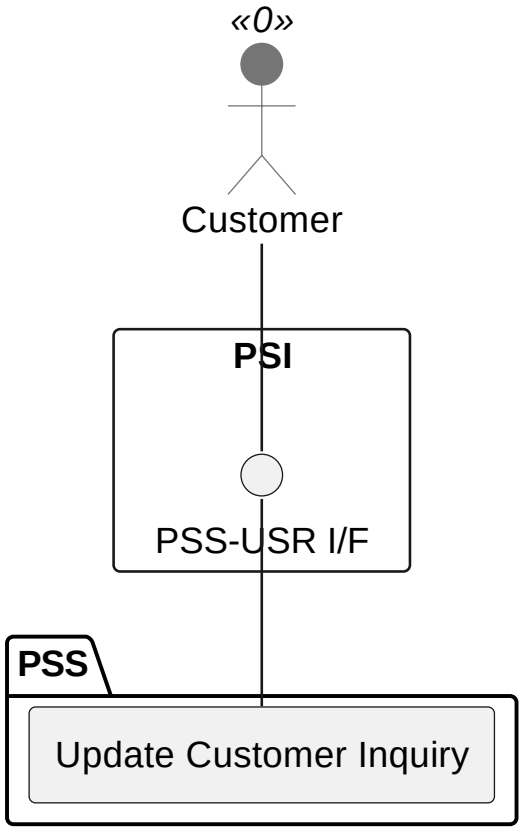

Figure 5.60: TOD-03-01-04: Update Customer Inquiry

Prerequisites

The customer inquiry exists in the PSS datastore.

Main operation

Updates the inquired provider list of an existing customer inquiry via a standard interface specification. The inquiry will be processed asynchronously, either via an algorithm or manually by a provider. To do that, the PSS may forward the inquiry to eligible providers via Event Management API. This includes

- sending inquiry cancellations to providers which are no longer part of the inquired providers list and
- initial sending of the inquiry to the providers newly added to the list of inquired providers.

Updates on a customer inquiry are restricted to the inquired providers list only, because any other change of the inquiry specification has to result in processing the inquiry again. Therefore, if changes need to be made to the inquiry specification, the active inquiry can be cancelled and a new one with the new specifications can be initiated. A PSS may support the customers in the definition process, e.g. by creating an inquiry based on a previously created one.

Details concerning the structure and processing of a customer inquiry are explained in TOD-03-01-01.

REST Endpoints

- PATCH /customerInquiry/v1/customerInquiry/{id}

Post Conditions

- The customer inquiry is successfully updated in the PSS datastore.
- The processor of the inquiry is notified.

Applicable Requirements

- PSI-03-01-04-01
- PSI-03-01-04-02
- PSI-03-01-04-03
- PSI-03-01-04-04
- PSI-03-01-04-05

eTOM Reference

The operation is based on 1.3.5.1 process identifiers from the eTOM.

5.3.1.5 TOD-03-01-05-Cancel_Customer_Inquiry

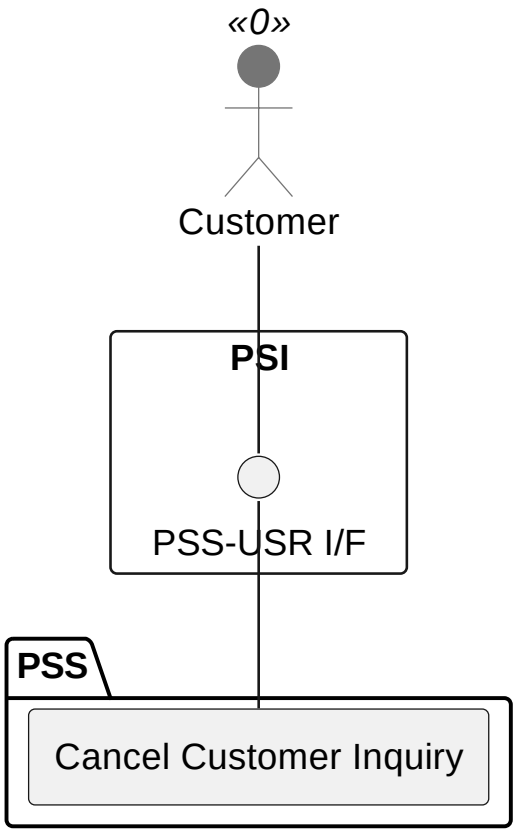

Figure 5.61: TOD-03-01-05: Cancel Customer Inquiry

Prerequisites

The customer inquiry exists in the PSS datastore.

Main operation

Cancels an existing customer inquiry via a standard interface specification. Due to the asynchronous processing, PSS forwards the cancellation state to eligible providers via the Event Management API.

REST Endpoints

- DELETE /customerInquiry/v1/customerInquiry/{id}

Post Conditions

- The customer inquiry is successfully marked as cancelled in the PSS datastore.
- The processor of the inquiry is notified.

Applicable Requirements

- PSI-03-01-05-01

eTOM Reference

The operation is based on 1.3.5.1 process identifiers from the eTOM.

5.3.2 TOD-03-02-Product_Order_Management

This task contains all operations to order products (and therefore services and resources).

The Product Order Management task takes care of the handling of orders sent by a customer to a PSS.

The customer wants to utilise a PSS to place an order. The PSS receives the order and is responsible to forward it to the respective provider that can realise the order items selected by the customer. The provider usually requires some time to process the order and update its state to `acknowledged` or `completed` for instance. Therefore, the PSS and the provider establish asynchronous communication via the Event Management API (see TOD-01-02) towards exchanging order updates. The PSS is then responsible to inform the customer about the up-to-date state of the order.

To prevent long delays for the customer, the governance of the PSS can set an order response time for each provider in their party profiles. The provider is then responsible to respect the response time and `acknowledge` the order within the expected time frame. The PSS is responsible to implement mechanisms to monitor the response times of the provider for each order and ensure the provider respects the response time provided by the governance of the PSS.

Additionally, the customer and provider need to be able to modify properties (e.g. the billing information) of an existing order in the PSS if the current state of the order allows that. For example, if an order is `completed`, any updates should be rejected by the PSS. If they want to change product characteristics (e.g. increase the bandwidth), a new order has to be created and will replace the existing one. They also need to be able to view an existing order or all the orders that are applicable to them in the PSS.

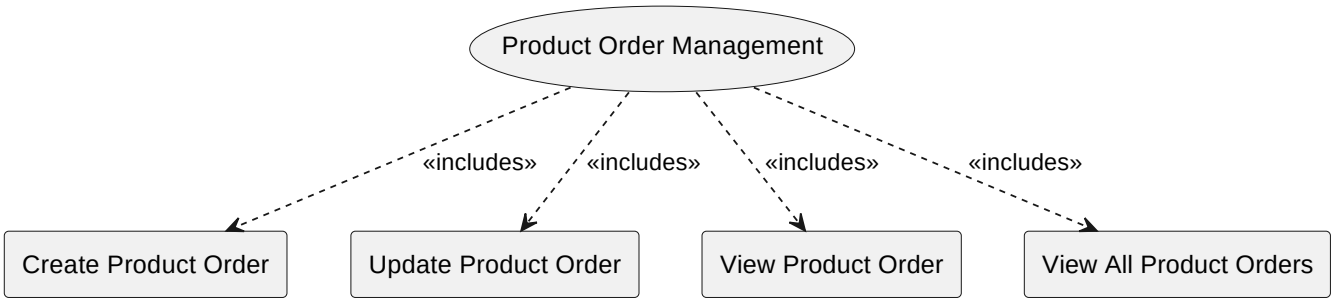

Figure 5.62: TOD-03-02: Product Order Management

	Customer	Provider	Other PSS	Governance
Create Product Order	✓		(✓)	
Update Product Order	✓	✓	(✓)	
View Product Order	✓	✓		
View All Product Orders	✓	✓		

Table 5.10: Product Order Management Matrix.

eTOM Reference

The task is based on the 1.3.2 process identifier from the eTOM.

5.3.2.1 TOD-03-02-01-Create_Product_Order

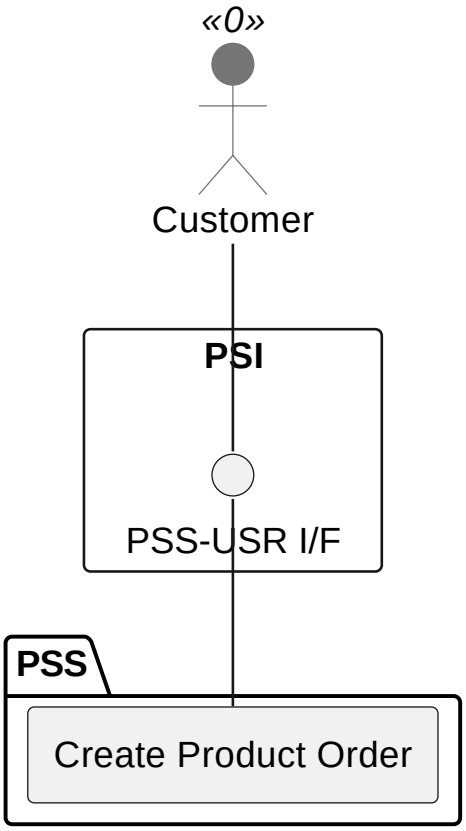

Figure 5.63: TOD-03-02-01: Create Product Order

Prerequisites

The product order does not exist in the PSS datastore.

Main operation

The customer creates a new product order to the PSS via a standard interface specification. The PSS then forwards the created order to the provider via Event Management API. This can be done in different stages of the product lifecycle:

1. Initially, the customer creates an order based on a product specification and offering from the catalog or matchmaking. The order is eventually fulfilled (on interface level) by **creating** product, service and resource instances in the inventory. It is not modified afterwards but remains archived.
2. The customer can request to change an existing product instance by creating a new product order. This can usually be done based on the boundaries defined in the product specification, e.g. by increasing the information rate. The new order is fulfilled (on interface level) by **updating** or **deleting** the product, service and resource instances in the inventory.

Some properties of the product order are:

- *productOrderItem* - List of product order items containing:
- *productOffering* - A reference to the ordered offering.

- *itemPrice* - A list of (one-time and recurring) item prices. An amount, usually of money, that represents the actual price paid by the customer for the order item.
- *action* - The action to be carried out on the product. Can be: add, modify, delete, noChange.
- *product* - A reference to the product to be modified or deleted (if applicable).
- *requestedStartDate*/*requestedEndDate* - The option for a start and end date deviating from the order.
- *productOrderItemRelationship* - A list of relations on item level, to reflect bundles and other dependencies.
- *orderTotalPrice* - A list of the combined prices of the items.
- *relatedParty* - A list of related parties for the order (e.g. customer, PSS and provider).

REST Endpoints

- POST /productOrdering/v1/productOrder

Post Conditions

The product is successfully created in the PSS datastore and later to the provider's datastore via the Event Management API.

Applicable Requirements

- PSI-03-02-01-01
- PSI-03-02-01-02
- PSI-03-02-01-03
- PSI-03-02-01-04

eTOM Reference

The operation is based on 1.3.3.5.2 process identifier from the eTOM.

5.3.2.2 TOD-03-02-02-Update_Product_Order

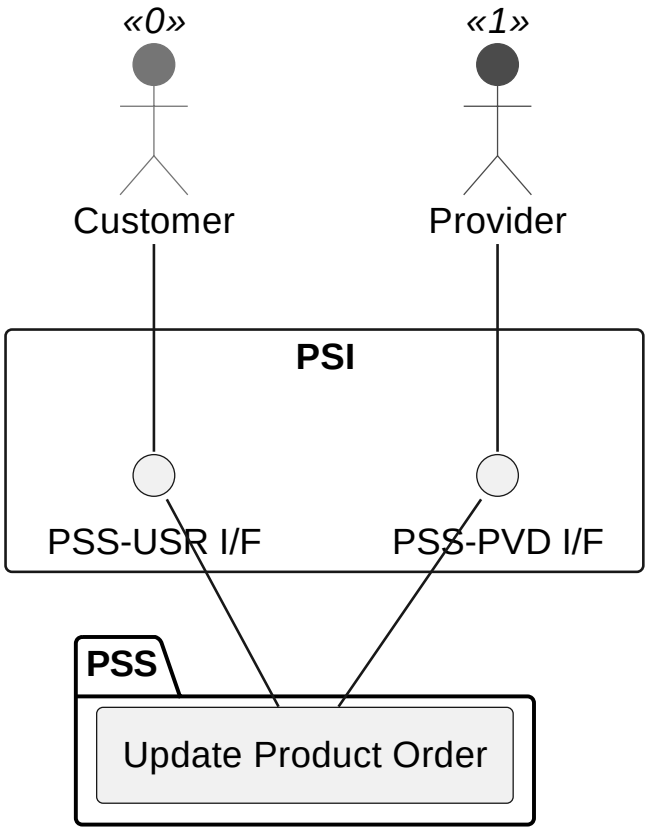

Figure 5.64: TOD-03-02-02: Update Product Order

Prerequisites

The product order exists in the PSS and provider's datastore.

Main operation

The customer updates an existing product order in the PSS via a standard interface specification. The PSS then forwards the update to the provider via the Event Management API. A provider can also update a product order in the PSS.

This operation is possible only if the current state of the order allows the change. It should not be possible to update orders that are `cancelled` or `completed`.

REST Endpoints

- PATCH /productOrdering/v1/productOrder/{id}

Post Conditions

The product order is successfully updated in the PSS and provider datastores.

Applicable Requirements

- PSI-03-02-02-01
- PSI-03-02-02-02

- PSI-03-02-02-03
- PSI-03-02-02-04

eTOM Reference

The operation is based on 1.3.3.5.3 process identifier from the eTOM.

5.3.2.3 TOD-03-02-03-View_Product_Order

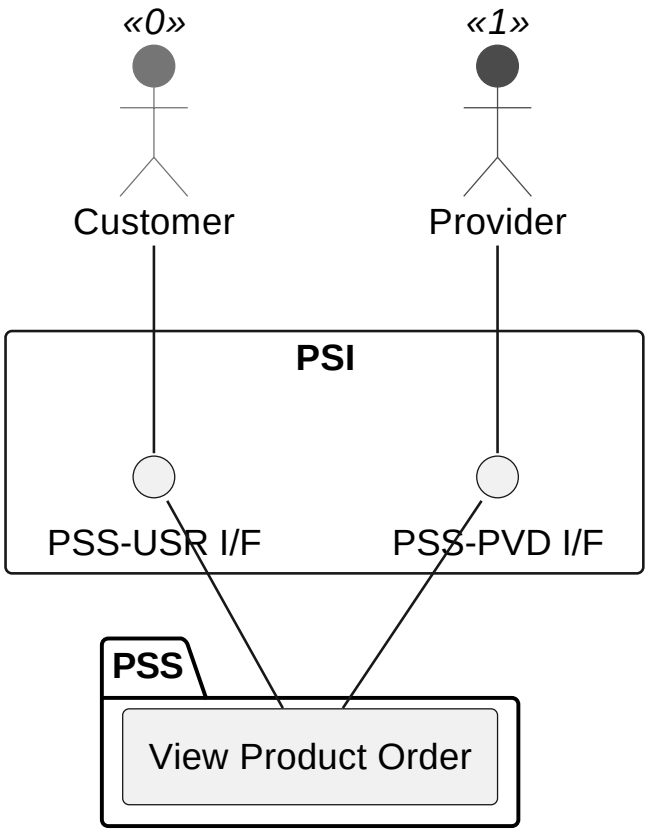

Figure 5.65: TOD-03-02-03: View Product Order

Prerequisites

The product order exists in the PSS datastore.

Main operation

Gets a product order with a specific identifier via a standard interface specification. The customer and the provider can request to view the product order from the PSS.

REST Endpoints

- GET /productOrdering/v1/productOrder/{id}

Post Conditions

The product order that the customer or provider can read, is successfully returned to be viewed.

Applicable Requirements

- PSI-03-02-03-01
- PSI-03-02-03-02

eTOM Reference

None

5.3.2.4 TOD-03-02-04-View_All_Product_Orders

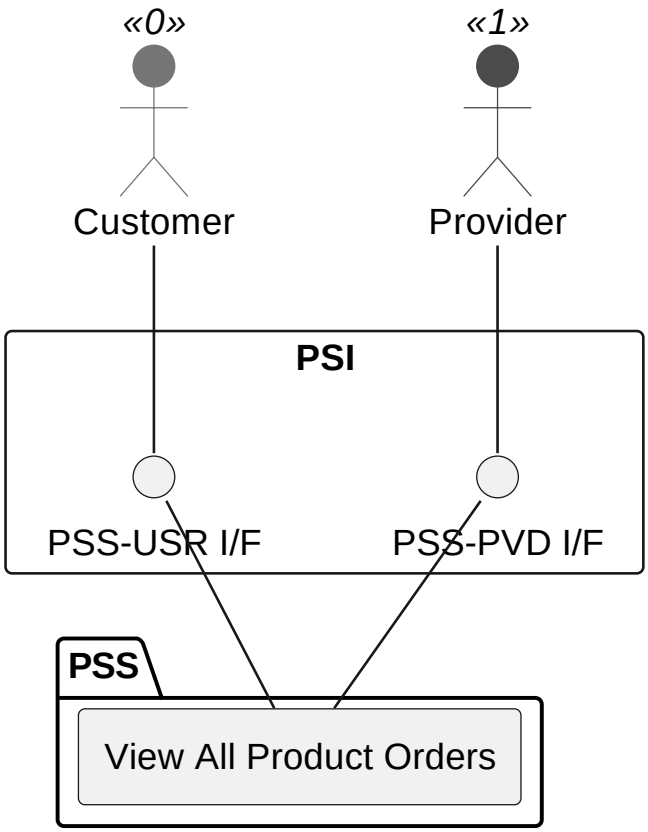

Figure 5.66: TOD-03-02-04: View All Product Orders

Prerequisites

Product orders exist in the PSS datastore.

Main operation

Gets all product orders of the customer from the PSS via a standard interface specification. The provider can also request to view all of their product orders from the PSS.

REST Endpoints

- GET /productOrdering/v1/productOrder

Post Conditions

All product orders which the customer or provider can read are successfully returned to be viewed.

Applicable Requirements

- PSI-03-02-04-01

eTOM Reference

None

5.3.3 TOD-03-03-Customer_Bill_Management

The Customer Bill Management task takes care of bills (invoices) produced for a customer for placed orders in the PSS. A customer bill or invoice is a document produced at the end of a regular back office process at the provider side which runs according to a bill cycle definition. The customer bill contains information about the total amount due to be paid by a customer for the ordered product(s) during the billing period, the due date for the payment, and other information like the order and attachment references.

A *provider* wants to utilise the PSS to publish the bill and make it available to the customer. Additionally, a provider might need to change the state of the bill in the PSS, for example when a customer has paid it.

The *customer* wants to utilise the PSS to find and retrieve one or several customer bills produced for them. Also, the provider can use the PSS to find and retrieve the bills that have been created by them to the PSS.

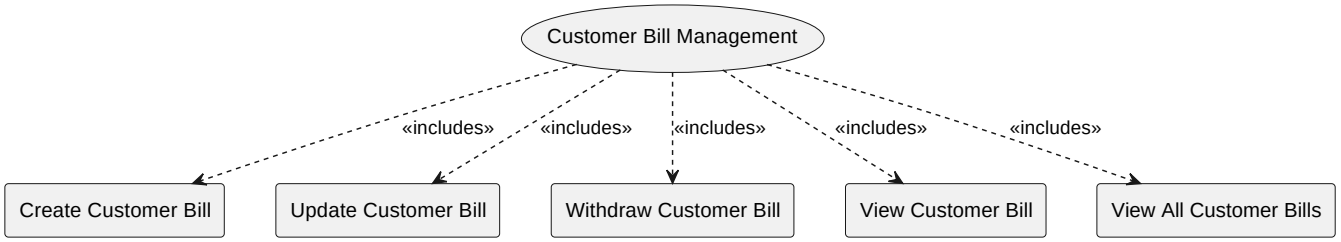

Figure 5.67: TOD-03-03: Customer Bill Management

	Customer	Provider	Other PSS	Governance
Create Customer Bill		✓	(✓)	
Update Customer Bill		✓	(✓)	
Withdraw Customer Bill		✓	(✓)	
View Customer Bill	✓	✓		
View All Customer Bills	✓	✓		

Table 5.11: Customer Bill Management Matrix.

eTOM Reference

The task is based on the 1.3.9 process identifier from the eTOM.

5.3.3.1 TOD-03-03-01-Create_Customer_Bill

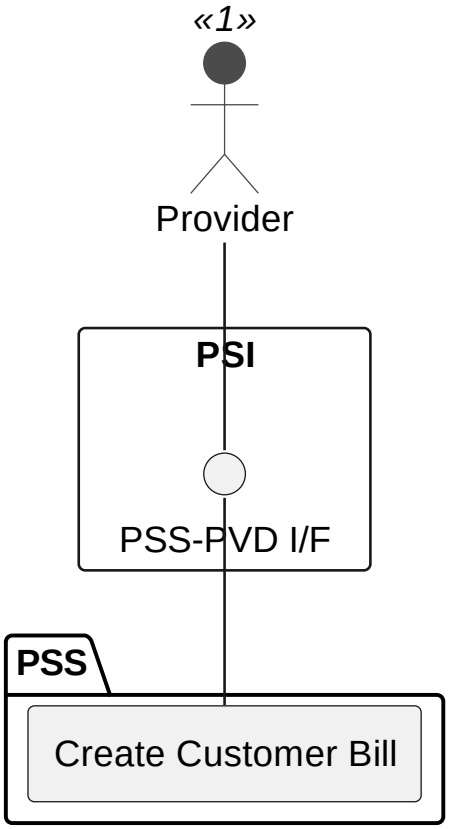

Figure 5.68: TOD-03-03-01: Create Customer Bill

Prerequisites

The customer bill does not exist in the PSS datastore.

Main operation

The provider creates a new customer bill to the PSS via a standard interface specification.

Some properties of the customer bill are:

- *billNo* - The bill number as a customer reference which is displayed on the bill.
- *billDocument* - A list of attachments comprising the bill.
- *amountDue* - The amount to be paid in a given currency.
- *paymentDueDate* - The date at which the amount due should have been paid.
- *state* - The status of the bill (e.g. ‘validated’, ‘sent’, ‘settled’, ‘partiallyPaid’).
- *billingAccount (optional)* - A reference to the account that is charged for the bill. It can also include the payment method.
- *appliedPayments (optional)* - A list of applied payments associated with the bill.

REST Endpoints

- POST /customerBillManagement/v1/customerBill

Post Conditions

The customer bill is successfully created in the PSS.

Applicable Requirements

- PSI-03-03-01-01
- PSI-03-03-01-02
- PSI-03-03-01-03
- PSI-03-03-01-04

eTOM Reference

The operation is based on 1.3.9.2 process identifier from the eTOM.

5.3.3.2 TOD-03-03-02-Update_Customer_Bill

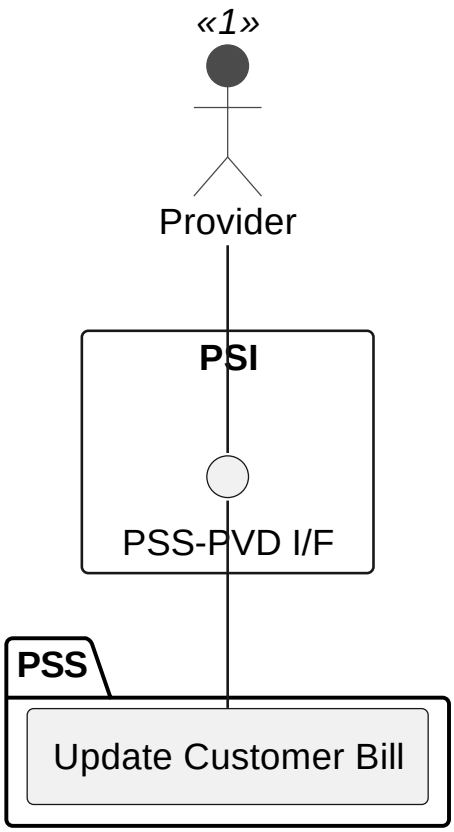

Figure 5.69: TOD-03-03-02: Update Customer Bill

Prerequisites

The customer bill exists in the PSS datastore.

Main operation

The provider updates the state of an existing customer bill in the PSS via a standard interface specification. It should not be possible to update the customer bill's properties that affect its traceability for tax reasons.

REST Endpoints

- PATCH /customerBillManagement/v1/customerBill/{id}

Post Conditions

The customer bill is successfully updated in the PSS.

Applicable Requirements

- PSI-03-03-02-01
- PSI-03-03-02-02

eTOM Reference

The operation is based on 1.3.9.4.3 process identifier from the eTOM.

5.3.3.3 TOD-03-03-03-View_Customer_Bill

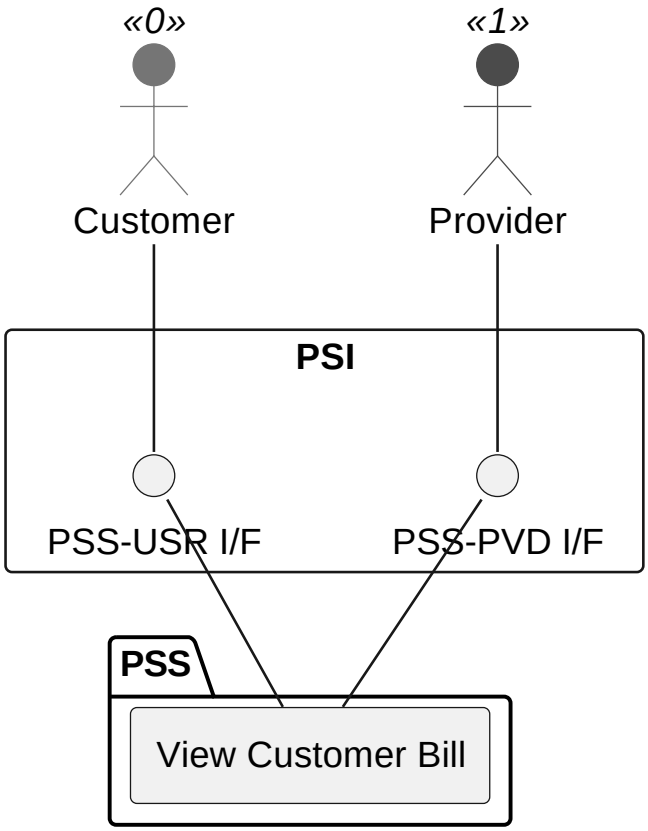

Figure 5.70: TOD-03-03-03: View Customer Bill

Prerequisites

The customer bill exists in the PSS datastore.

Main operation

Gets a customer bill with a specific identifier via a standard interface specification.

REST Endpoints

- GET /customerBillManagement/v1/customerBill/{id}

Post Conditions

The customer bill that the customer or the provider can read, is successfully returned to be viewed.

Applicable Requirements

- PSI-03-03-03-01
- PSI-03-03-03-02
- PSI-03-03-03-03
- PSI-03-03-03-04
- PSI-03-03-03-05
- PSI-03-03-03-06
- PSI-03-03-03-07

eTOM Reference

The operation is based on 1.3.9.2.2 process identifier from the eTOM.

5.3.3.4 TOD-03-03-04-View_All_Customer_Bills

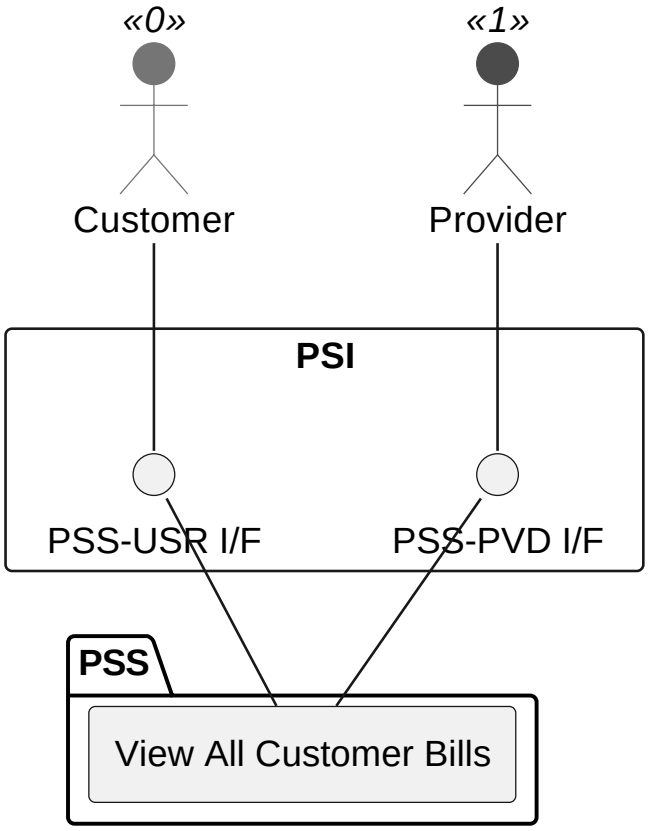

Figure 5.71: TOD-03-03-04: View All Customer Bills

Prerequisites

Customer bills exist in the PSS datastore.

Main operation

Gets all customer bills from the PSS via a standard interface specification. Filtering based at least on the state and the bill number can be applied in addition.

REST Endpoints

- GET /customerBillManagement/v1/customerBill

Post Conditions

All (filtered) customer bills which the customer or the provider can read are successfully returned to be viewed.

Applicable Requirements

- PSI-03-03-04-01
- PSI-03-03-04-02

eTOM Reference

The operation is based on 1.3.9.2.2 process identifier from the eTOM.

5.3.3.5 TOD-03-03-05-Withdraw_Customer_Bill

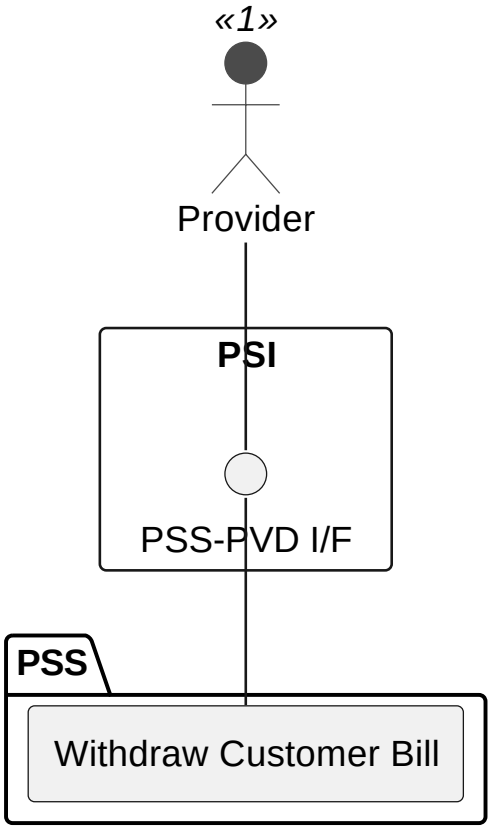

Figure 5.72: TOD-03-03-02: Withdraw Customer Bill

Prerequisites

The customer bill exists in the PSS datastore.

Main operation

The provider withdraws an existing customer bill in the PSS.

REST Endpoints

- DELETE /customerBillManagement/v1/customerBill/{id}

Post Conditions

The customer bill is successfully set to the state *widtdrawn* in the PSS.

Applicable Requirements

- PSI-03-03-05-01

eTOM Reference

None

5.3.4 TOD-03-04-Mission_Management

The Mission Management task enables the customer to plan and monitor mission related communication products. While this is an PSS-internal workflow, exposing it via PSI allows advanced and profession-specific software to make use of it as well.

The mission is defined by the customer using a name and a timeframe. They then define geographies (i.e., areas or points) where it will take place and optionally teams that operate there. Note that this can be done in any order and also be updated over time as the mission evolves. Based on these, the customer defines communication needs like internet access, site-2-site IP trunks or telephony. When they found a matching product (e.g. by issuing an Inquiry), a reference to it can be stored as well to enable access to further operations such as monitoring or change requests.

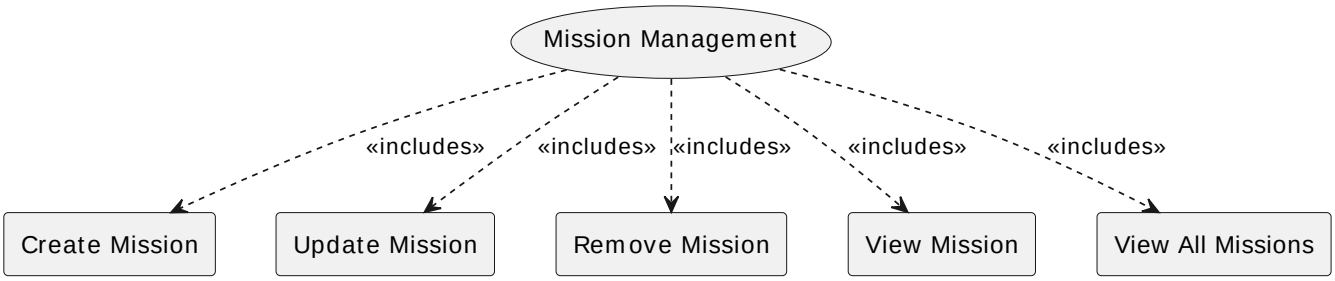

Figure 5.73: TOD-03-04: Mission Management

	Customer	Provider	Other PSS	Governance
Create Mission	✓			
Update Mission	✓			
Remove Mission	✓			
View Mission	✓			
View All Missions	✓			

Table 5.12: Mission Management Matrix.

eTOM Reference

The task is not based on the eTOM.

5.3.4.1 TOD-03-04-01-Create_Mission

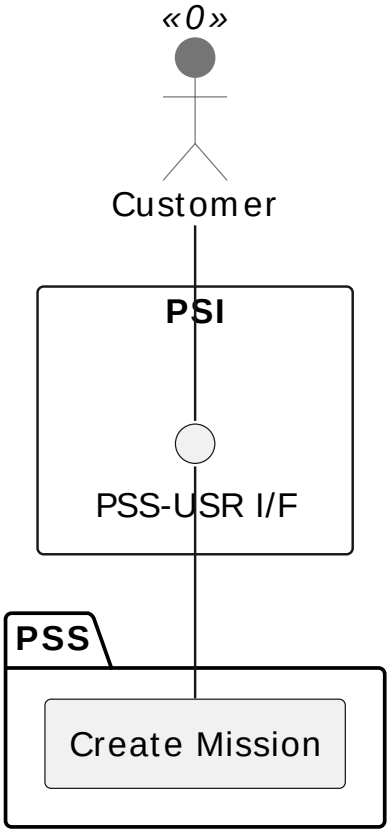

Figure 5.74: TOD-03-04-01: Create Mission

Prerequisites

The mission does not exist in the PSS datastore.

Main operation

Creates a new mission via a standard interface specification.

Some properties of a mission are:

- *name* - Short name of the target offering
- *description* - Description of the target offering
- *timeframe* - Start and end time of the mission
- *characteristic* - Additional properties as defined by the PSS (e.g. default values for assets)
- *place* - List of places where the mission is conducted
- *asset* - List of assets that are required for the mission, potentially linked to one or more
- places
- pre-owned resources/services
- ordered products

- related assets, e.g. required hardware for a service

Utilization periods of assets may be planned in various combinations, depending on the state of planning:

- Templates may include purely relative times (e.g. “one week after mission start”), which shall be translated to absolute times when the corresponding anchor is known.
- The implementation may choose to retain the offset to adjust the absolute time when the anchor is changed, and only delete it when the user opts to detach the times.

Note that teams may be expressed as **resources** and therefore may be managed as **assets** as well. This requires corresponding resource specifications (e.g. of a profession) in the catalog.

REST Endpoints

- POST /missionManagement/v1/mission

Post Conditions

The mission is successfully created in the PSS datastore.

Applicable Requirements

- PSI-03-04-01-01
- PSI-03-04-01-02
- PSI-03-04-01-03
- PSI-03-04-01-04
- PSI-03-04-01-05
- PSI-03-04-01-06
- PSI-03-04-01-07

eTOM Reference

The operation is not based on the eTOM.

5.3.4.2 TOD-03-04-02-Update_Mission

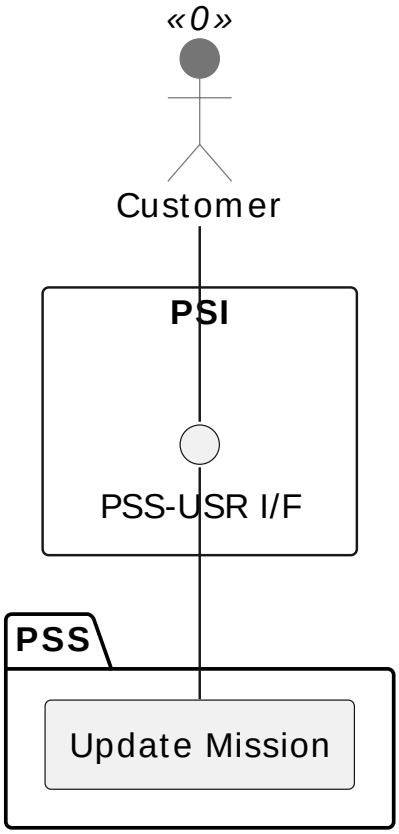

Figure 5.75: TOD-03-04-02: Update Mission

Prerequisites

The mission exists in the PSS datastore.

Main operation

Updates an existing mission via a standard interface specification. This may be done explicitly by the customer because of changes in places or requirements, but also by another subsystem that automatically assigns products, services or resources because of an order.

REST Endpoints

- PATCH /missionManagement/v1/mission/{id}

Post Conditions

The mission is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-03-04-02-01
- PSI-03-04-02-02
- PSI-03-04-02-03

eTOM Reference

The operation is not based on the eTOM.

5.3.4.3 TOD-03-04-03-Remove_Mission

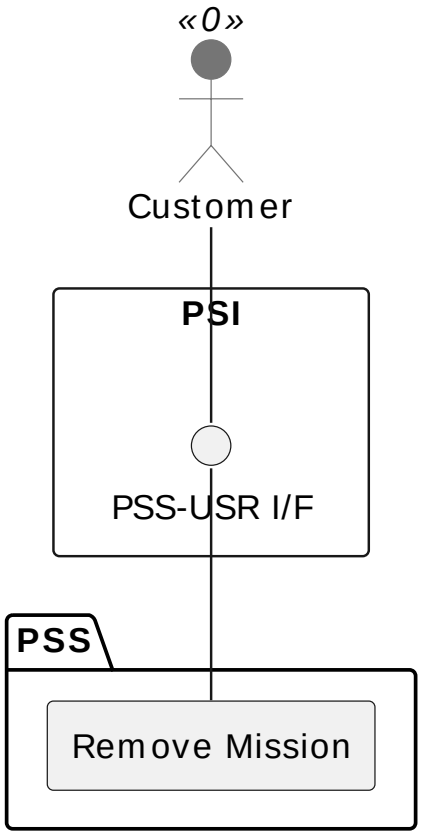

Figure 5.76: TOD-03-04-03: Remove Mission

Prerequisites

The mission exists in the PSS datastore.

Main operation

Removes a mission either by deleting it or indicating it is no longer valid, via a standard interface specification.

REST Endpoints

- DELETE /missionManagement/v1/mission/{id}

Post Conditions

The mission is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-03-04-03-01
- PSI-03-04-03-02

eTOM Reference

The operation is not based on the eTOM.

5.3.4.4 TOD-03-04-04-View_Mission

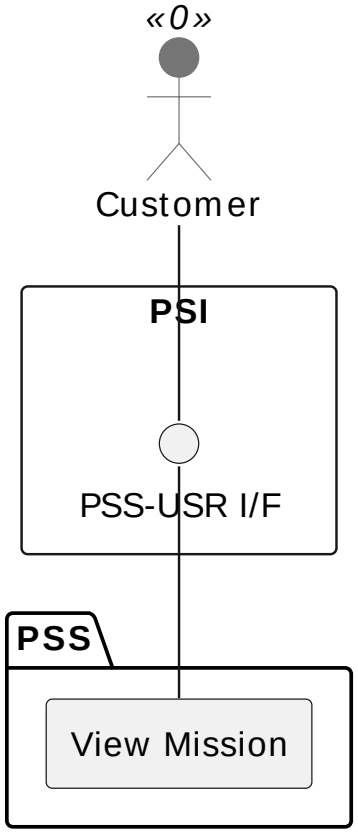

Figure 5.77: TOD-03-04-04: View Mission

Prerequisites

The mission exists in the PSS datastore.

Main operation

Gets a mission of the provider with a specific identifier via a standard interface specification.

REST Endpoints

- GET /missionManagement/v1/mission/{id}

Post Conditions

The mission is successfully returned to be viewed.

Applicable Requirements

- PSI-03-04-04-01
- PSI-03-04-04-02

eTOM Reference

The operation is not based on the eTOM.

5.3.4.5 TOD-03-04-05-View_All_Missions

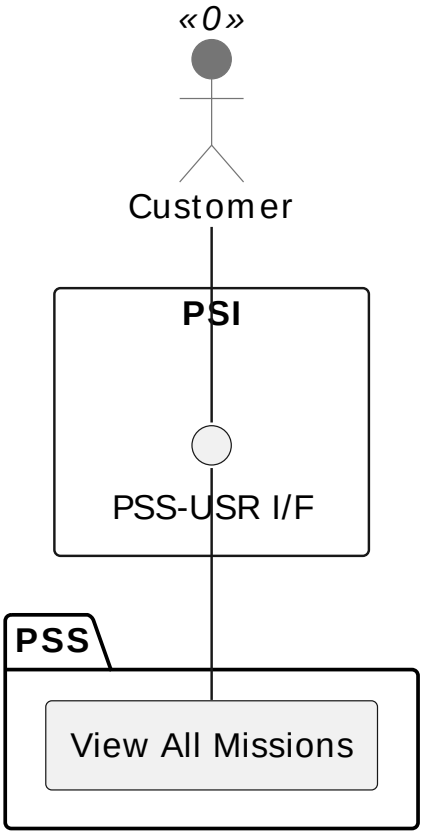

Figure 5.78: TOD-03-04-05: View All Missions

Prerequisites

Missions of the customer exist in the PSS datastore.

Main operation

Gets all missions of the customer via a standard interface specification.

REST Endpoints

- GET /missionManagement/v1/mission

Post Conditions

All missions of the customer are successfully returned to be viewed.

Applicable Requirements

- PSI-03-04-05-01

eTOM Reference

The operation is not based on the eTOM.

5.4 TOD-04-Template_Management

The category consists of tasks and operations related to managing templates for resources, services and products. These are created out of the JSON Schemas described in [PSI-ICD] and allow a quick-start in defining specifications for TOD-02-Product-Publishing.

5.4.1 TOD-04-01-Resource_Template_Management

The Resource Template Management task takes care of handling templates (descriptors) with static and dynamic fields containing all the necessary information that would help a provider register a resource specification to a PSS.

The governance of the PSS is responsible to create and maintain the resource templates and group them by resource type. For example, there can be resource templates for modems, antennas, satellite constellations, etc. The templates must be provider-agnostic, meaning all the providers should be able to use them to pool resource specifications.

The provider is able to request all the available resource templates and filter them by resource type or template name. Once they have the target template, they can update the default values proposed by the PSS's governance and define custom values. Tailored this way, the provider can register a resource specification to the PSS, while saving significant time in defining it from scratch.

The customer is also able to request available resource templates in order to be able to easily declare their own resources to the PSS and then use them as part of a customer inquiry.

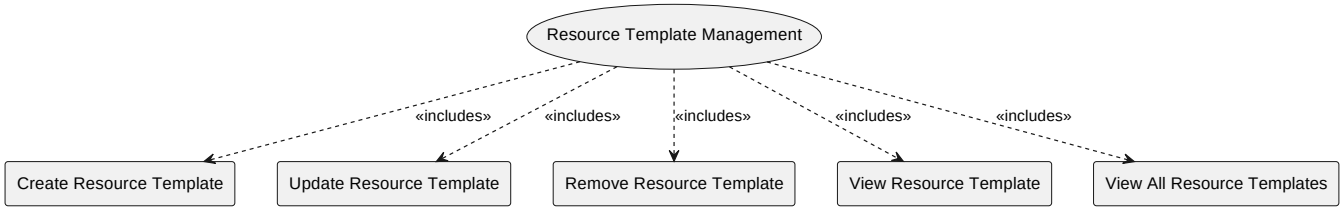

Figure 5.79: TOD-04-01: Resource Template Management

	Customer	Provider	Other PSS	Governance
Create Resource Template				✓
Update Resource Template				✓
Remove Resource Template				✓
View Resource Template	✓	✓		✓
View All Resource Templates	✓	✓		✓

Table 5.13: Resource Template Management Matrix.

Applicable Requirements

- PSI-04-01-00-01
- PSI-04-01-00-02

eTOM Reference

None

5.4.1.1 TOD-04-01-01-Create_Resource_Template

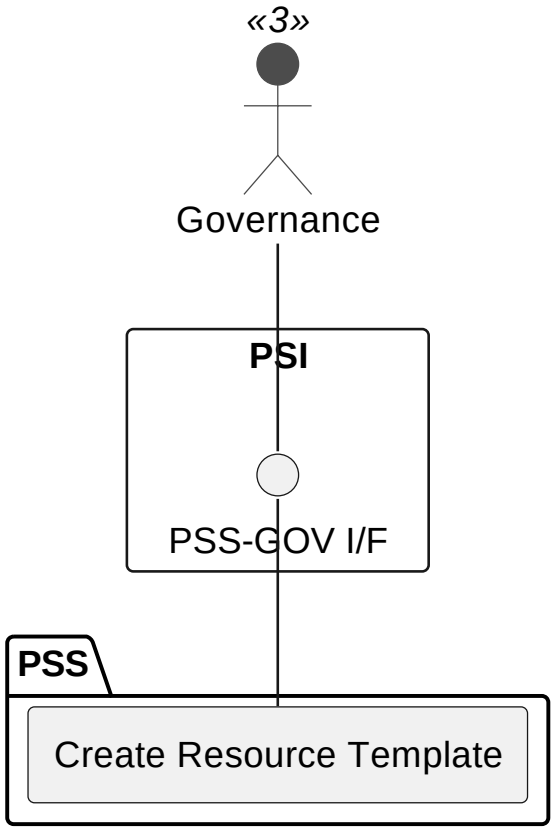

Figure 5.80: TOD-04-01-01: Create Resource Template

Prerequisites

The resource template does not exist in the PSS datastore.

Main operation

The governance creates a new resource template in the PSS with predefined field values for a resource specification of a given resource type (e.g. modem, antenna). The resource template can then be used by providers to register a resource specification by replacing the default values in the template with their resource specific values.

REST Endpoints

- POST /resourceCatalog/v1/resourceTemplate

Post Conditions

The resource template is successfully created in the PSS datastore.

Applicable Requirements

- PSI-04-01-01-01

- PSI-04-01-01-02

eTOM Reference

None

5.4.1.2 TOD-04-01-02-Update_Resource_Template

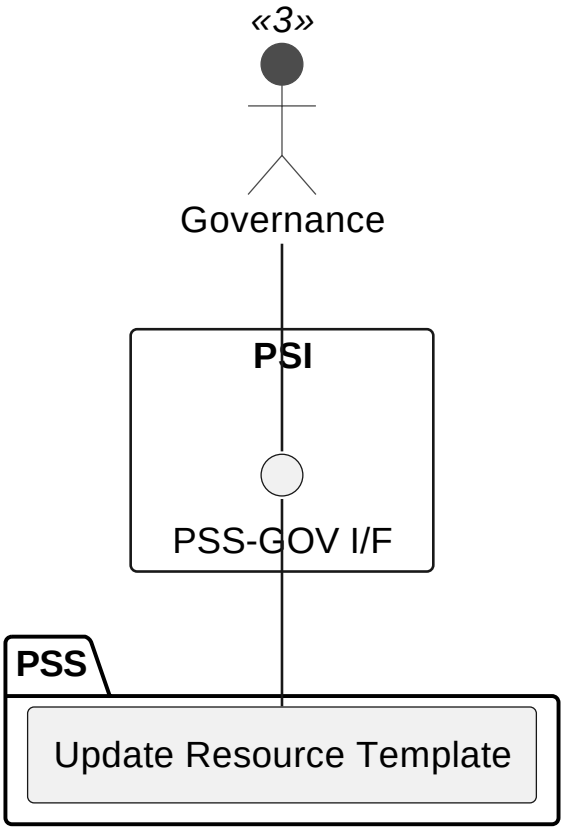

Figure 5.81: TOD-04-01-02: Update Resource Template

Prerequisites

The resource template exists in the PSS datastore.

Main operation

Updates an existing resource template via a standard interface specification available to the governance only.

REST Endpoints

- PATCH /resourceCatalog/v1/resourceTemplate/{id}

Post Conditions

The resource template is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-04-01-02-01

- PSI-04-01-02-02

eTOM Reference

None

5.4.1.3 TOD-04-01-03-Remove_Resource_Template

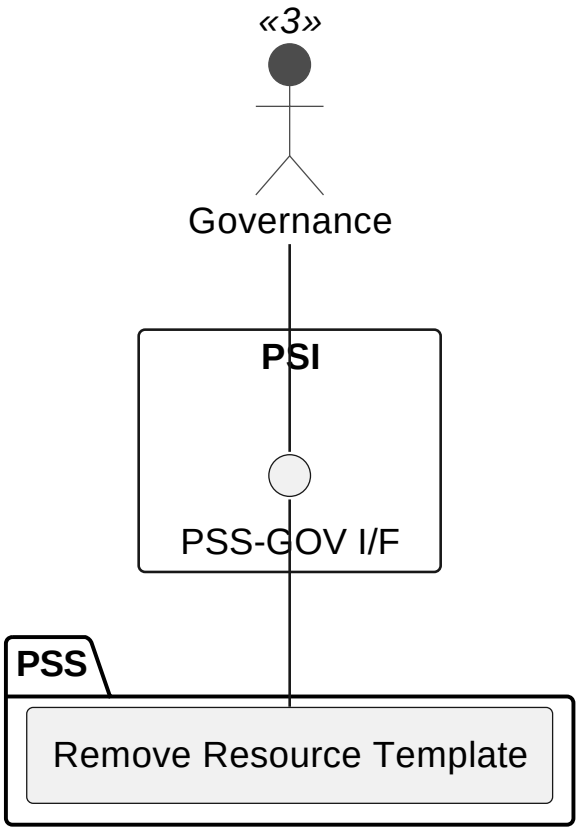

Figure 5.82: TOD-04-01-03: Remove Resource Template

Prerequisites

The resource template exists in the PSS datastore.

Main operation

Removes a resource template either by deleting it or indicating it is no longer valid, via a standard interface specification available to the governance only.

REST Endpoints

- DELETE /resourceCatalog/v1/resourceTemplate/{id}

Post Conditions

The resource template is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-04-01-03-01
- PSI-04-01-03-02

eTOM Reference

None

5.4.1.4 TOD-04-01-04-View_Resource_Template

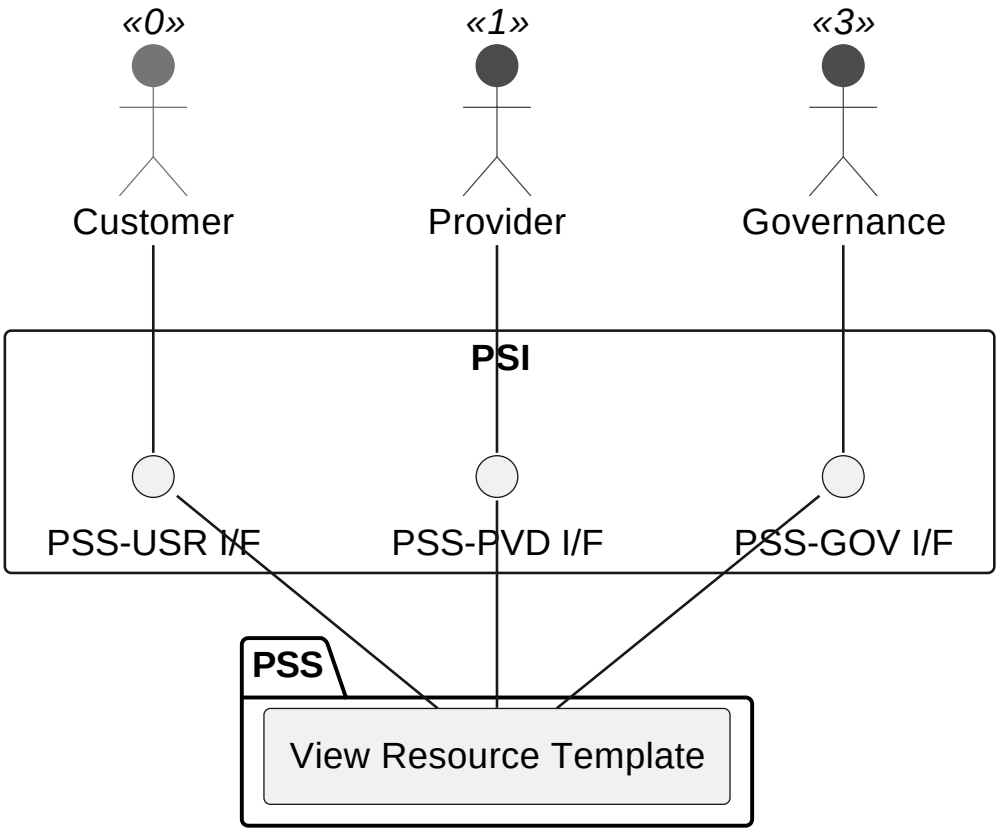

Figure 5.83: TOD-04-01-04: View Resource Template

Prerequisites

The resource template exists in the PSS datastore.

Main operation

Gets a resource template with a specific identifier via a standard interface specification.

REST Endpoints

- GET /resourceCatalog/v1/resourceTemplate/{id}

Post Conditions

The resource template is successfully returned to be viewed.

Applicable Requirements

- PSI-04-01-04-01
- PSI-04-01-04-02

eTOM Reference

None

5.4.1.5 TOD-04-01-05-View_All_Resource_Templates

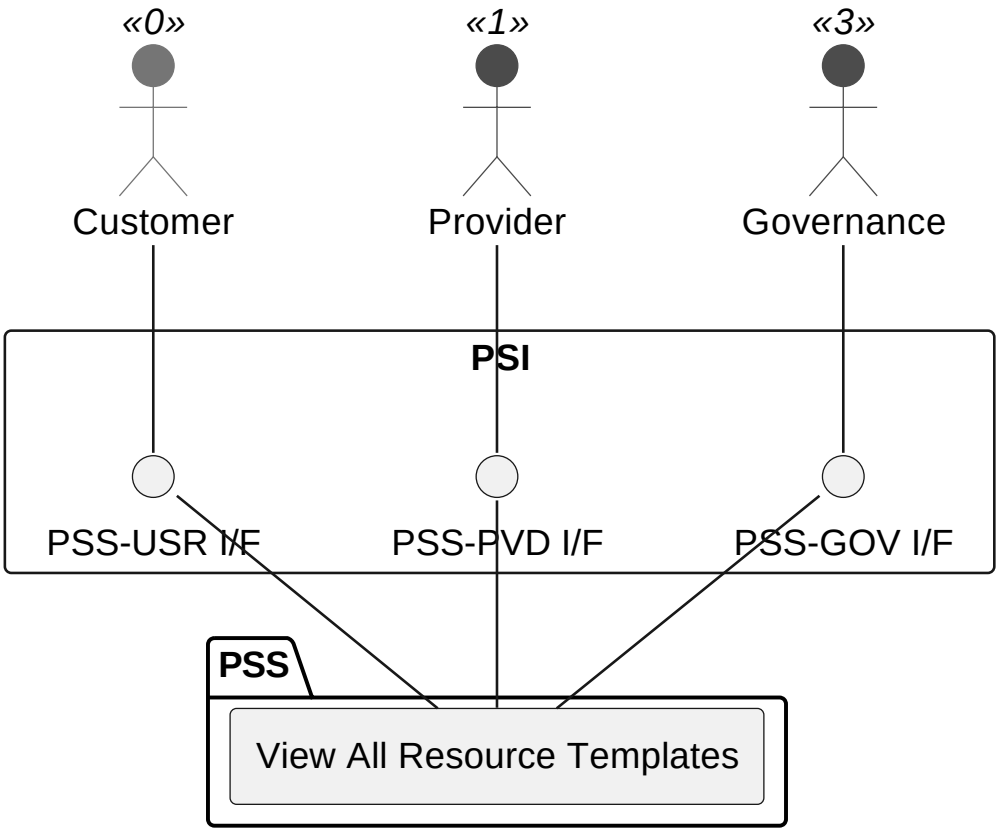

Figure 5.84: TOD-04-01-05: View All Resource Templates

Prerequisites

Resource templates exist in the PSS datastore.

Main operation

Gets all resource templates via a standard interface specification. These can be filtered by name and resource type.

REST Endpoints

- GET /resourceCatalog/v1/resourceTemplate

Post Conditions

All resource templates matching the filter criteria are successfully returned to be viewed.

Applicable Requirements

- PSI-04-01-05-01

eTOM Reference

None

5.4.2 TOD-04-02-Service_Template_Management

The Service Template Management task takes care of handling templates (descriptors) with the static and dynamic fields containing all the necessary information that would help a provider register a service specification to a PSS.

The governance of the PSS is responsible to create and maintain the service templates and group them by service type. For example, there can be service templates for internet access, telephony, site-to-site IP-Trunk, etc. The templates must be provider-agnostic, meaning all the providers should be able to use them to pool service specifications.

The provider is able to request all the available service templates and filter them by service type or template name. Once they have the target template, they can update the default values proposed by the PSS's governance and define custom values. Tailored this way, the provider can register a service specification to the PSS, while saving significant time in defining it from scratch.

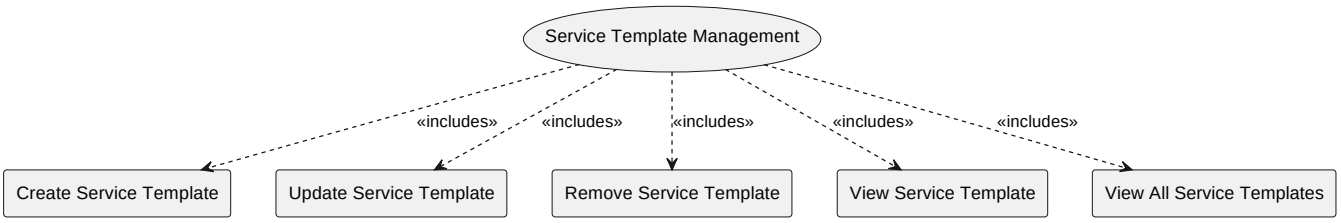

Figure 5.85: TOD-04-02: Service Template Management

	Customer	Provider	Other PSS	Governance
Create Service Template				✓
Update Service Template				✓
Remove Service Template				✓
View Service Template		✓		✓
View All Service Templates		✓		✓

Table 5.14: Service Template Management Matrix.

Applicable Requirements

- PSI-04-02-00-01
- PSI-04-02-00-02

eTOM Reference

None

5.4.2.1 TOD-04-02-01-Create_Service_Template

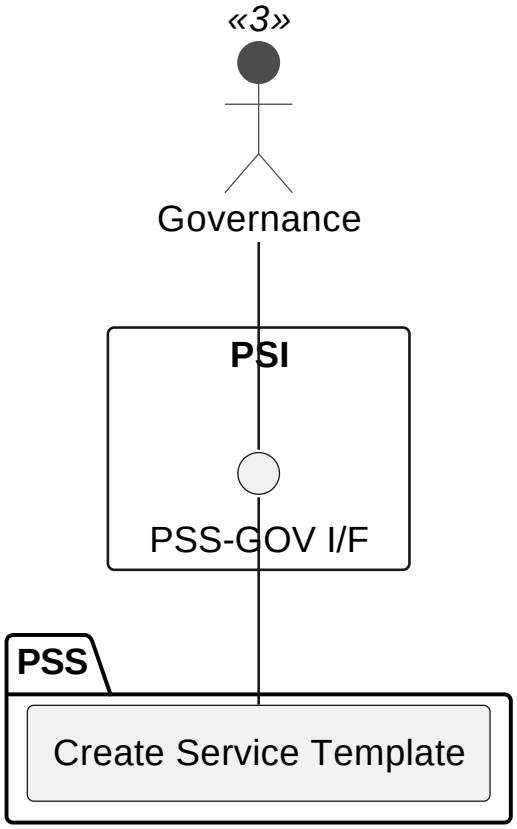

Figure 5.86: TOD-04-02-01: Create Service Template

Prerequisites

The service template does not exist in the PSS datastore.

Main operation

The governance creates a new service template in the PSS with predefined field values for a service specification of a given service type (e.g. internet access, telephony). The service template can then be used by providers to register a service specification by replacing the default values in the template with their service specific values.

REST Endpoints

- POST /serviceCatalog/v1/serviceTemplate

Post Conditions

The service template is successfully created in the PSS datastore.

Applicable Requirements

- PSI-04-02-01-01
- PSI-04-02-01-02

eTOM Reference

None

5.4.2.2 TOD-04-02-02-Update_Service_Template

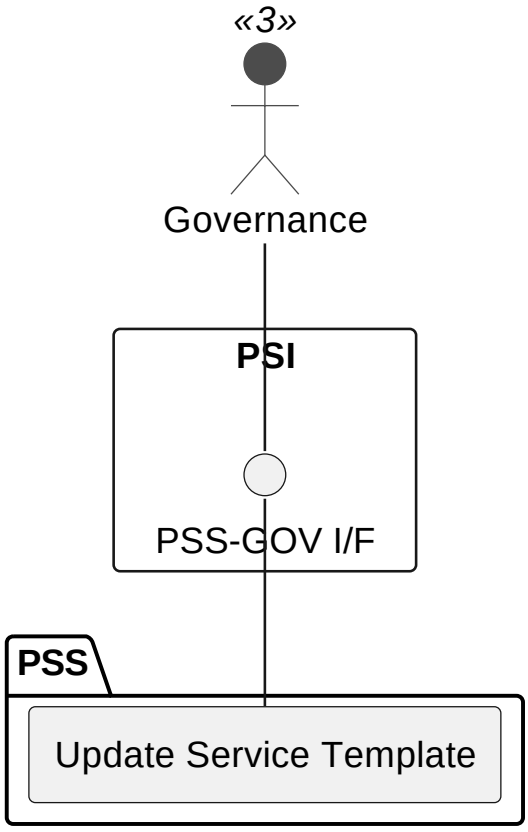

Figure 5.87: TOD-04-02-02: Update Service Template

Prerequisites

The service template exists in the PSS datastore.

Main operation

Updates an existing service template via a standard interface specification available to the governance only.

REST Endpoints

- PATCH /serviceCatalog/v1/serviceTemplate/{id}

Post Conditions

The service template is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-04-02-02-01

eTOM Reference

None

5.4.2.3 TOD-04-02-03-Remove_Service_Template

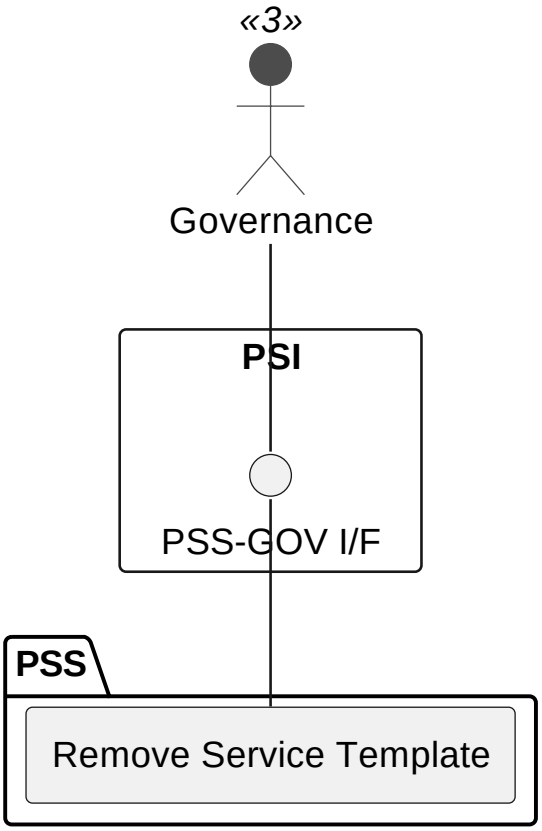

Figure 5.88: TOD-04-02-03: Remove Service Template

Prerequisites

The service template exists in the PSS datastore.

Main operation

Removes a service template either by deleting it or indicating it is no longer valid, via a standard interface specification available to the governance only.

REST Endpoints

- DELETE /serviceCatalog/v1/serviceTemplate/{id}

Post Conditions

The service template is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-04-02-03-01
- PSI-04-02-03-02

eTOM Reference

None

5.4.2.4 TOD-04-02-04-View_Service_Template

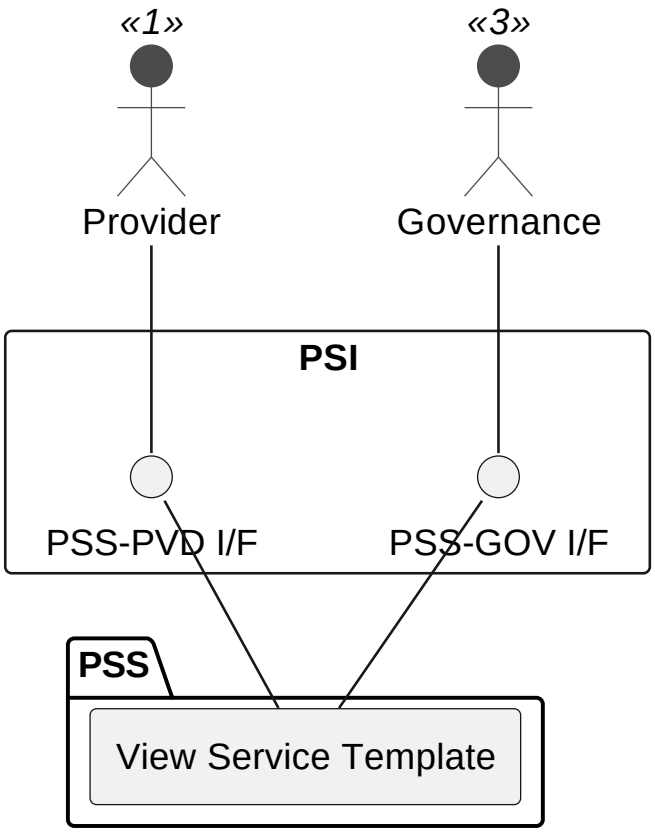

Figure 5.89: TOD-04-02-04: View Service Template

Prerequisites

The service template exists in the PSS datastore.

Main operation

Gets a service template with a specific identifier via a standard interface specification.

REST Endpoints

- GET /serviceCatalog/v1/serviceTemplate/{id}

Post Conditions

The service template is successfully returned to be viewed.

Applicable Requirements

- PSI-04-02-04-01
- PSI-04-02-04-02

eTOM Reference

None

5.4.2.5 TOD-04-02-05-View_All_Service_Templates

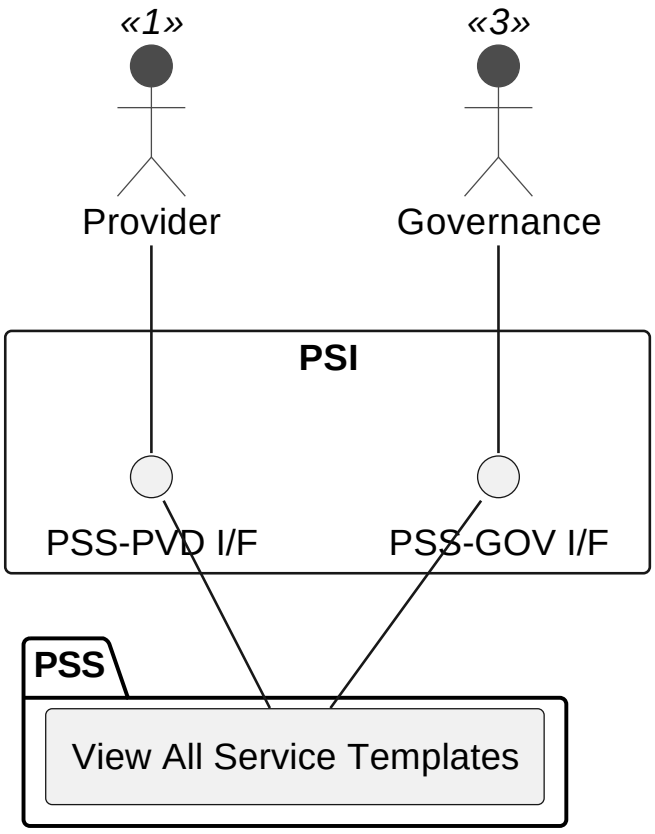

Figure 5.90: TOD-04-02-05: View All Service Templates

Prerequisites

Service templates exist in the PSS datastore.

Main operation

Gets all service template via a standard interface specification. These can be filtered by name and service type.

REST Endpoints

- GET /serviceCatalog/v1/serviceTemplate

Post Conditions

All service templates matching the filter criteria are successfully returned to be viewed.

Applicable Requirements

- PSI-04-02-05-01

eTOM Reference

None

5.4.3 TOD-04-03-Product_Template_Management

The Product Template Management task takes care of handling templates (descriptors) with the static and dynamic fields containing all the necessary information that would help a provider register a product specification to a PSS.

The governance of the PSS is responsible to create and maintain the product templates and group them by product type. For example, there can be product templates for internet access, telephony, site-to-site IP-Trunk, terminal, etc. The templates must be provider-agnostic, meaning all the providers should be able to use them to pool product specifications.

The provider is able to request all the available product templates and filter them by product type or template name. Once they have the target template, they can update the default values proposed by the PSS's governance and define custom values. Tailored this way, the provider can register a product specification to the PSS, while saving significant time in defining it from scratch.

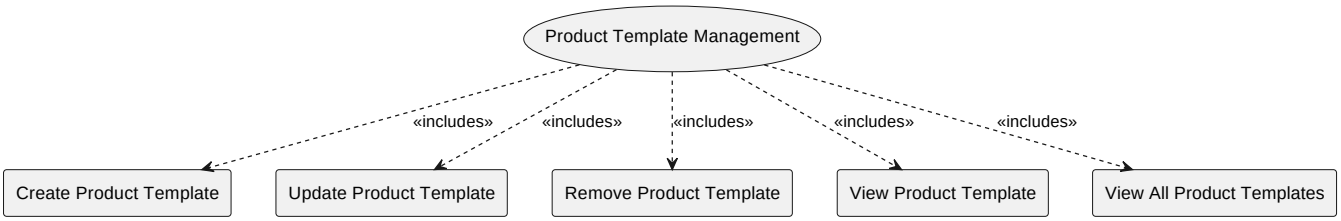

Figure 5.91: TOD-04-03: Product Template Management

	Customer	Provider	Other PSS	Governance
Create Product Template				✓
Update Product Template				✓
Remove Product Template				✓
View Product Template		✓		✓
View All Product Templates		✓		✓

Table 5.15: Product Template Management Matrix.

Applicable Requirements

- PSI-04-03-00-01
- PSI-04-03-00-02

eTOM Reference

None

5.4.3.1 TOD-04-03-01-Create_Product_Template

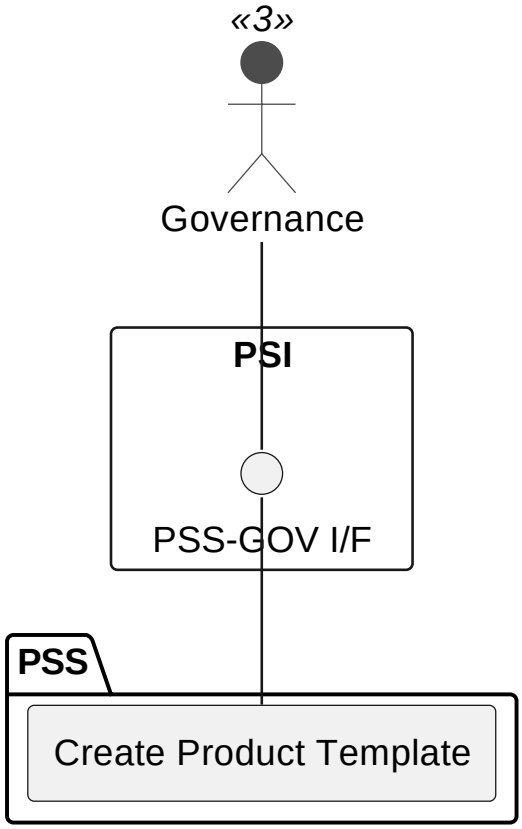

Figure 5.92: TOD-04-03-01: Create Product Template

Prerequisites

The product template does not exist in the PSS datastore.

Main operation

The governance creates a new product template in the PSS with predefined field values for a product specification of a given product type (e.g. internet access, telephony, terminal). The product template can then be used by providers to register a product specification by replacing the default values in the template with their product specific values.

REST Endpoints

- POST /productCatalog/v1/productTemplate

Post Conditions

The product template is successfully created in the PSS datastore.

Applicable Requirements

- PSI-04-03-01-01
- PSI-04-03-01-02

eTOM Reference

None

5.4.3.2 TOD-04-03-02-Update_Product_Template

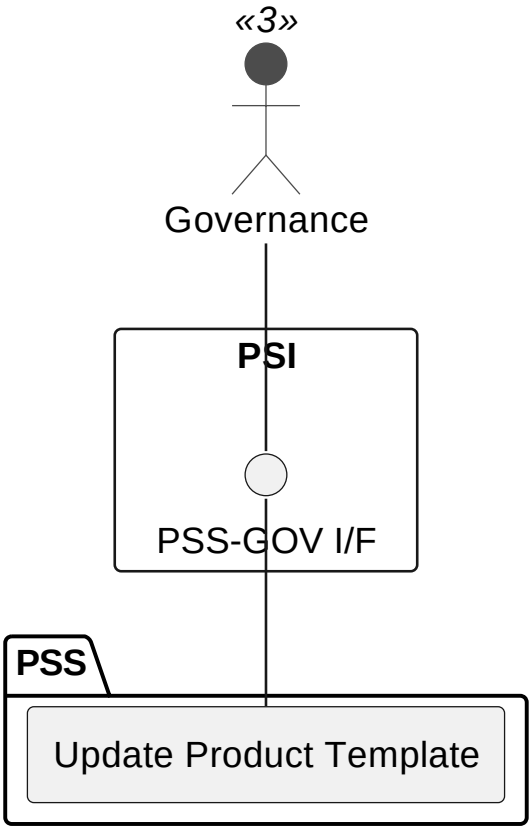

Figure 5.93: TOD-04-03-02: Update Product Template

Prerequisites

The product template exists in the PSS datastore.

Main operation

Updates an existing product template via a standard interface specification available to the governance only.

REST Endpoints

- PATCH /productCatalog/v1/productTemplate/{id}

Post Conditions

The product template is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-04-03-02-01

eTOM Reference

None

5.4.3.3 TOD-04-03-03-Remove_Product_Template

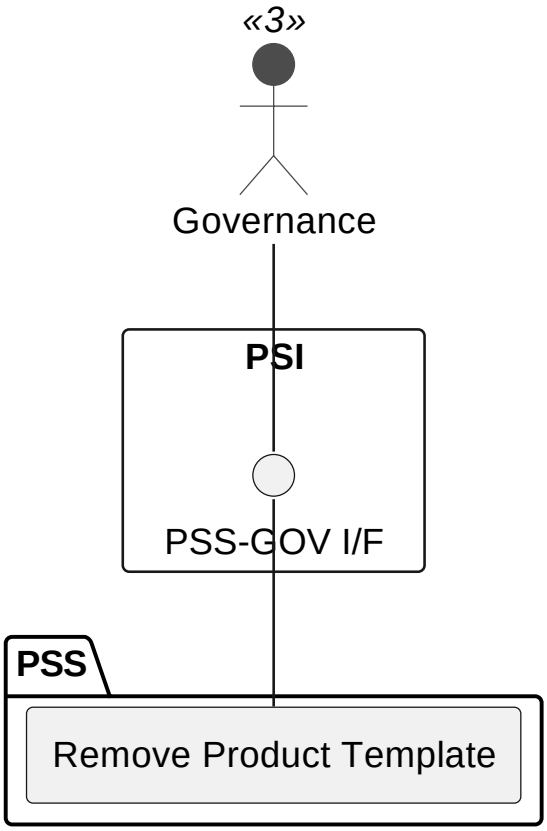

Figure 5.94: TOD-04-03-03: Remove Product Template

Prerequisites

The product template exists in the PSS datastore.

Main operation

Removes a product template either by deleting it or indicating it is no longer valid, via a standard interface specification available to the governance only.

REST Endpoints

- DELETE /productCatalog/v1/productTemplate/{id}

Post Conditions

The product template is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-04-03-03-01
- PSI-04-03-03-02

eTOM Reference

None

5.4.3.4 TOD-04-03-04-View_Product_Template

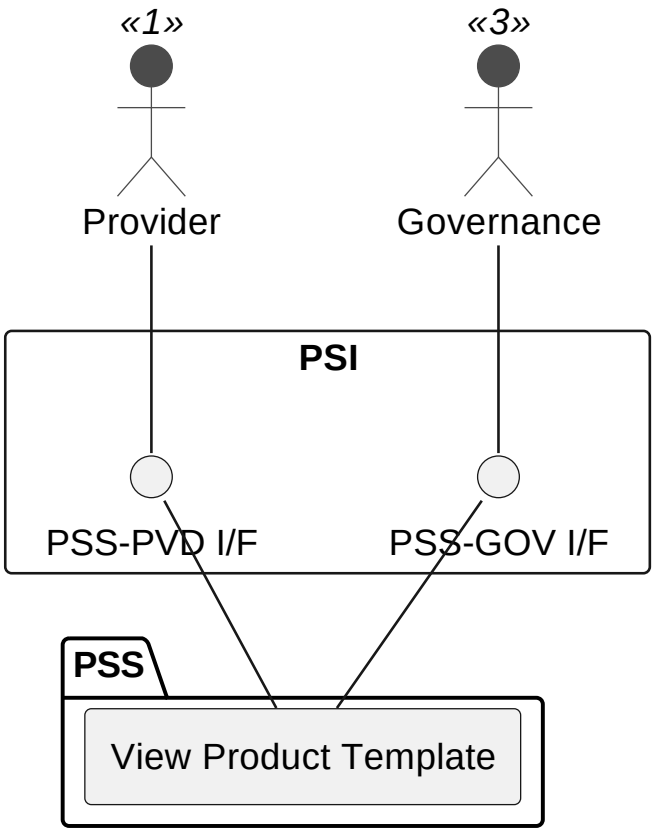

Figure 5.95: TOD-04-03-04: View Product Template

Prerequisites

The product template exists in the PSS datastore.

Main operation

Gets a product template with a specific identifier via a standard interface specification.

REST Endpoints

- GET /productCatalog/v1/productTemplate/{id}

Post Conditions

The product template is successfully returned to be viewed.

Applicable Requirements

- PSI-04-03-04-01
- PSI-04-03-04-03

eTOM Reference

None

5.4.3.5 TOD-04-03-05-View_All_Product_Templates

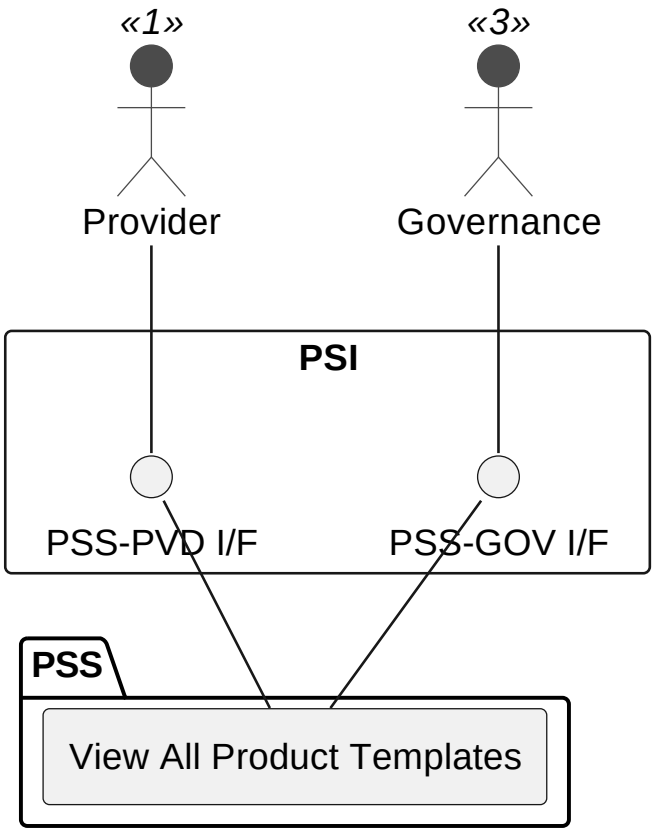

Figure 5.96: TOD-04-03-05: View All Product Templates

Prerequisites

Product templates exist in the PSS datastore.

Main operation

Gets all product templates via a standard interface specification. These can be filtered by name and product type.

REST Endpoints

- GET /productCatalog/v1/productTemplate

Post Conditions

All product templates matching the filter criteria are successfully returned to be viewed.

Applicable Requirements

- PSI-04-03-05-01

eTOM Reference

None

5.5 TOD-05-Inventory_Management

The category consists of tasks and operations related to managing the inventory of products, services and resources. While services (and products containing them) are created in the order process, resources can be held in a warehouse and checked for availability via the stock management.

5.5.1 TOD-05-01-Resource_Inventory_Management

The Resource Inventory Management task takes care of the maintenance of resources in the PSS, brought in by providers or customers themselves.

Resources are created based on their specifications and reflect the actual characteristics of an existing instance. There are two different scenarios to do this:

- For *on-demand* resources, the entity may or may not exist in the provider system. They are unknown to the PSS and are only queried via the Stock Management API (cf. TOD-05-04). Only upon order acceptance (or even afterwards) it is created in the PSS and "lives" for the requested period of time.
- *Committed* resources are created in the PSS beforehand. This allows the PSS to internally check the availability in any given timespan. Every order (or reservation, if supported) will create a sub-entity that is related to the master-entity and handled like above.

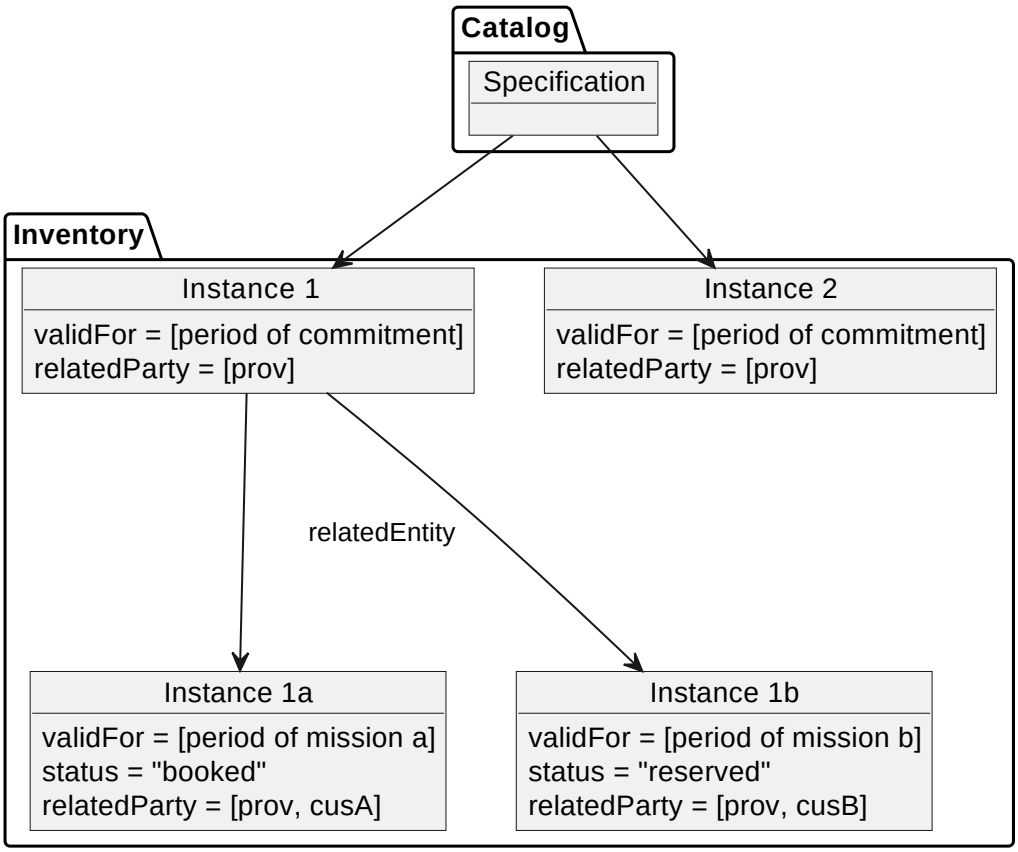

Figure 5.97: TOD-05-01: Tree of Resources.

Depending on the implementation, the resource can be further subdivided, e.g. if it is shared with another user or resold by a service provider. If the resource instance is offered to customers, it has to be wrapped in a product instance resembling their corresponding specifications.

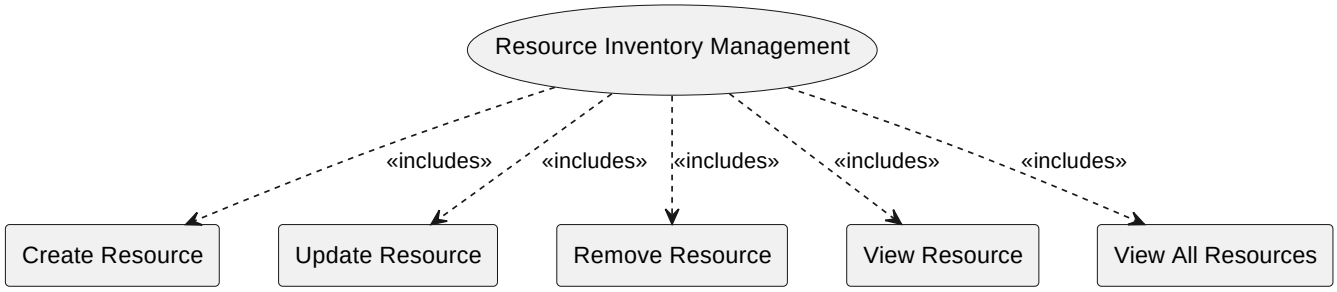

Figure 5.98: TOD-05-01: Resource Inventory Management

	Customer	Provider	Other PSS	Governance
Create Resource	✓	✓		
Update Resource	✓	✓		
Remove Resource	✓	✓		
View Resource	✓	✓		✓
View All Resource	✓	✓		✓

Table 5.16: Resource Inventory Management Matrix.

Applicable Requirements

- PSI-05-01-00-01

eTOM Reference

The task is based on the 1.5.4.5 process identifier from the eTOM.

5.5.1.1 TOD-05-01-01-Create_Resource

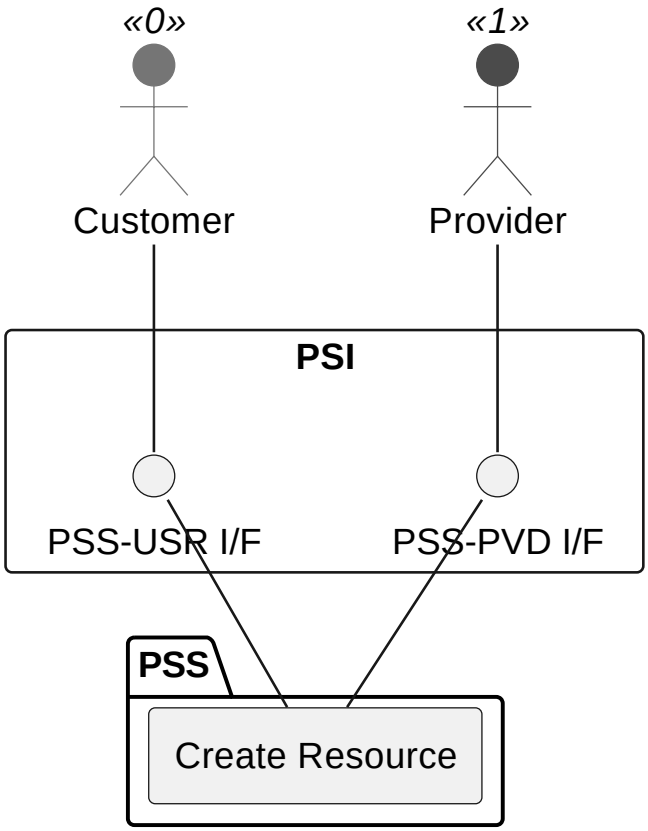

Figure 5.99: TOD-05-01-01: Create Resource

Prerequisites

The resource does not exist in the PSS datastore.

Main operation

Creates a new resource instance with its characteristics via a standard interface based on a specification.

Some properties of a resource are:

- *name* - Short name of the target resource
- *description* - Description of the target resource
- *category* - Category (resource type) of the target resource like terminals, bandwidth, etc.
- *resourceSpecification* - The specification this resource is based on
- *resourceStatus* - The status of the resources (e.g. available or reserved)
- *type*/*schemaLocation* - Name and reference to the JSON Schema defining the type of this resource.
- *resourceCharacteristic* - List of characteristics i.e. technicals of the resource such as frequency band, Tx/Rx frequency, etc.
- *relatedParty* - References to the provider owning the resource and the customer that booked it (if applicable)

- *startOperatingDate* / *endOperatingDate* - Operational time period of the resource

REST Endpoints

- POST /resourceInventory/v1/resource

Post Conditions

The resource is successfully created in the PSS datastore.

Applicable Requirements

- PSI-05-01-01-01
- PSI-05-01-01-02
- PSI-05-01-01-03
- PSI-05-01-01-04
- PSI-05-01-01-05
- PSI-05-01-01-06

eTOM Reference

The operation is based on the 1.5.4.5.1 process identifier from the eTOM.

5.5.1.2 TOD-05-01-02-Update_Resource

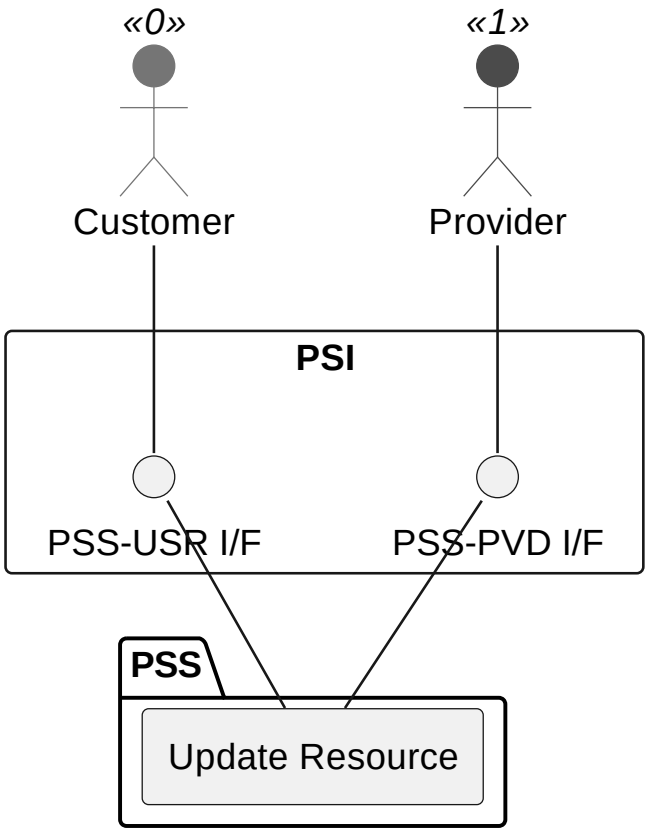

Figure 5.100: TOD-05-01-02: Update Resource

Prerequisites

The resource exists in the PSS datastore.

Main operation

Updates an existing resource via a standard interface.

REST Endpoints

- PATCH /resourceInventory/v1/resource/{id}

Post Conditions

The resource is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-05-01-02-01

eTOM Reference

The operation is based on the 1.5.4.5.1 process identifier from the eTOM.

5.5.1.3 TOD-05-01-03-Remove_Resource

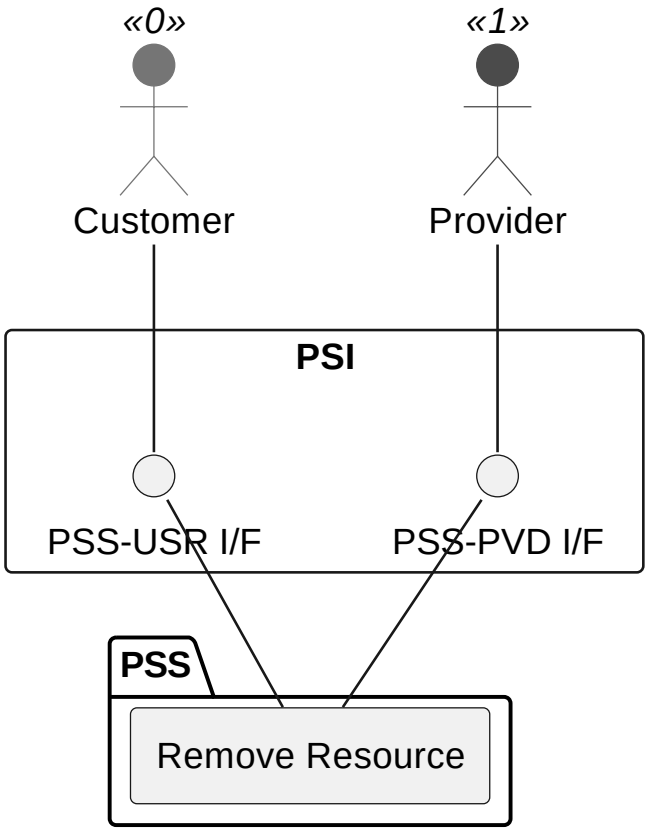

Figure 5.101: TOD-05-01-03: Remove Resource

Prerequisites

The resource exists in the PSS datastore.

Main operation

Removes a resource either by deleting it or indicating it is no longer valid, via a standard interface.

REST Endpoints

- DELETE /resourceInventory/v1/resource/{id}

Post Conditions

The resource is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-05-01-03-01
- PSI-05-01-03-02

eTOM Reference

The operation is based on the 1.5.4.5.1 process identifier from the eTOM.

5.5.1.4 TOD-05-01-04-View_Resource

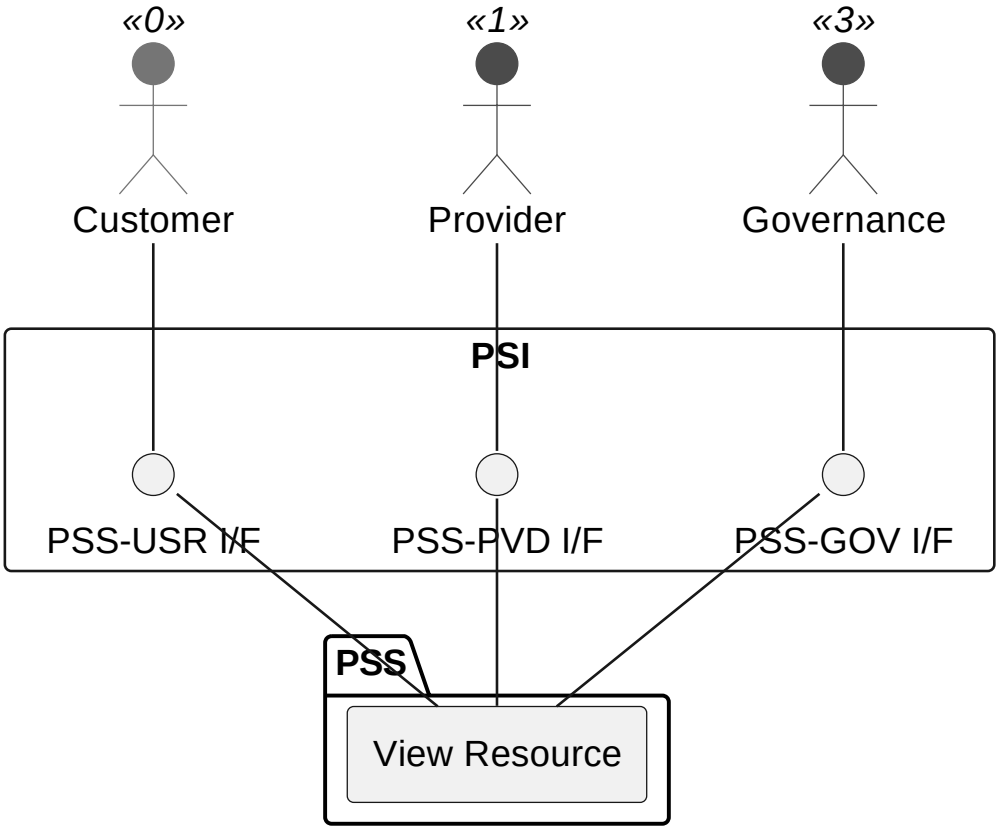

Figure 5.102: TOD-05-01-04: View Resource

Prerequisites

The resource exists in the PSS datastore.

Main operation

Gets a resource with a specific identifier via a standard interface. Customers can view their own declared resources and those they booked.

REST Endpoints

- GET /resourceInventory/v1/resource/{id}

Post Conditions

The resource is successfully returned to be viewed.

Applicable Requirements

- PSI-05-01-04-01
- PSI-05-01-04-02

eTOM Reference

The operation is based on the 1.5.4.5 process identifier from the eTOM.

5.5.1.5 TOD-05-01-05-View_All_Resources

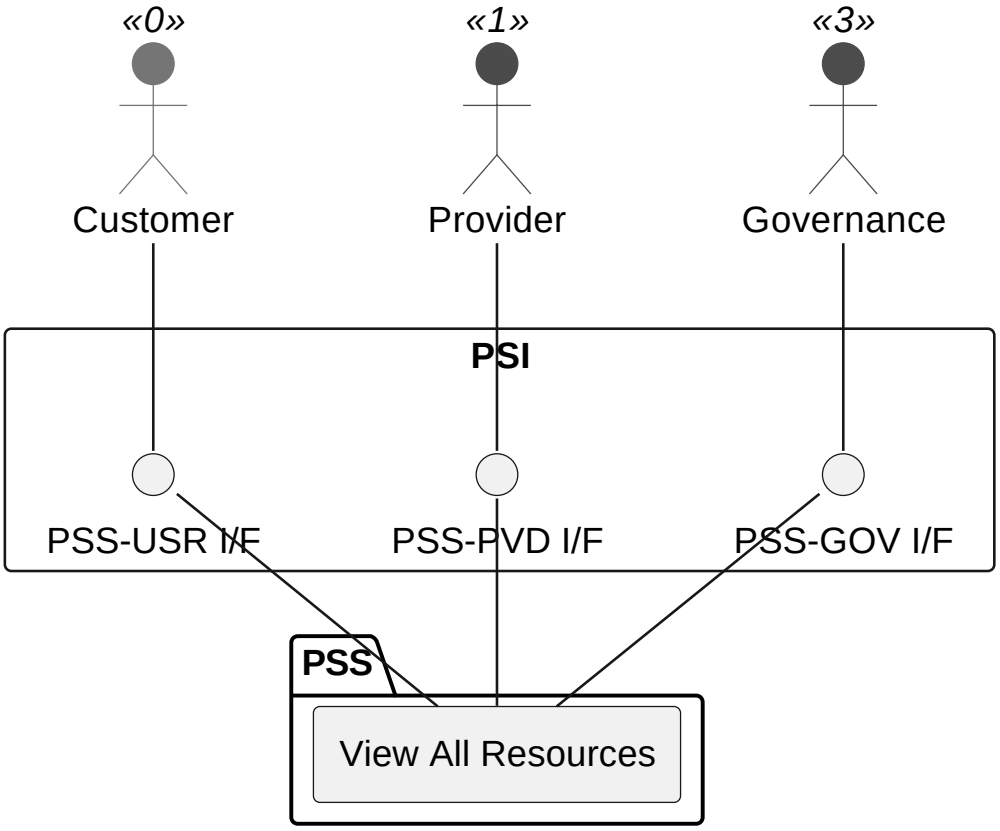

Figure 5.103: TOD-05-01-05: View All Resources

Prerequisites

Resources exist in the PSS datastore.

Main operation

Gets all visible resources previously registered in the PSS via a standard interface. These can be filtered at least by resource type. Customers can view their own declared resources and those they booked.

REST Endpoints

- GET /resourceInventory/v1/resource

Post Conditions

All visible resources are successfully returned to be viewed.

Applicable Requirements

- PSI-05-01-05-01
- PSI-05-01-05-02

eTOM Reference

The operation is based on the 1.5.4.5 process identifier from the eTOM.

5.5.2 TOD-05-02-Service_Inventory_Management

The Service Inventory Management task takes care of the maintenance of services in the PSS, brought in by providers.

The provider creates services as part of order fulfilment. Each instance represents a concrete implementation of a service specification for a customer and is bound to a product instance. They can be also be modified or deleted over time when the customer issues change requests.

The customer can see all their booked services to review their characteristics.

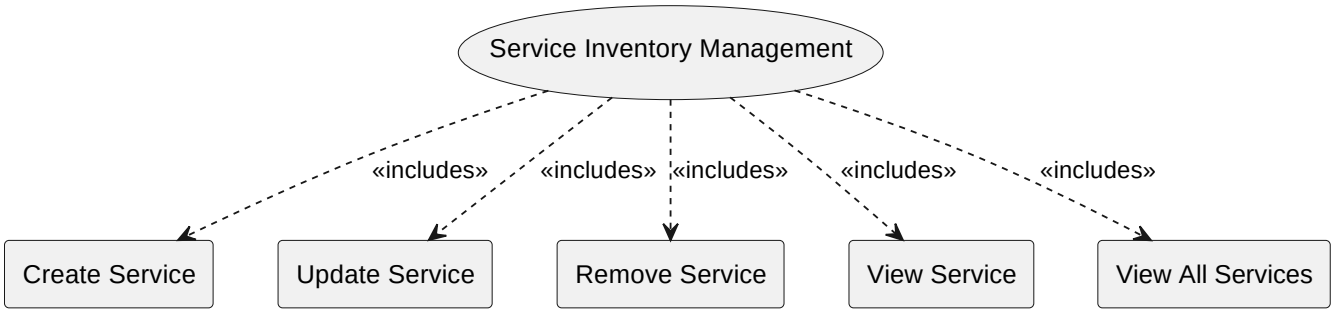

Figure 5.104: TOD-05-02: Service Inventory Management

	Customer	Provider	Other PSS	Governance
Create Service		✓		
Update Service		✓		
Remove Service		✓		
View Service	✓	✓		✓
View All Services	✓	✓		✓

Table 5.17: Service Inventory Management Matrix.

Applicable Requirements

- PSI-05-02-00-01

eTOM Reference

The task is based on the 1.4.4.1 process identifier from the eTOM.

5.5.2.1 TOD-05-02-01-Create_Service

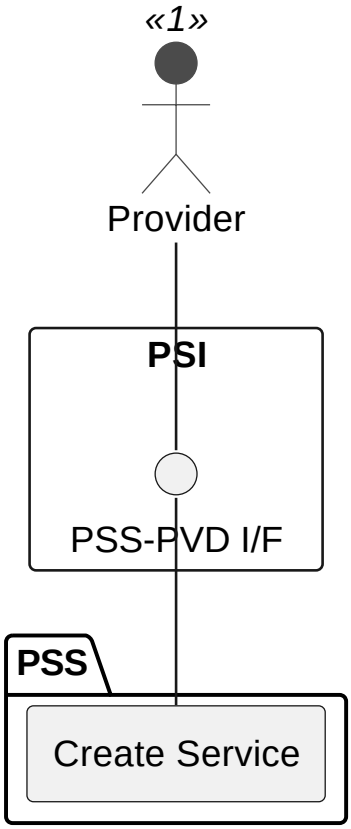

Figure 5.105: TOD-05-02-01: Create Service

Prerequisites

The service does not exist in the PSS datastore.

Main operation

Creates a new service instance with its characteristics and references to resources via a standard interface based on a specification.

Some properties of a service are:

- *name* - Short name of the target service
- *description* - Description of the target service
- *category* - Category (service type) of the target service like internet access, telephony, IP-Trunk, etc.
- *supportingResource* - List of resources that are required to realise the target service
- *type*/*schemaLocation* - Name and reference to the JSON Schema defining the type of this service.
- *serviceCharacteristic* - List of characteristics of the target service such as forwardCIR, returnCIR, etc.
- *relatedParty* - References to the provider owning the resource and the customer that booked it
- *state* - Current status of the service (e.g. inactive, active, terminated, etc.)

- *startDate* / *endDate* - Time period of validity of the service

REST Endpoints

- POST /serviceInventory/v1/service

Post Conditions

The service is successfully created in the PSS datastore.

Applicable Requirements

- PSI-05-02-01-01
- PSI-05-02-01-02

eTOM Reference

The operation is based on the 1.4.4.1 process identifier from the eTOM.

5.5.2.2 TOD-05-02-02-Update_Service

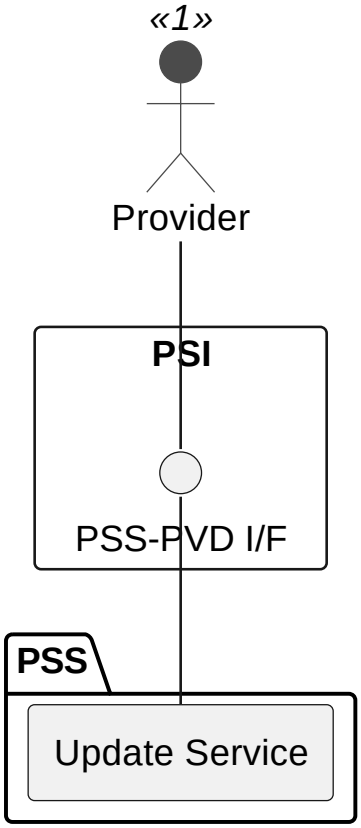

Figure 5.106: TOD-05-02-02: Update Service

Prerequisites

The service exists in the PSS datastore.

Main operation

Updates an existing service via a standard interface.

REST Endpoints

- PATCH /serviceInventory/v1/service/{id}

Post Conditions

The service is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-05-02-02-01

eTOM Reference

The operation is based on the 1.4.4.1 process identifier from the eTOM.

5.5.2.3 TOD-05-02-03-Remove_Service

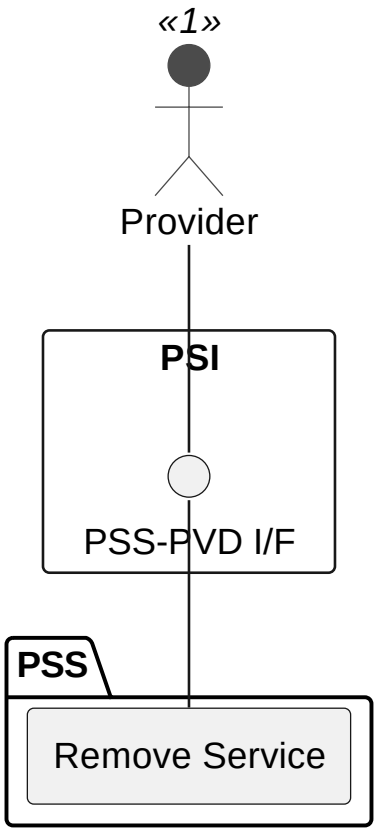

Figure 5.107: TOD-05-02-03: Remove Service

Prerequisites

The service exists in the PSS datastore.

Main operation

Removes a service either by deleting it or indicating it is no longer valid, via a standard interface.

REST Endpoints

- DELETE /serviceInventory/v1/service/{id}

Post Conditions

The service is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-05-02-03-01
- PSI-05-02-03-02

eTOM Reference

The operation is based on the 1.4.4.1.1 process identifier from the eTOM.

5.5.2.4 TOD-05-02-04-View_Service

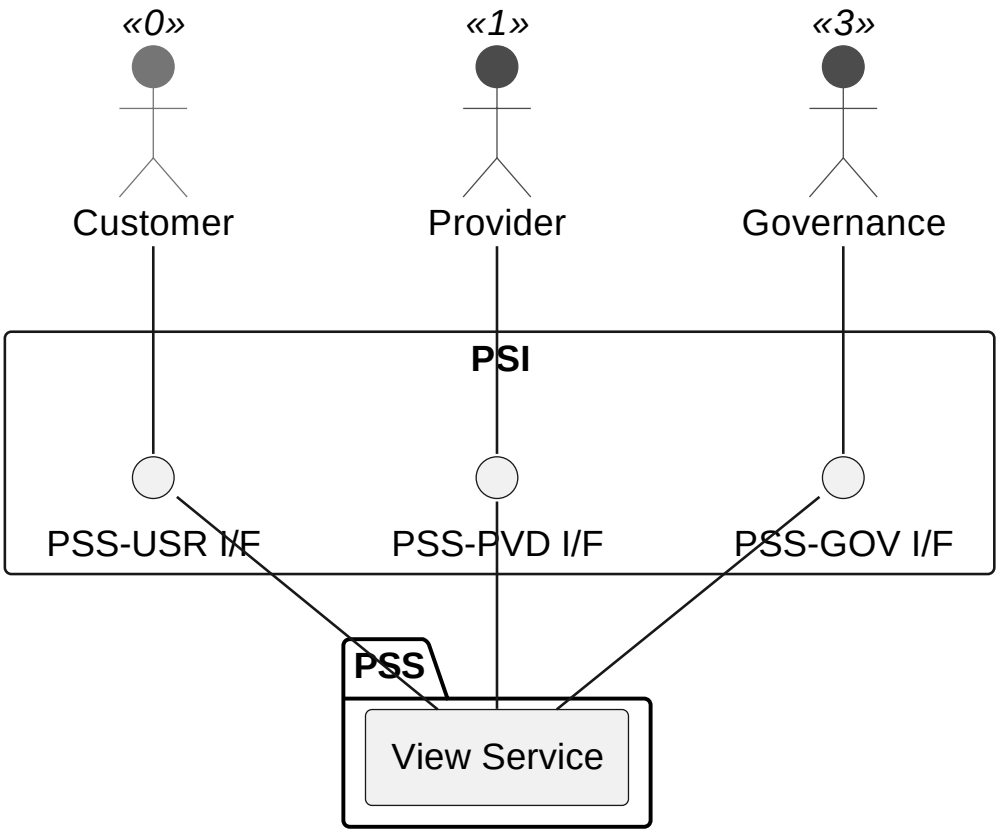

Figure 5.108: TOD-05-02-04: View Service

Prerequisites

The service exists in the PSS datastore.

Main operation

Gets a service with a specific identifier via a standard interface. Customers can only see the services they booked.

REST Endpoints

- GET /serviceInventory/v1/service/{id}

Post Conditions

The service is successfully returned to be viewed.

Applicable Requirements

- PSI-05-02-04-01
- PSI-05-02-04-02

eTOM Reference

The operation is based on the 1.4.4.1 process identifier from the eTOM.

5.5.2.5 TOD-05-02-05-View_All_Services

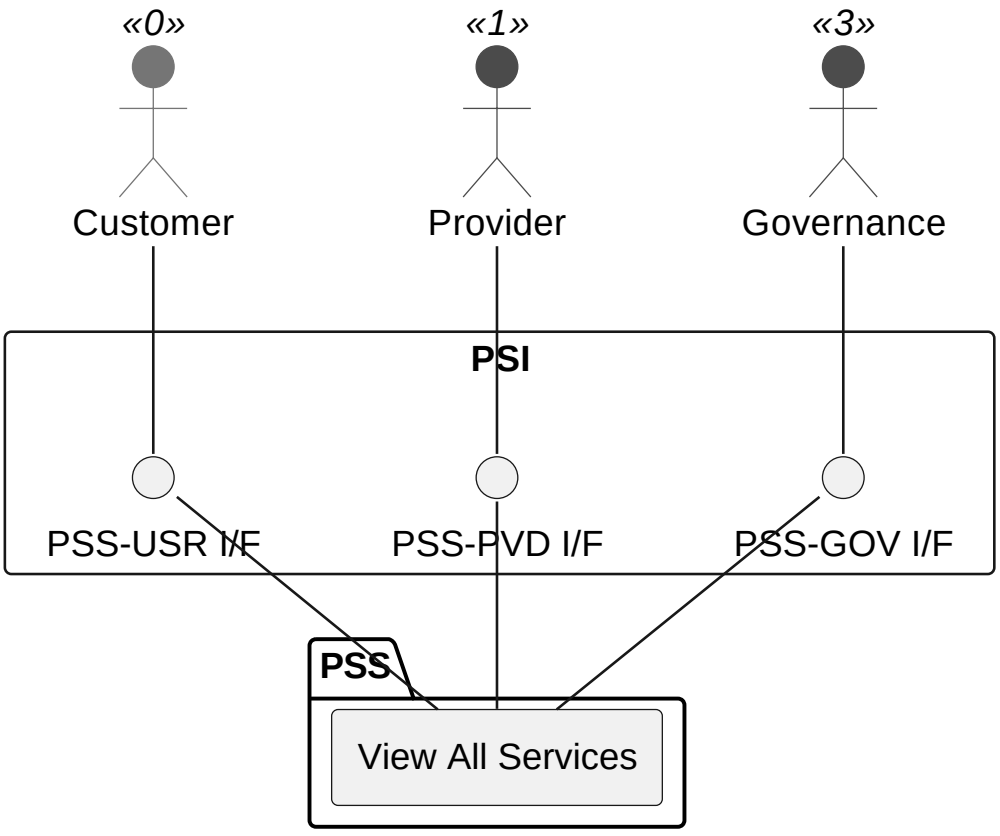

Figure 5.109: TOD-05-02-05: View All Services

Prerequisites

Services exist in the PSS datastore.

Main operation

Gets all services via a standard interface. These can be filtered at least by service type. Customers can only see the services they booked.

REST Endpoints

- GET /serviceInventory/v1/service

Post Conditions

All visible services are successfully returned to be viewed.

Applicable Requirements

- PSI-05-02-05-01
- PSI-05-02-05-02

eTOM Reference

The operation is based on the 1.4.4.1 process identifier from the eTOM.

5.5.3 TOD-05-03-Product_Inventory_Management

The Product Inventory Management task takes care of the maintenance of products in the PSS, brought in by providers.

There are two different scenarios to do this:

- A product is created if an order of the corresponding service or resource is fulfilled.
- As stated in TOD-05-01, a product can be stocked and therefore created any time along with its resource.

The PSS manages a priority for each active product instance, which stems from the customer profile or their inquiry. It is to be stored in a characteristic and can be used for conflict resolution (e.g. pre-emption) when higher priority requests can otherwise not be fulfilled.

They can be also be modified or deleted over time when the customer issues change requests. The customer can see all their booked products to review their characteristics.

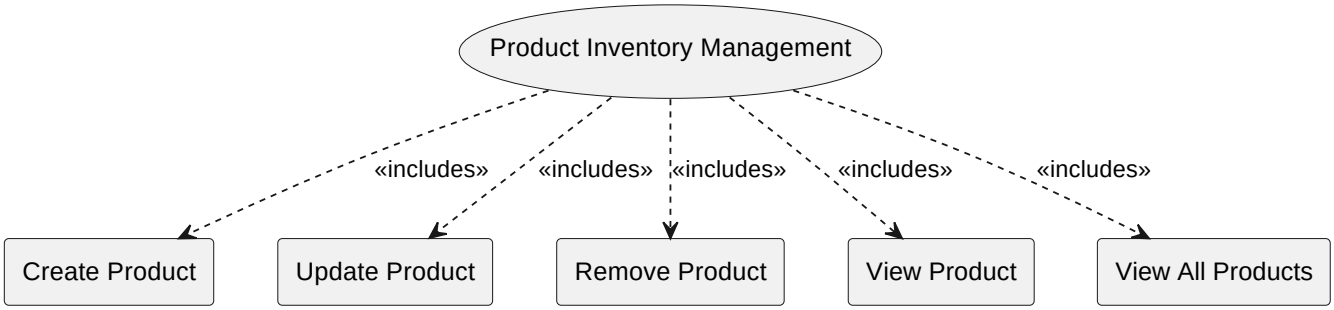

Figure 5.110: TOD-05-03: Product Inventory Management

	Customer	Provider	Other PSS	Governance
Create Product		✓		
Update Product		✓		
Remove Product		✓		
View Product	✓	✓		✓
View All Products	✓	✓		✓

Table 5.18: Product Inventory Management Matrix.

Applicable Requirements

- PSI-05-03-00-01
- PSI-05-03-00-02

eTOM Reference

The task is based on the 1.2.11 process identifier from the eTOM.

5.5.3.1 TOD-05-03-01-Create_Product

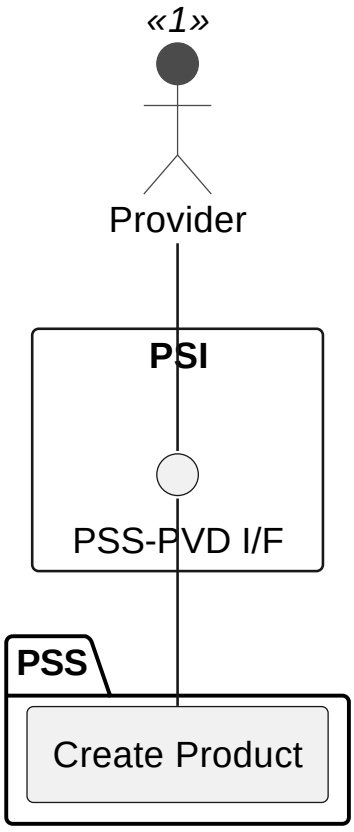

Figure 5.111: TOD-05-03-01: Create Product

Prerequisites

The product does not exist in the PSS datastore.

Main operation

Creates a new product with its characteristics and references to resource/services via a standard interface.

> Note: It is possible to create a product which will be available in the future by setting the *startDate* property with a future time reference.

Some properties of a product are:

- *name* - Short name of the target product
- *description* - Description of the target product
- *product* - If the product is a bundle of multiple products, a list of the related products
- *realizingResource* - List of resources that are required to realise the target product
- *realizingService* - List of services that are required to realise the target product
- *type*/*schemaLocation* - Name and reference to the JSON Schema defining the type of this product.
- *productCharacteristic* - List of distinctive features of the target product such as 'networkUptime', 'dataAllowance', etc.
- *relatedParty* - References to the provider that sold the product and the customer that booked it
- *status* - Current lifecycle status of the product (e.g. created, active, terminated, etc.)
- *startDate* / *terminationDate* - Time period of validity of the product

REST Endpoints

- POST /productInventory/v1/product

Post Conditions

The product is successfully created in the PSS datastore.

Applicable Requirements

- PSI-05-03-01-01
- PSI-05-03-01-02

eTOM Reference

The operation is based on the 1.2.11 process identifier from the eTOM.

5.5.3.2 TOD-05-03-02-Update_Product

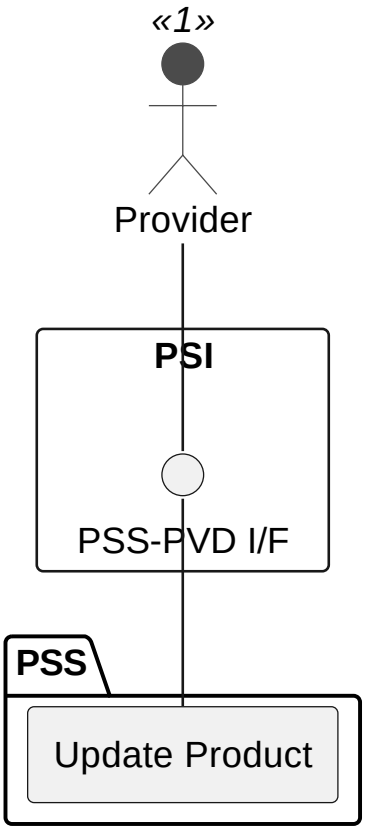

Figure 5.112: TOD-05-03-02: Update Product

Prerequisites

The product exists in the PSS datastore.

Main operation

Updates an existing product via a standard interface.

REST Endpoints

- PATCH /productInventory/v1/product/{id}

Post Conditions

The product is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-05-03-02-01

eTOM Reference

The operation is based on the 1.2.11 process identifier from the eTOM.

5.5.3.3 TOD-05-03-03-Remove_Product

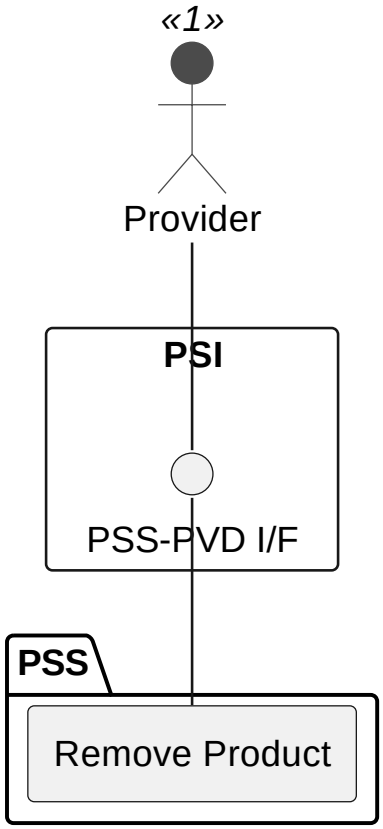

Figure 5.113: TOD-05-03-03: Remove Product

Prerequisites

The product exists in the PSS datastore.

Main operation

Removes a product either by deleting it or indicating it is no longer valid, via a standard interface.

REST Endpoints

- DELETE /productInventory/v1/product/{id}

Post Conditions

The product is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-05-03-03-01
- PSI-05-03-03-02

eTOM Reference

The operation is based on the 1.2.11 process identifier from the eTOM.

5.5.3.4 TOD-05-03-04-View_Product

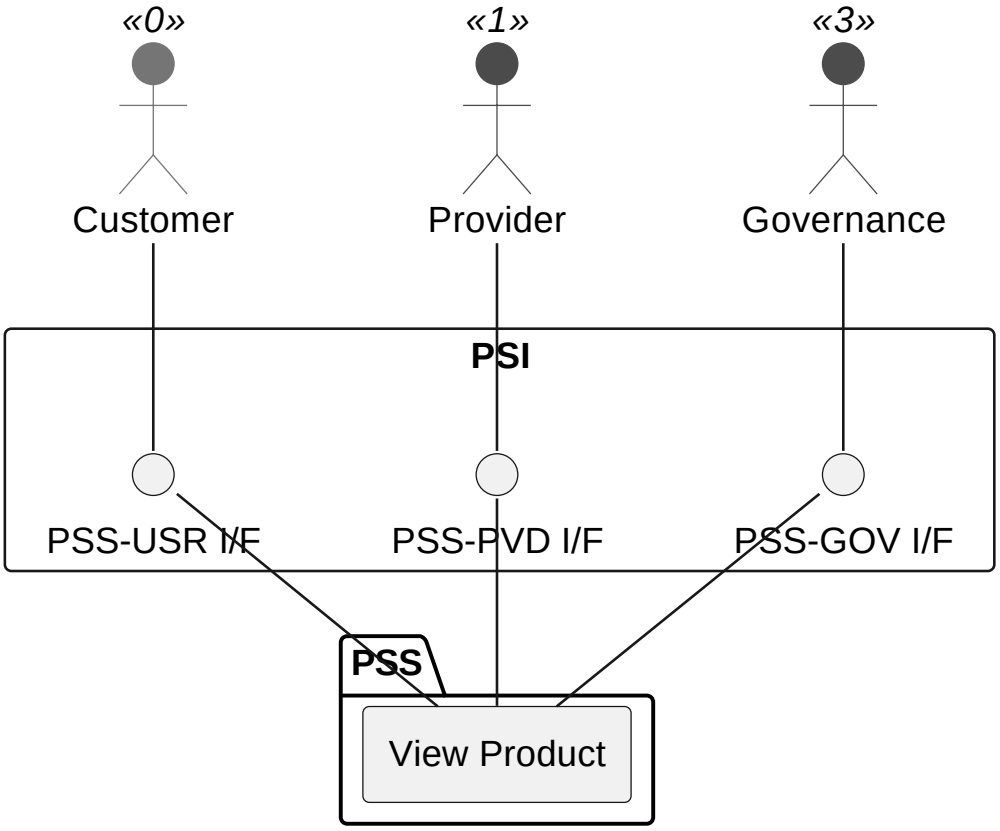

Figure 5.114: TOD-05-03-04: View Product

Prerequisites

The product exists in the PSS datastore.

Main operation

Gets a product with a specific identifier via a standard interface. Customers can only see the products they booked and providers can only see the products they have created.

REST Endpoints

- GET /productInventory/v1/product/{id}

Post Conditions

The product is successfully returned to be viewed.

Applicable Requirements

- PSI-05-03-04-01
- PSI-05-03-04-02

eTOM Reference

The operation is based on the 1.2.11 process identifier from the eTOM.

5.5.3.5 TOD-05-03-05-View_All_Products

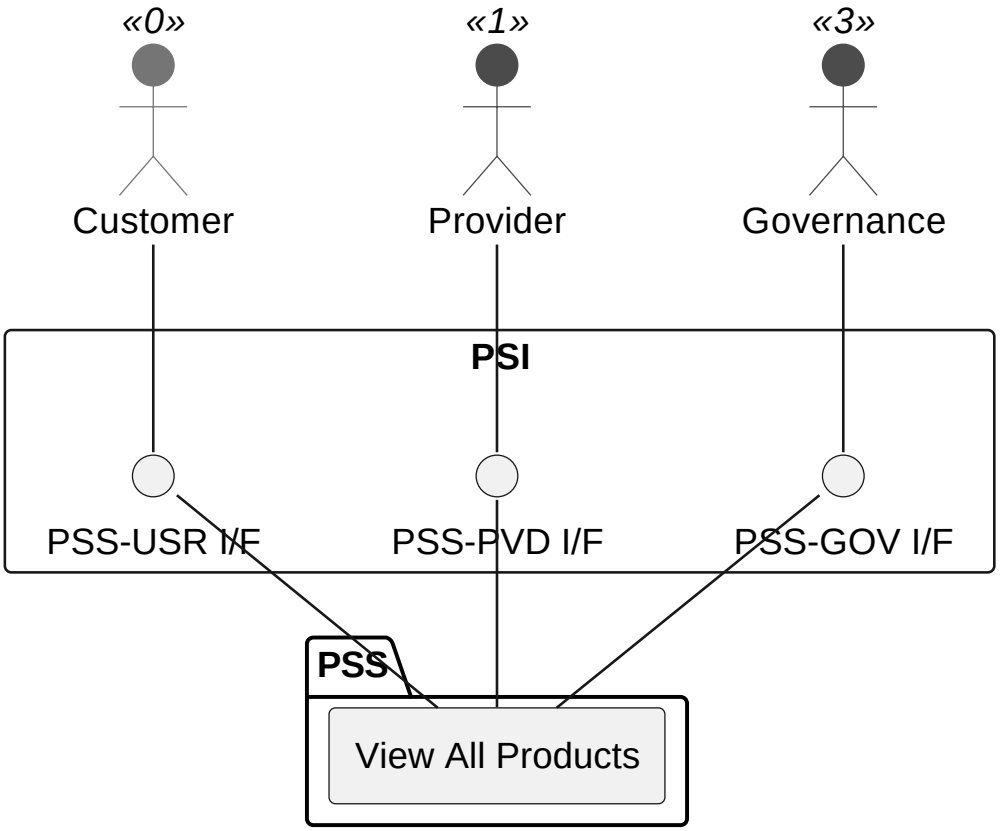

Figure 5.115: TOD-05-03-05: View All Products

Prerequisites

Products exist in the PSS datastore.

Main operation

Gets all products via a standard interface. Customers can only see the products they booked.

REST Endpoints

- GET /productInventory/v1/product

Post Conditions

All visible products are successfully returned to be viewed.

Applicable Requirements

- PSI-05-03-05-01
- PSI-05-03-05-02

eTOM Reference

The operation is based on the 1.2.11 process identifier from the eTOM.

5.5.4 TOD-05-04-Stock_Management

The Stock Management task wraps the inventories to allow a PSS (on behalf of a customer) to check the availability of a provider's product. There are more operations that are performed internally on the provider side, which are not covered by the PSID but may be implemented consistently with TM Forum.

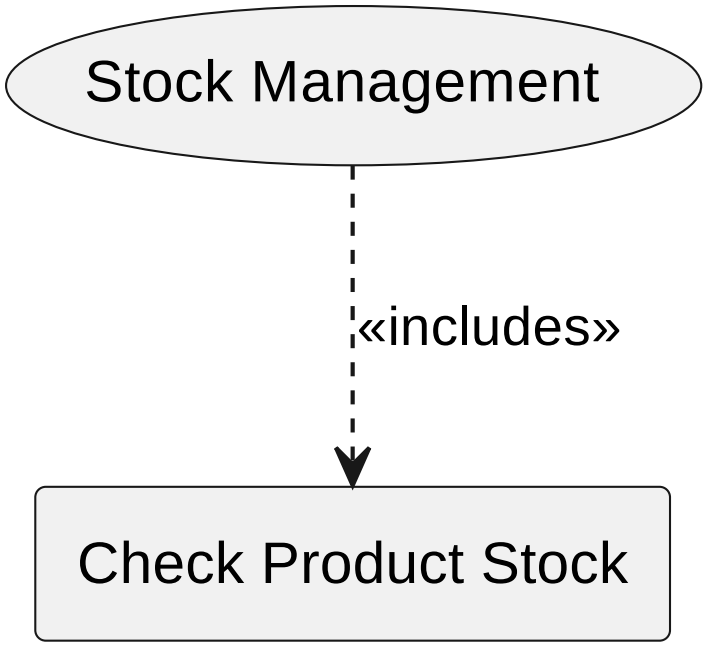

Figure 5.116: TOD-05-04: Stock Management

	Customer	Provider	Other PSS	Governance	PSS
Check Product Stock					✓

Table 5.19: Stock Management Matrix.

Please note that in this case the provider's system implements the endpoints and the PSS acts as the client.

eTOM Reference

The task is based on the 1.5.4.6 process identifier from the eTOM.

5.5.4.1 TOD-05-04-01-Check_Product_Stock

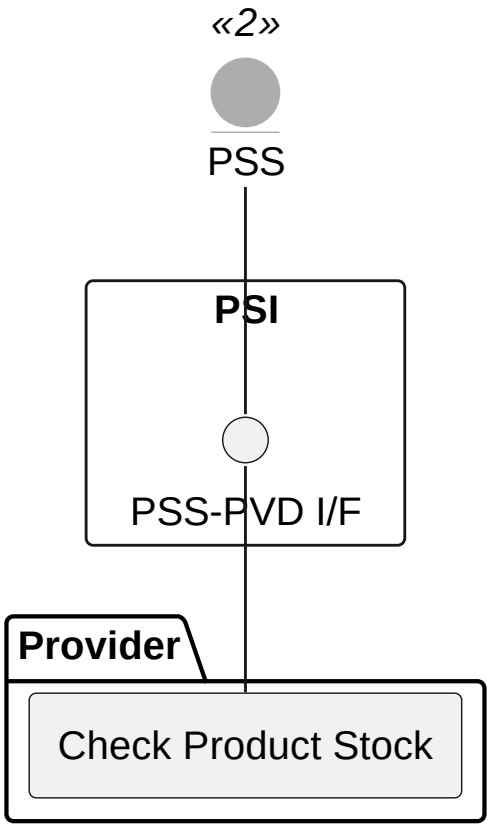

Figure 5.117: TOD-05-04-01: Check Product Stock

Prerequisites

At least one Product Specification must be known.

Main operation

The customer selects an on-demand product they found via matchmaking in the PSS. Since the availability is unknown to the system, it sends a request to the provider to check it. It contains the following information:

- *instantSyncCheck* - Whether the result should be returned as part of the response. If false, the action is created on the provider side and can be fetched later.
- *provideAlternative* - Whether the provider is allowed to return alternatives if the product is not available.
- *requestedAvailabilityStart* / *requestedAvailabilityEnd* - The time interval where the product is to be booked. Usually equal to the interval defined in the inquiry.
- *checkProductStockItem* - A list of product specifications for which the availability is requested. Each one also contains the characteristics the stocked product shall have. If a characteristic is not listed, it is irrelevant for the check.

The action is created via the `POST` endpoint and will get a unique ID by the server. The result is written to the `availabilityResult` field of each requested item, indicating whether it is available, unavailable or alternatives are provided in the `alternate` field. Though the `instantSyncCheck` signals the expectation of the client, the

server can **always** return two status codes which have to be handled appropriately (e.g. by visualising it to the customer):

- Code `200` means that the result is available immediately in the first response.
- Code `201` means that the result can be fetched later via the `GET` endpoint using the ID.

REST Endpoints

- POST /stock/v1/checkProductStock
- GET /stock/v1/checkProductStock/{id}

Post Conditions

The action is created on the provider system. Depending on the return code, the result is available immediately or after a delay.

Applicable Requirements

- PSI-05-04-01-01
- PSI-05-04-01-02
- PSI-05-04-01-03

eTOM Reference

The operation is based on the 1.5.4.6 and 1.6.8.2 processes identifier from the eTOM.

5.6 TOD-06-Quality_Management

The category consists of tasks and operations related to managing the service level objectives (SLO) and service level specifications (SLS), which in turn are used to define service level agreements (SLAs) and declare monitoring of services and resources on provider side. It also contains tasks and operations related to performance monitoring jobs and performance monitoring reports which allow for assessing the performance characteristics of a service.

5.6.1 TOD-06-01-Service_Level_Objective_Management

The Service Level Objective Management task takes care of the maintenance of service level objectives (SLOs) in the PSS. These SLOs direct and control the performance of services to meet the expectations of the customers. They are combined in SLSs (see TOD-06-02), which are part of service specifications and product offerings.

Expected performance of the service is defined by thresholds attached to key indicators. The service provider creates an SLO based on the key indicators (KPIs/KQIs) which are pre-defined by the governance (see TOD-06-03).

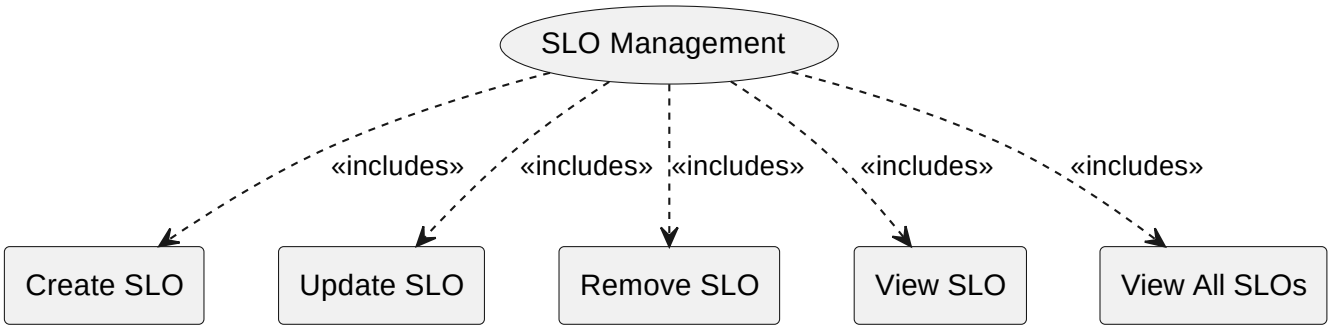

Figure 5.118: TOD-06-01: Service Level Objective (SLO) Management

	Customer	Provider	Other PSS	Governance
Create SLO		✓		
Update SLO		✓		
Remove SLO		✓		
View SLO	✓	✓		✓
View All SLOs	✓	✓		✓

Table 5.20: Service Level Objective Management Matrix.

eTOM Reference

The task is based on the 1.4.7 process identifier from the eTOM.

5.6.1.1 TOD-06-01-01-Create_Service_Level_Objective

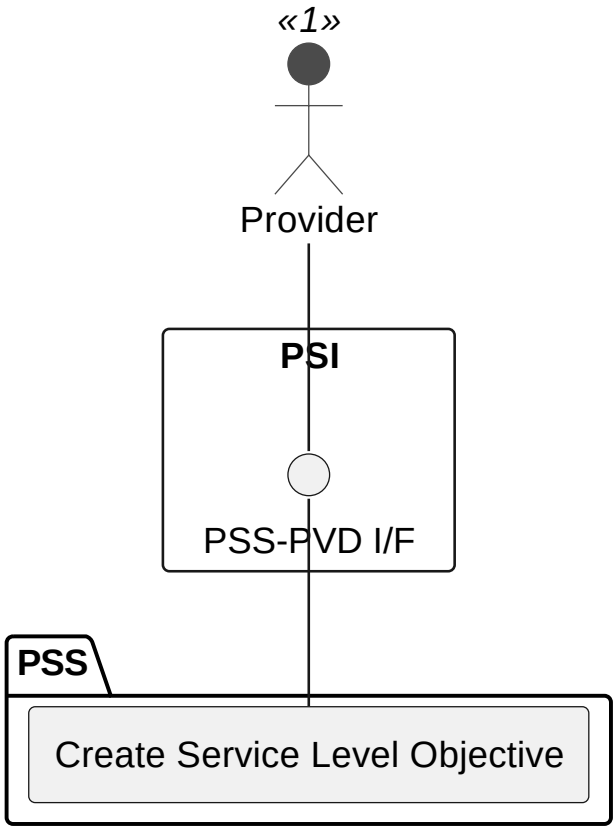

Figure 5.119: TOD-06-01-01: Create Service Level Objective

Prerequisites

The service level objective does not exist in the PSS datastore.

Main operation

Creates a new service level objective instance via a standard interface.

Some properties of a service level objective are:

- *name* - Short name of the service level objective
- *validFor* - The validity of the service level objective
- *keyIndicator* - The service level indicator (SLI) used for this SLO
- *threshold* - A list of thresholds that applies to this objective
- *toleranceTarget* - Indicating the allowable variation of a compliance goal within the tolerance period
- *tolerancePeriod* - Defines the period for the applicability of the tolerance target
- *applicability* - The applicability of the SLO in relation to the schedule

REST Endpoints

- POST /serviceQuality/v1/serviceLevelObjective

Post Conditions

The service level objective is successfully created in the PSS datastore.

Applicable Requirements

- PSI-06-01-01-01

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.1.2 TOD-06-01-02-Update_Service_Level_Objective

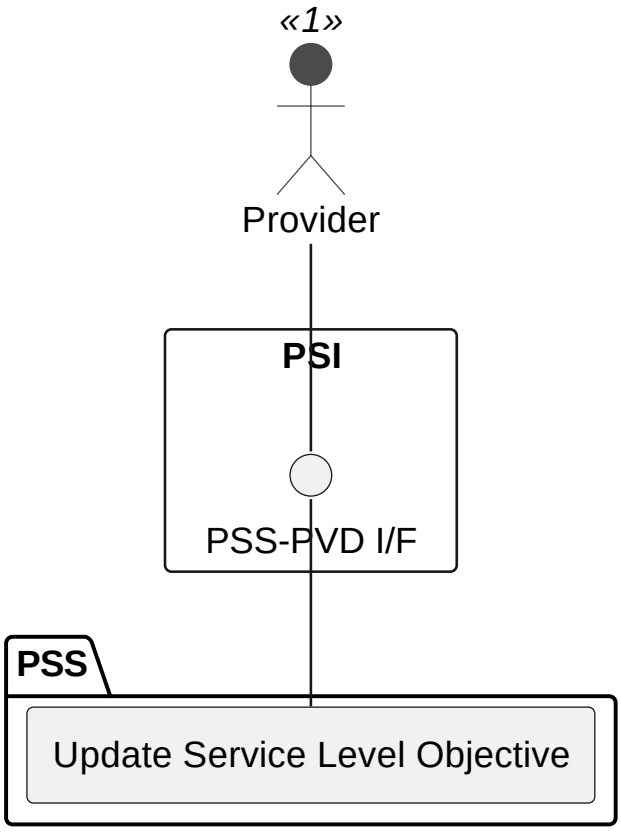

Figure 5.120: TOD-06-01-02: Update Service Level Objective

Prerequisites

The service level objective exists in the PSS datastore.

Main operation

Updates an existing service level objective instance via a standard interface.

REST Endpoints

- PATCH /serviceQuality/v1/serviceLevelObjective/{id}

Post Conditions

The service level objective is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-06-01-02-01

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.1.3 TOD-06-01-03-Remove_Service_Level_Objective

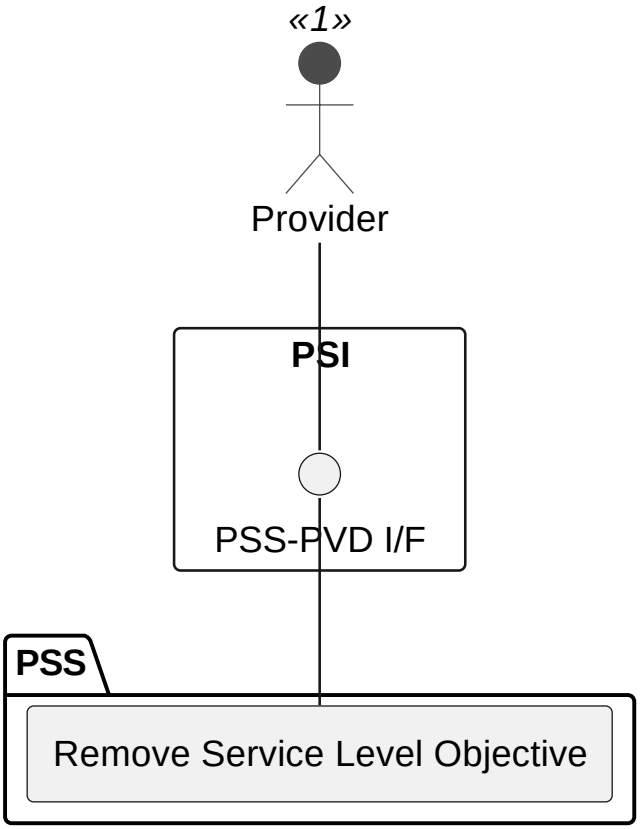

Figure 5.121: TOD-06-01-03: Remove Service Level Objective

Prerequisites

The service level objective exists in the PSS datastore.

Main operation

Removes a service level objective instance either by deleting it or indicating it is no longer valid, via a standard interface.

REST Endpoints

- DELETE /serviceQuality/v1/serviceLevelObjective/{id}

Post Conditions

The service level objective is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-06-01-03-01
- PSI-06-01-03-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.1.4 TOD-06-01-04-View_Service_Level_Objective

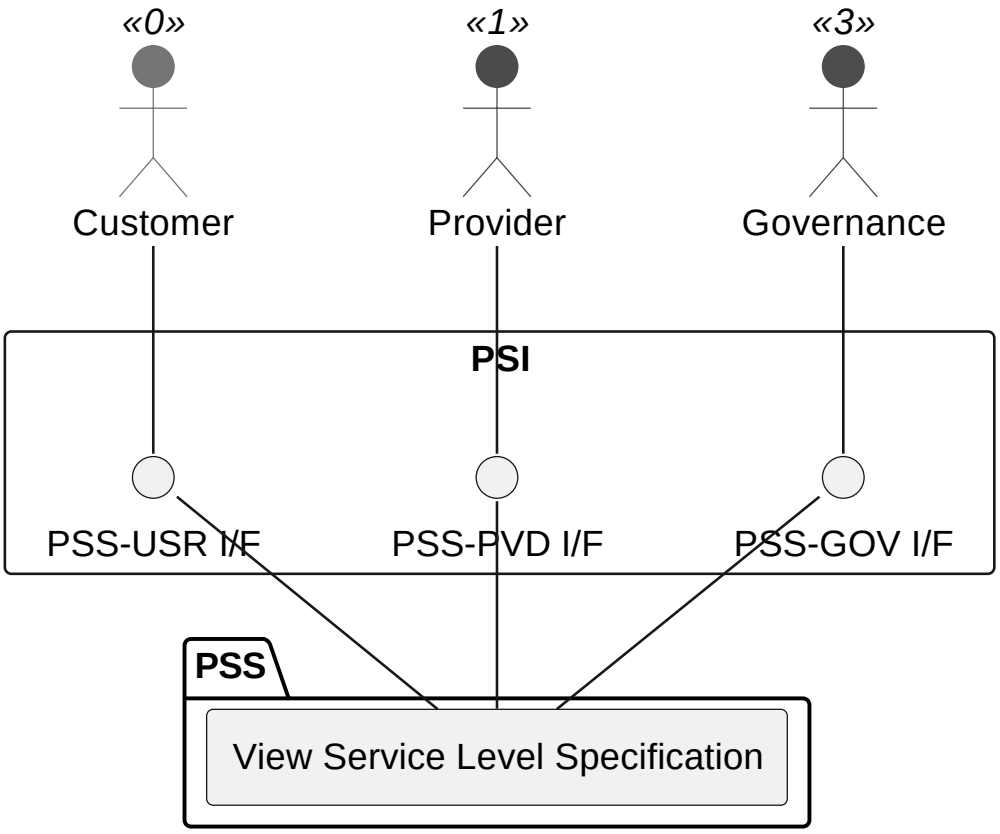

Figure 5.122: TOD-06-01-04: View Service Level Objective

Prerequisites

The service level objective exists in the PSS datastore.

Main operation

Gets a service level objective instance via a standard interface. Customers can only view the service level objectives granted by providers.

REST Endpoints

- GET /serviceQuality/v1/serviceLevelObjective/{id}

Post Conditions

The service level objective is successfully returned for viewing.

Applicable Requirements

- PSI-06-01-04-01
- PSI-06-01-04-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.1.5 TOD-06-01-05-View_All_Service_Level_Objective

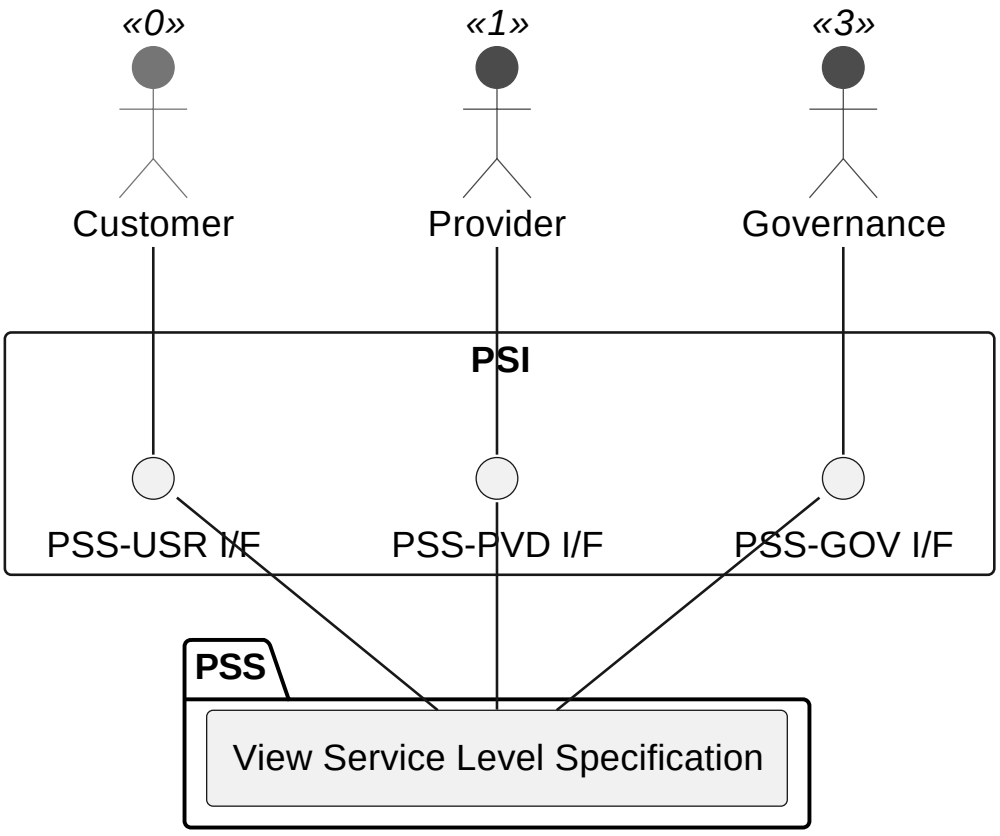

Figure 5.123: TOD-06-01-05: View All Service Level Objectives

Prerequisites

Service level objectives exist in the PSS datastore.

Main operation

Gets all service level objective instances via a standard interface. Customers can only view the service level objectives granted by providers.

REST Endpoints

- GET /serviceQuality/v1/serviceLevelObjective

Post Conditions

All visible service level objectives are successfully returned for viewing.

Applicable Requirements

- PSI-06-01-05-01
- PSI-06-01-05-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.2 TOD-06-02-Service_Level_Specification

The Service Level Specification Management task takes care of the maintenance of service level specifications (SLSs) in the PSS. An SLS represents a predefined or negotiated set of service level objectives (SLO - see TOD-06-01). The requirements for service instances that must be met can then be represented as SLS, which are linked from the corresponding product offerings and/or service specifications.

When an order is issued, a legally binding SLA between the parties might result from an SLS. As per its nature, it is often documented in PDF or other text formats. Those documents can be stored in the document management (see TOD-01-03).

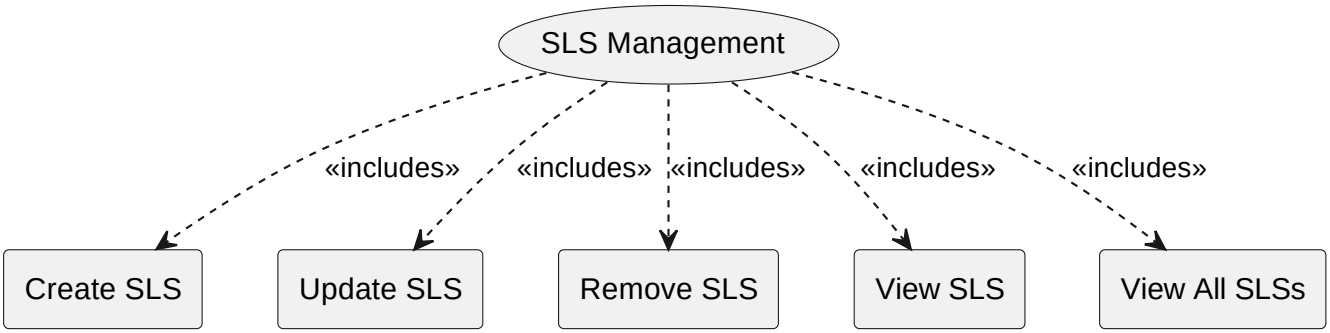

Figure 5.124: TOD-06-02: Service Level Specification (SLS) Management

	Customer	Provider	Other PSS	Governance
Create SLS		✓		
Update SLS		✓		
Remove SLS		✓		
View SLS	✓	✓		✓
View All SLSs	✓	✓		✓

Table 5.21: Service Level Specification Management Matrix.

eTOM Reference

The task is based on the 1.4.7 process identifier from the eTOM.

5.6.2.1 TOD-06-02-01-Create_Service_Level_Specification

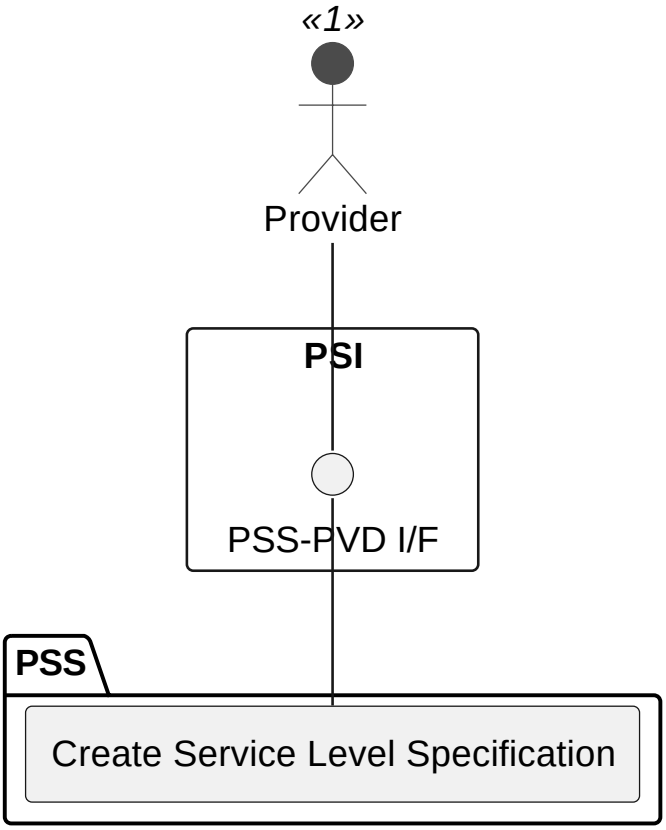

Figure 5.125: TOD-06-02-01: Create Service Level Specification

Prerequisites

The service level specification does not exist in the PSS datastore and referenced service level objectives are available.

Main operation

Creates a new service level specification instance via a standard interface.

Some properties of a service level specification are:

- *name* - Short name of the service level specification
- *description* - A brief introduction of the service level specification
- *relatedServiceLevelObjective* - A list of objectives belonging to this service level specification
- *validFor* - The validity of the service level specification

REST Endpoints

- POST /serviceQuality/v1/serviceLevelSpecification

Post Conditions

The service level specification is successfully created in the PSS datastore.

Applicable Requirements

- PSI-06-02-01-01

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.2.2 TOD-06-02-02-Update_Service_Level_Specification

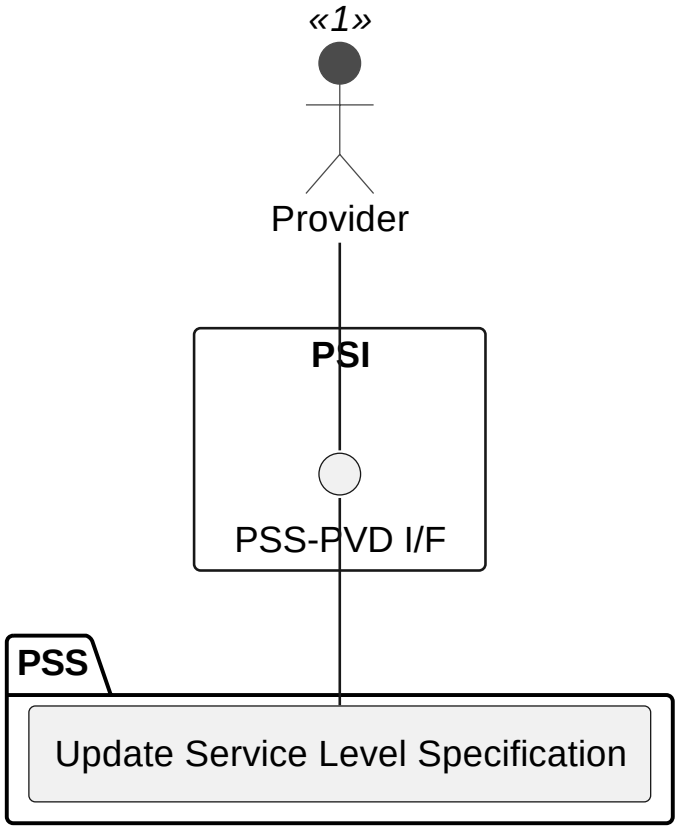

Figure 5.126: TOD-06-02-02: Update Service Level Specification

Prerequisites

The service level specification exists in the PSS datastore and referenced service level objectives are available.

Main operation

Updates a service level specification instance via a standard interface.

REST Endpoints

- PATCH /serviceQuality/v1/serviceLevelSpecification/{id}

Post Conditions

The service level specification is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-06-02-02-01

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.2.3 TOD-06-02-03-Remove_Service_Level_Specification

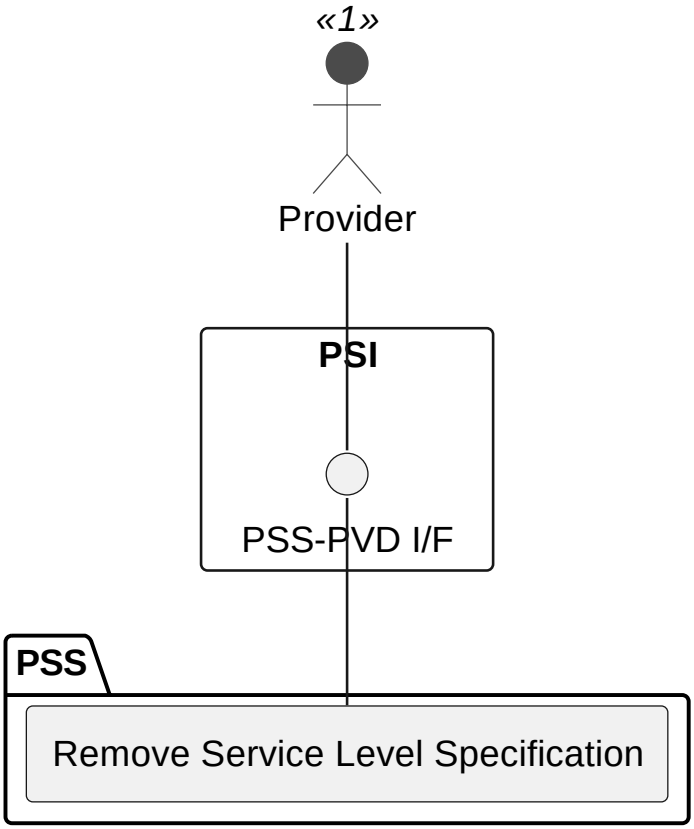

Figure 5.127: TOD-06-02-03: Remove Service Level Specification

Prerequisites

The service level specification exists in the PSS datastore.

Main operation

Removes a service level specification instance via a standard interface.

REST Endpoints

- DELETE /serviceQuality/v1/serviceLevelSpecification/{id}

Post Conditions

The service level specification is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-06-02-03-01
- PSI-06-02-03-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.2.4 TOD-06-02-04-View_Service_Level_Specification

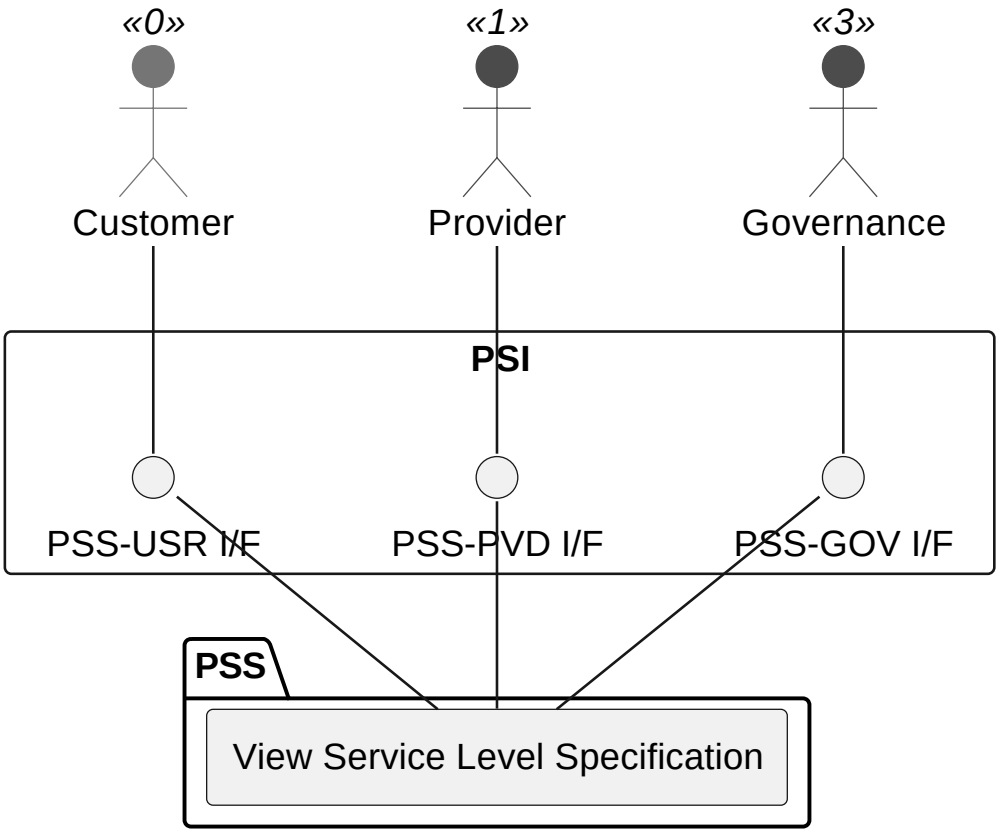

Figure 5.128: TOD-06-02-04: View Service Level Specification

Prerequisites

The service level specification exists in the PSS datastore.

Main operation

Gets a service level specification instance via a standard interface. Customers can only view the service level specifications granted by providers.

REST Endpoints

- GET /serviceQuality/v1/serviceLevelSpecification/{id}

Post Conditions

The service level specification is successfully returned for viewing.

Applicable Requirements

- PSI-06-02-04-01
- PSI-06-02-04-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.2.5 TOD-06-02-05-View_All_Service_Level_Specification

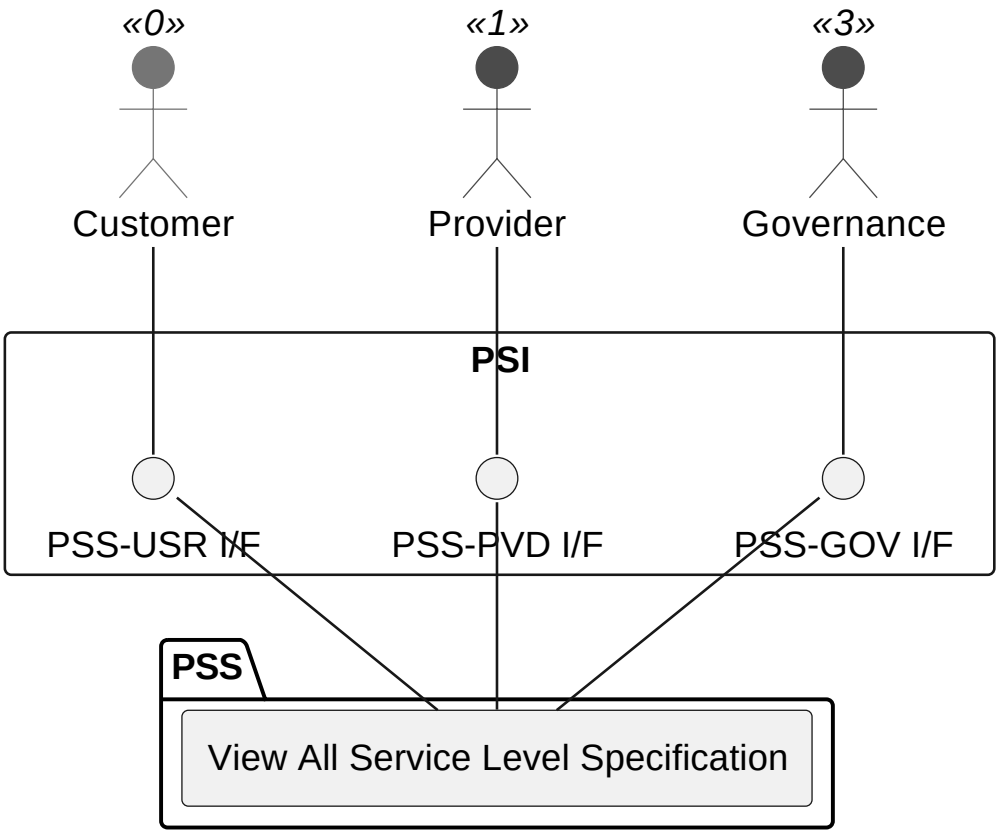

Figure 5.129: TOD-06-02-05: View All Service Level Specification

Prerequisites

The service level specification exists in the PSS datastore.

Main operation

Gets all service level specification instance via a standard interface. Customers can only view the service level specifications granted by providers.

REST Endpoints

- GET /serviceQuality/v1/serviceLevelSpecification

Post Conditions

All visible service level specifications are successfully returned for viewing.

Applicable Requirements

- PSI-06-02-05-01

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.3 TOD-06-03-Key_Indicator_Management

The Key Indicator Management API task takes care of the maintenance of key indicators in the PSS. Key Indicators (KPI/KQI) are technical performance or quality parameters of the service. They are used by the provider to create Service Level Objectives which define threshold values per key indicator (see TOD-06-01). These thresholds determine the service quality commitment of the service provider.

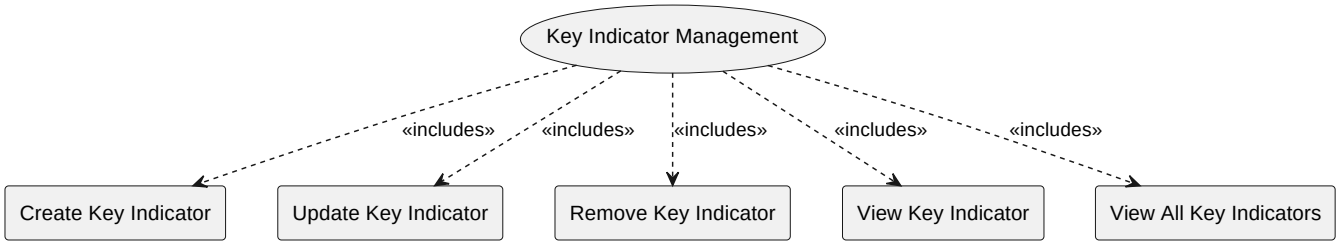

Figure 5.130: **TOD-06-03**: Key Indicator Management

	Customer	Provider	Other PSS	Governance
Create Key Indicator				✓
Update Key Indicator				✓
Remove Key Indicator				✓
View Key Indicator		✓	✓	✓
View Key Indicators		✓	✓	✓

Table 5.22: Key Indicator Management Matrix.

eTOM Reference

The task is based on the 1.4.7 process identifier from the eTOM.

5.6.3.1 TOD-06-03-01-Create_Key_Indicator

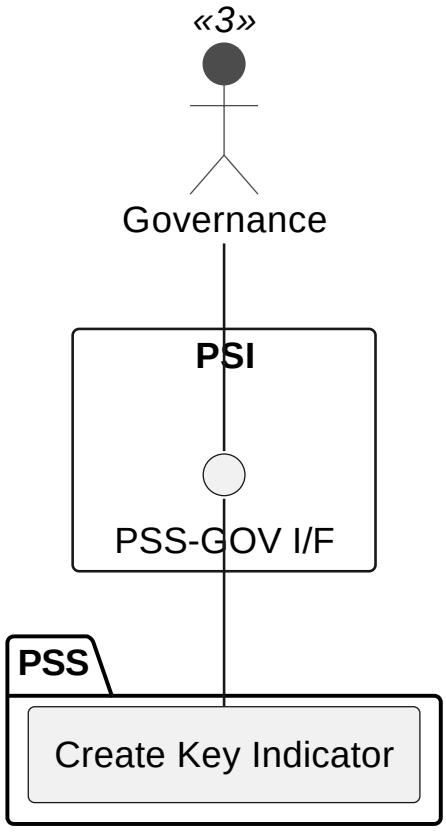

Figure 5.131: **TOD-06-03-01**: Create Key Indicator

Prerequisites

The key indicator does not exist in the PSS datastore.

Main operation

Creates a new key indicator instance via a standard interface.

Some properties of a key indicator are:

- *name* - Short name of the key indicator
- *indicatorType* - KPI or KQI
- *isBundled* - A boolean that specifies whether the key indicator represents a single key indicator (false) or a key indicator that represents an aggregation (true)
- *validFor* - The validity of the key indicator
- *keyIndicatorRelationship* - A list of key indicator relationships related to this object

REST Endpoints

- POST /serviceQuality/v1/keyIndicator

Post Conditions

The key indicator is successfully created in the PSS datastore.

Applicable Requirements

- PSI-06-03-01-01
- PSI-06-03-01-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.3.2 TOD-06-03-02-Update_Key_Indicator

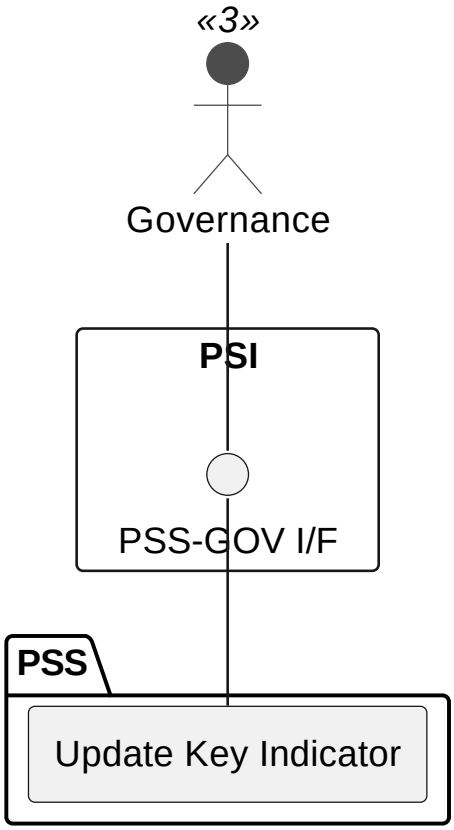

Figure 5.132: **TOD-06-03-02**: Update Key Indicator

Prerequisites

The key indicator exists in the PSS datastore.

Main operation

Updates a key indicator instance via a standard interface.

REST Endpoints

- PATCH /serviceQuality/v1/keyIndicator/{id}

Post Conditions

The key indicator is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-06-03-02-01
- PSI-06-03-02-02
- PSI-06-03-02-03

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.3.3 TOD-06-03-03-Remove_Key_Indicator

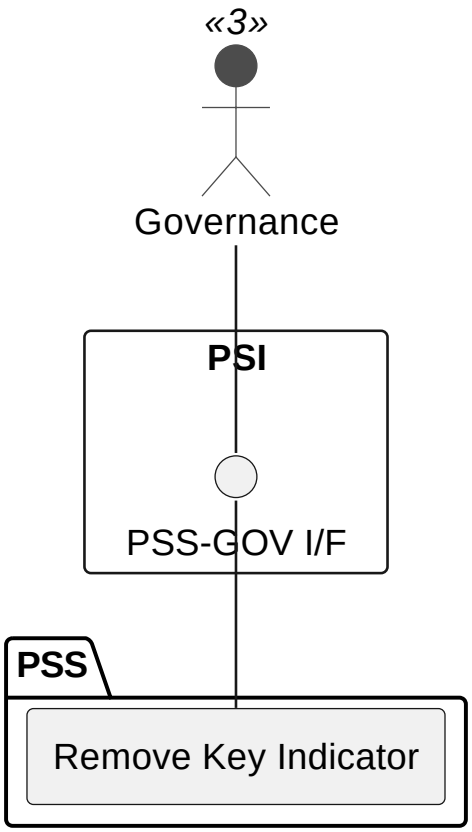

Figure 5.133: **TOD-06-03-03**: Remove Key Indicator

Prerequisites

The Key Indicator exists in the PSS datastore and is not used by available Service Level Objective.

Main operation

Removes a Key Indicator instance via a standard interface.

REST Endpoints

- DELETE /serviceQuality/v1/keyIndicator/{id}

Post Conditions

The Key Indicator is successfully deleted or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-06-03-03-01
- PSI-06-03-03-02
- PSI-06-03-03-03
- PSI-06-03-03-04

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.3.4 TOD-06-03-04-View_Key_Indicator

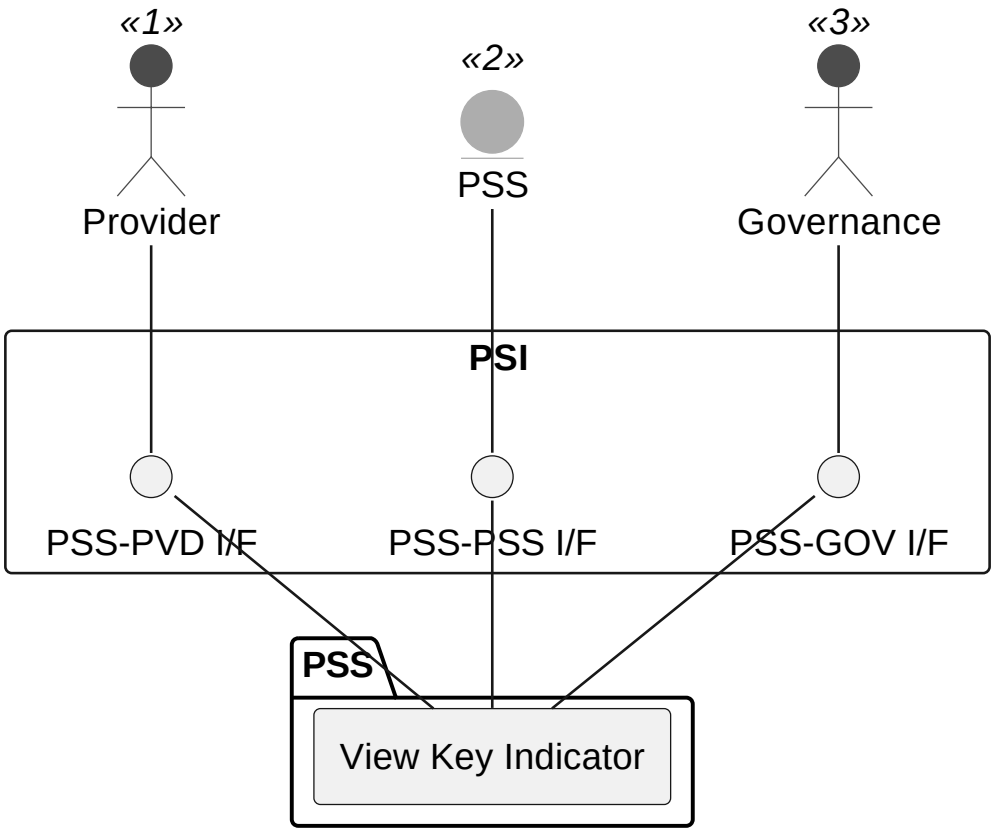

Figure 5.134: **TOD-06-03-04**: View Key Indicator

Prerequisites

The Key Indicator exists in the PSS datastore.

Main operation

Gets a Key Indicator instance via a standard interface.

REST Endpoints

- GET /serviceQuality/v1/keyIndicator/{id}

Post Conditions

The Key Indicator is successfully returned for viewing.

Applicable Requirements

- PSI-06-03-04-01
- PSI-06-03-04-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.3.5 TOD-06-03-05-View_All_Key_Indicator

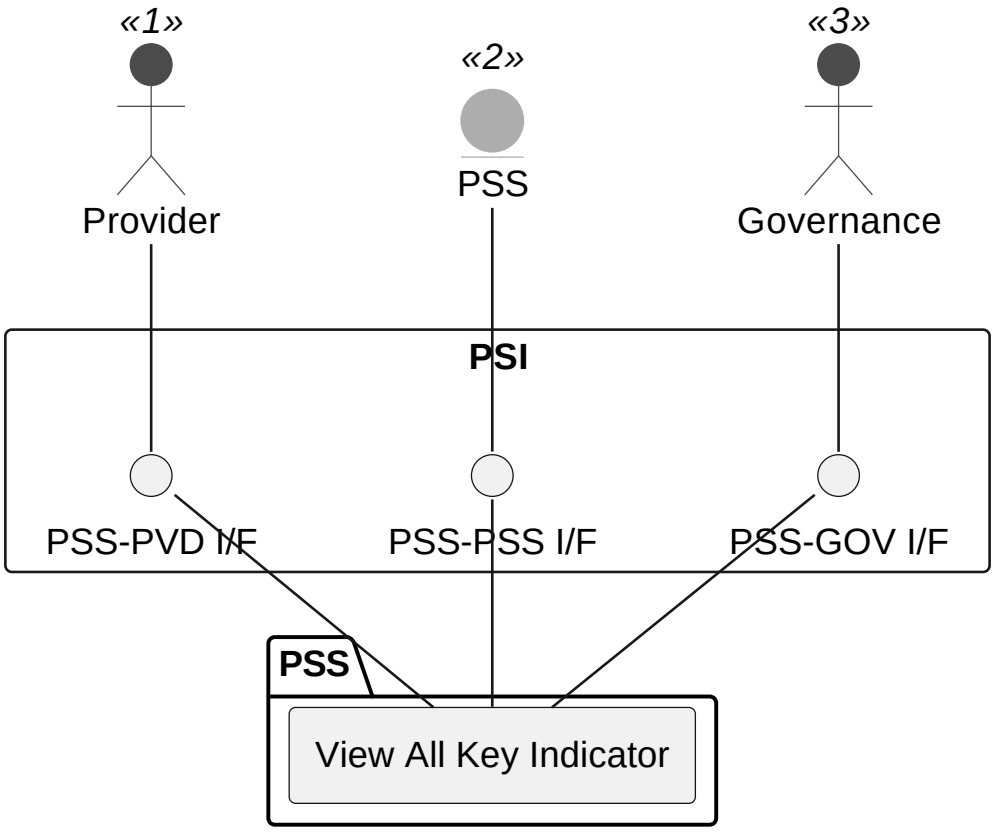

Figure 5.135: **TOD-06-03-05**: View All Key Indicator

Prerequisites

The Key Indicator exists in the PSS datastore.

Main operation

Gets all Key Indicator instance via a standard interface.

REST Endpoints

- GET /serviceQuality/v1/keyIndicator

Post Conditions

All visible Key Indicators are successfully returned for viewing.

Applicable Requirements

- PSI-06-03-05-01
- PSI-06-03-05-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.4 TOD-06-04-Performance_Monitoring_Job_Management

The Performance Monitoring Job Management task takes care of the maintenance of PM jobs in the PSS.

The performance monitoring job is responsible for provisioning appropriate measurement points, and performance objectives, together with measurement intervals and schedules, to measure applicable KPIs/KQIs specific to the service and defined by service level objectives.

Performance monitoring jobs are typically associated with an SLS but can be used for an on-demand performance measurement that is initiated for a limited time, typically a single run or non-continuous run, to carry out the performance measurement tests and support troubleshooting during service assurance. Management of a performance monitoring job is depicted in the following diagram.

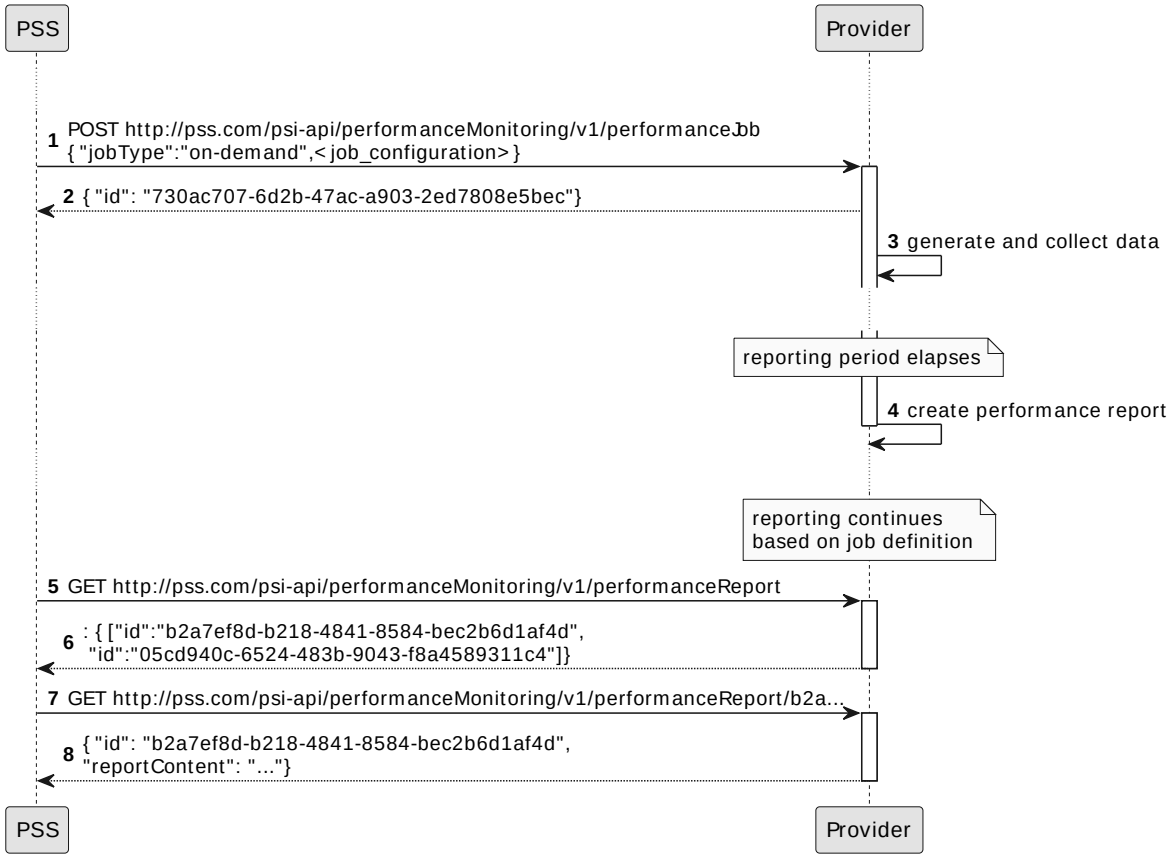

Figure 5.136: **TOD-06-04**: Performance Monitoring Job Sequence

Operations applicable to the performance monitoring job are listed below.

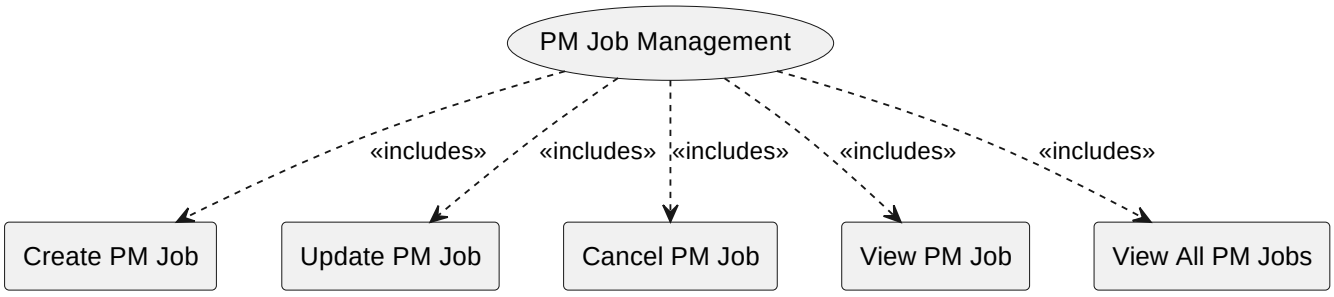

Figure 5.137: **TOD-06-04**: Performance Monitoring Job Management

	Customer	Provider	Other PSS	Governance
Create PM Job	✓		✓	
Update PM Job	✓		✓	
Cancel PM Job	✓		✓	
View PM Job	✓		✓	✓
View All PM Jobs	✓		✓	✓

Table 5.23: Performance Monitoring Job Management Matrix.

eTOM Reference

The task is based on the 1.4.7 process identifier from the eTOM.

5.6.4.1 TOD-06-04-01-Create_Performance_Monitoring_Job

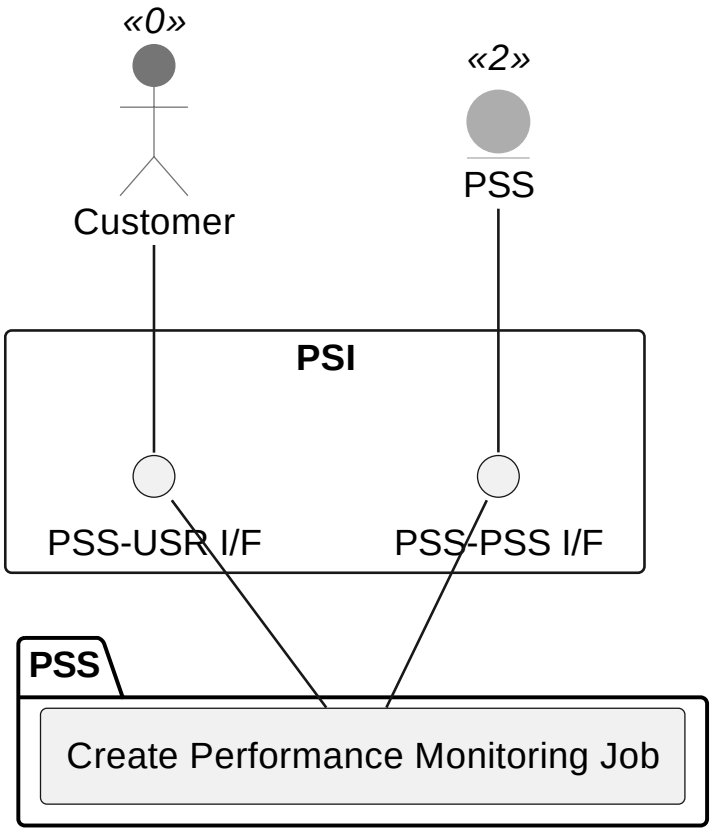

Figure 5.138: **TOD-06-04-01**: Create Performance Monitoring Job

Prerequisites

Service for which PM job needs to be executed exists in the PSS datastore. Applicable key indicators are defined in the PSS datastore.

Main operation

Creates a new Performance Monitoring Job instance via a standard interface.

Some properties of a Performance Monitoring Job are:

- *granularity* - Sampling rate of the collection or production of performance indicators, e.g. 10 milliseconds, 1 second, 5 minutes, 24 hours
- *jobType* - Always on-demand
- *outputFormat* - Output formats for the Performance Report: json, xml, avro, csv
- *reportingPeriod* - Defines the interval for the report generation
- *resultFormat* - Defines how the provider will deliver the Performance Report generated by the job to the PSS:

- payload data will be delivered directly in the “Get Report” API response
- attachment data will be stored in the file and the API response will contain a URL to the file
- *scheduleDefinition* - The schedule definition for running jobs
- *servicePayloadSpecificAttributes* - an extension point for service-specific performance monitoring configuration. It includes a definition of observed service and applicable performance monitoring objectives.

REST Endpoints

- POST /performanceMonitoring/v1/performanceJob

Post Conditions

The PM job is successfully created in the PSS datastore.

Applicable Requirements

- PSI-06-04-01-01
- PSI-06-04-01-02
- PSI-06-04-01-03
- PSI-06-04-01-04
- PSI-06-04-01-05

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.4.2 TOD-06-04-02-Update_Performance_Monitoring_Job

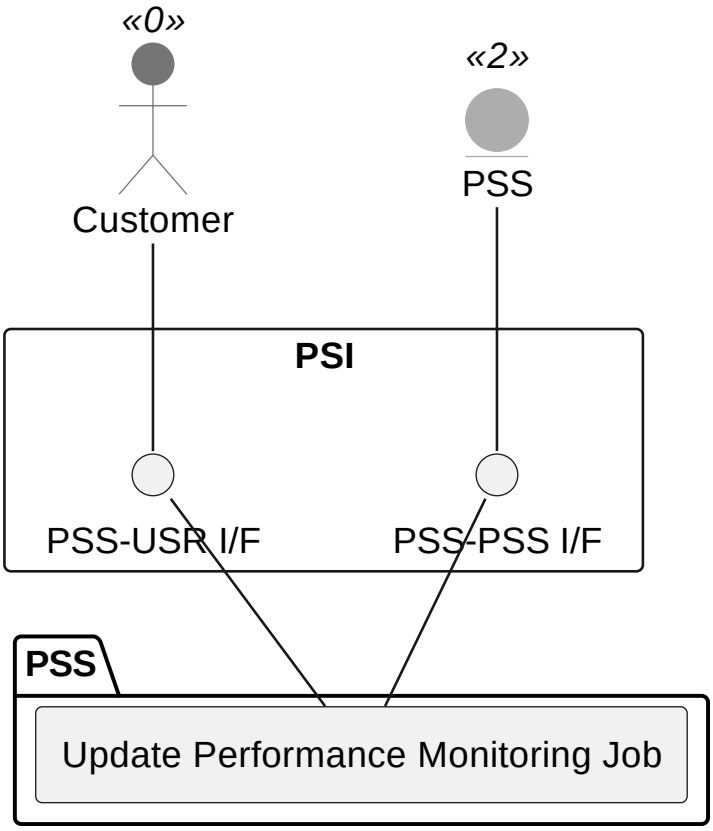

Figure 5.139: **TOD-06-04-02**: Update Performance Monitoring Job

Prerequisites

The Performance Monitoring Job exists in the PSS datastore.

Main operation

Updates a Performance Monitoring Job instance via a standard interface. Suspends/Resumes a Performance Monitoring Job instance via a standard interface.

REST Endpoints

- POST /performanceMonitoring/v1/modifyPerformanceJob

Post Conditions

The Performance Monitoring Job is successfully updated in the PSS datastore.

Applicable Requirements

- PSI-06-04-02-01
- PSI-06-04-02-02
- PSI-06-04-02-03

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.4.3 TOD-06-04-03-Cancel_Performance_Monitoring_Job

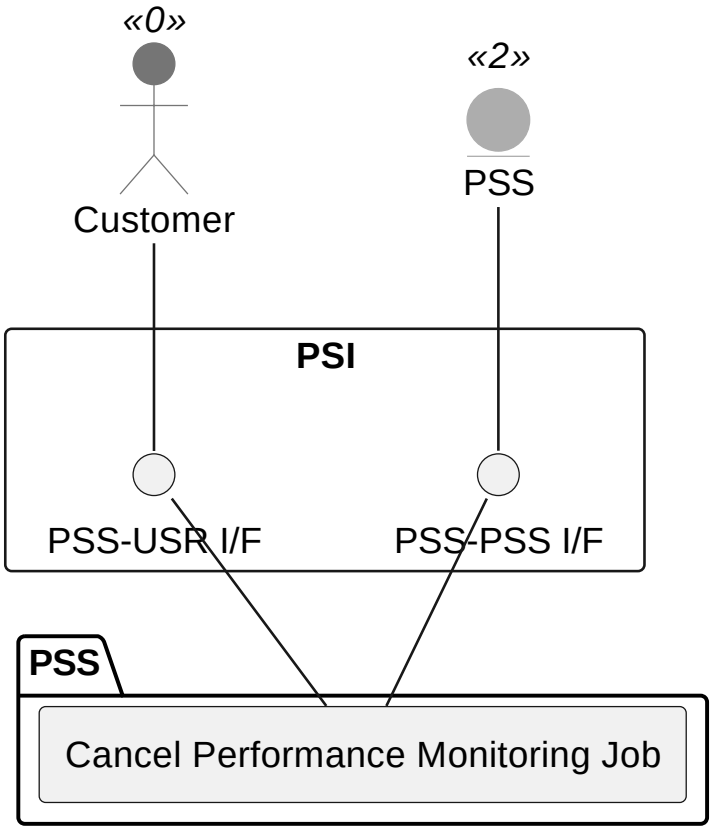

Figure 5.140: **TOD-06-04-03**: Cancel Performance Monitoring Job

Prerequisites

The Performance Monitoring Job exists in the PSS datastore.

Main operation

Cancels a Performance Monitoring Job instance via a standard interface.

REST Endpoints

- POST /performanceMonitoring/v1/cancelPerformanceJob

Post Conditions

The Performance Monitoring Job is successfully canceled or indicated it is no longer valid in the PSS datastore.

Applicable Requirements

- PSI-06-04-03-01
- PSI-06-04-03-02
- PSI-06-04-03-03

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.4.4 TOD-06-04-04-View_Performance_Monitoring_Job

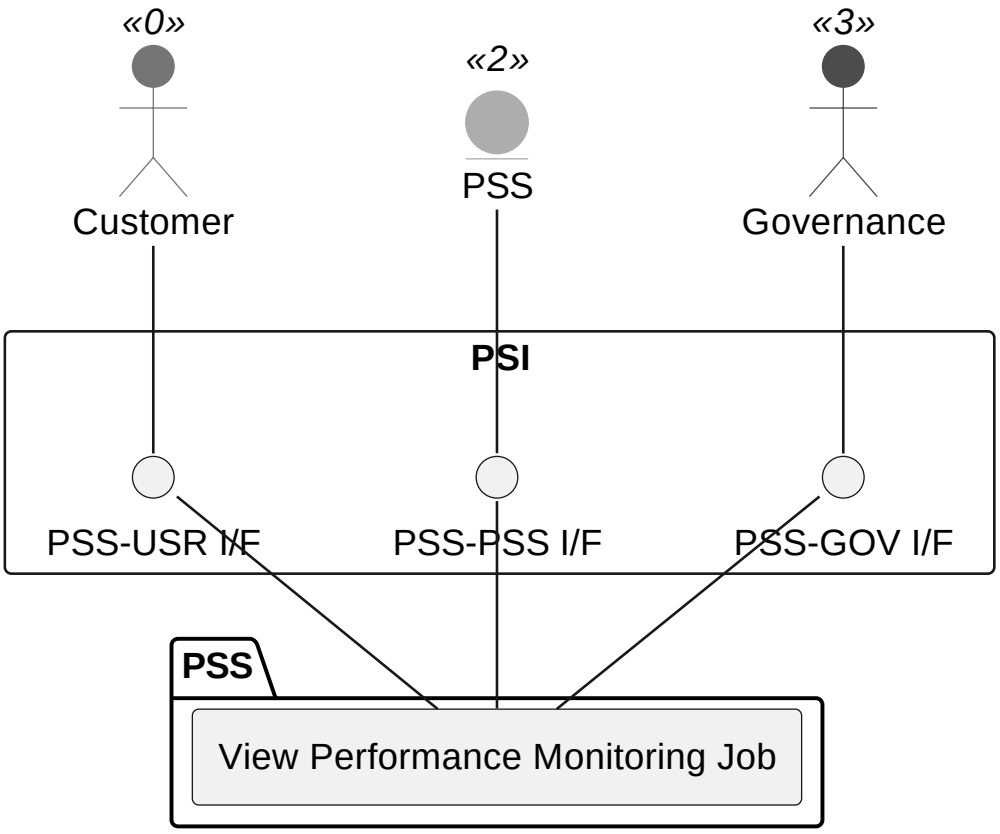

Figure 5.141: **TOD-06-04-04**: View Performance Monitoring Job

Prerequisites

The Performance Monitoring Job exists in the PSS datastore.

Main operation

Gets a Performance Monitoring Job instance via a standard interface.

REST Endpoints

- GET /performanceMonitoring/v1/performanceJob/{id}

Post Conditions

The Performance Monitoring Job is successfully returned for viewing.

Applicable Requirements

- PSI-06-04-04-01
- PSI-06-04-04-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.4.5 TOD-06-04-05-View_All_Performance_Monitoring_Job

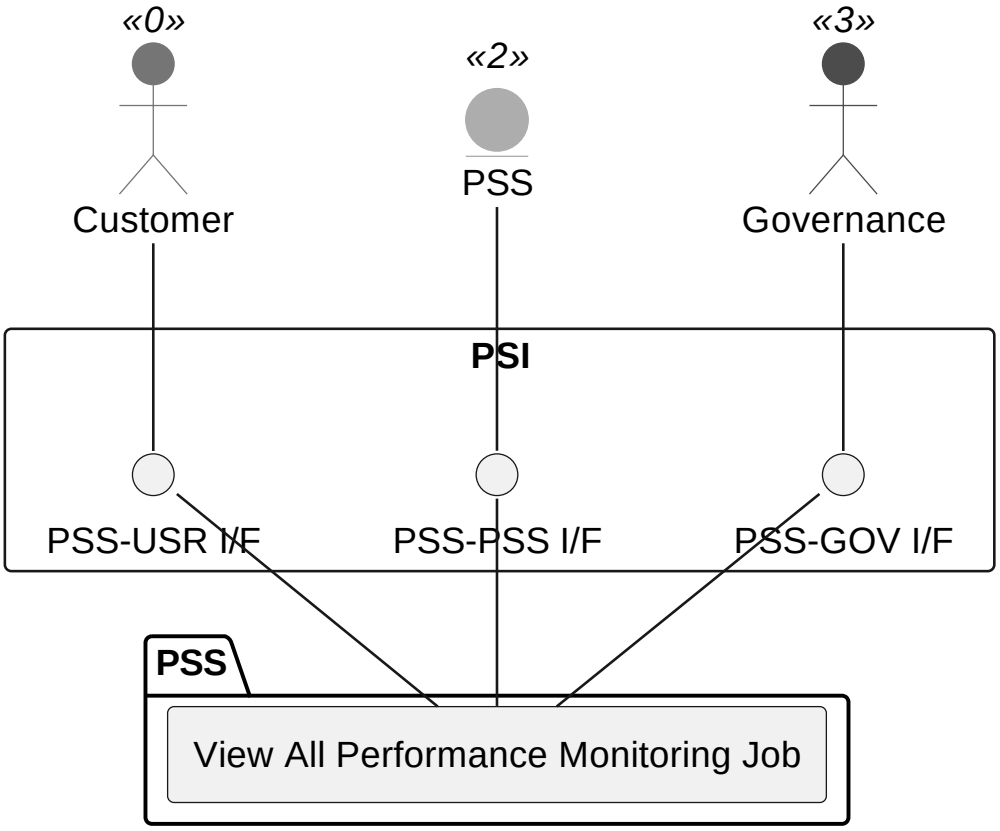

Figure 5.142: **TOD-06-04-05**: View All Performance Monitoring Job

Prerequisites

The Performance Monitoring Job exists in the PSS datastore.

Main operation

Gets all Performance Monitoring Job instances via a standard interface.

REST Endpoints

- GET /performanceMonitoring/v1/performanceJob
- POST /performanceMonitoring/v1/performanceJobComplexQuery

Post Conditions

All visible Performance Monitoring Jobs are successfully returned for viewing.

Applicable Requirements

- PSI-06-04-05-01
- PSI-06-04-05-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.5 TOD-06-05-Performance_Monitoring_Report_Management

The Performance Monitoring Report Management task takes care of the maintenance of PM reports in the PSS. Performance Monitoring Reports show data collected by the service provider. This data can be generated in two ways. First, when a service with an attached SLS is provisioned, the provider starts collecting performance measurements related to the SLS. In order to get the performance data, the PSS can request an ad-hoc report creation which is depicted in the diagram below.

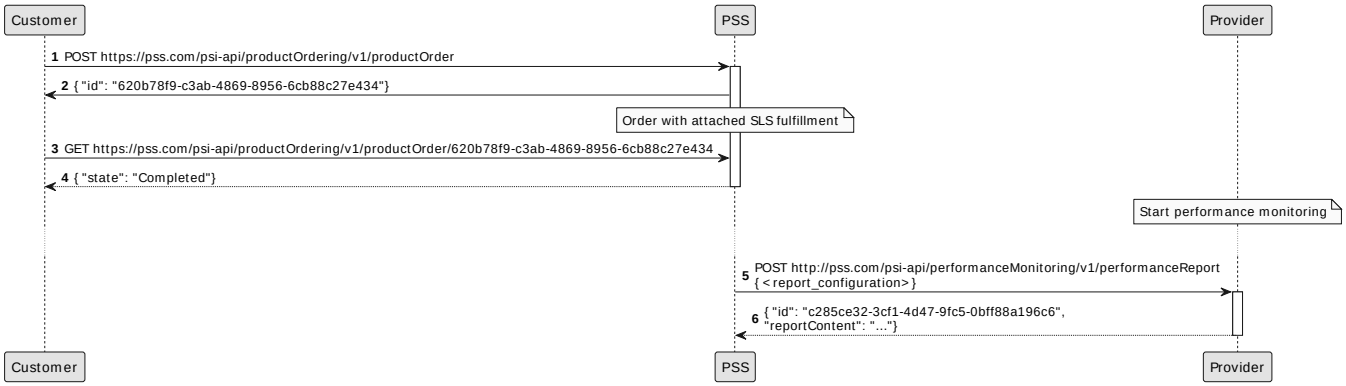

Figure 5.143: **TOD-06-05**: Performance Monitoring Report Sequence

A second option is to create an on-demand Performance Monitoring Job which will collect data and put it in the report (see TOD-06-04).

Operations available for managing a Performance Monitoring Report are listed in the following diagram.

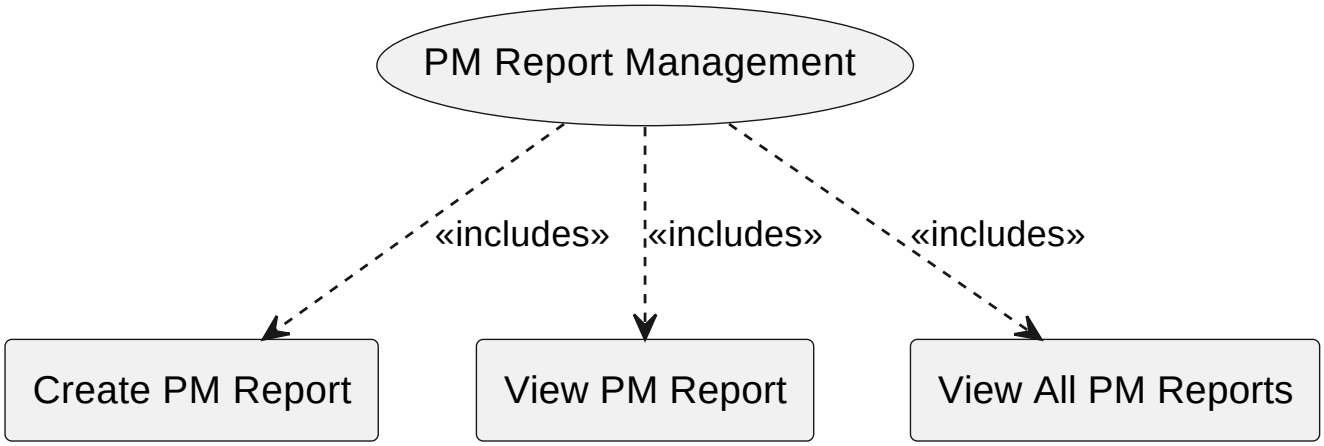

Figure 5.144: **TOD-06-05**: Performance Monitoring Report Management

	Customer	Provider	Other PSS	Governance
Create PM Report	✓		✓	
View PM Report	✓		✓	✓
View All PM Reports	✓		✓	✓

Table 5.24: Performance Monitoring Report Management Matrix.

eTOM Reference

The task is based on the 1.4.7 process identifier from the eTOM.

5.6.5.1 TOD-06-05-01-Create_Performance_Monitoring_Report

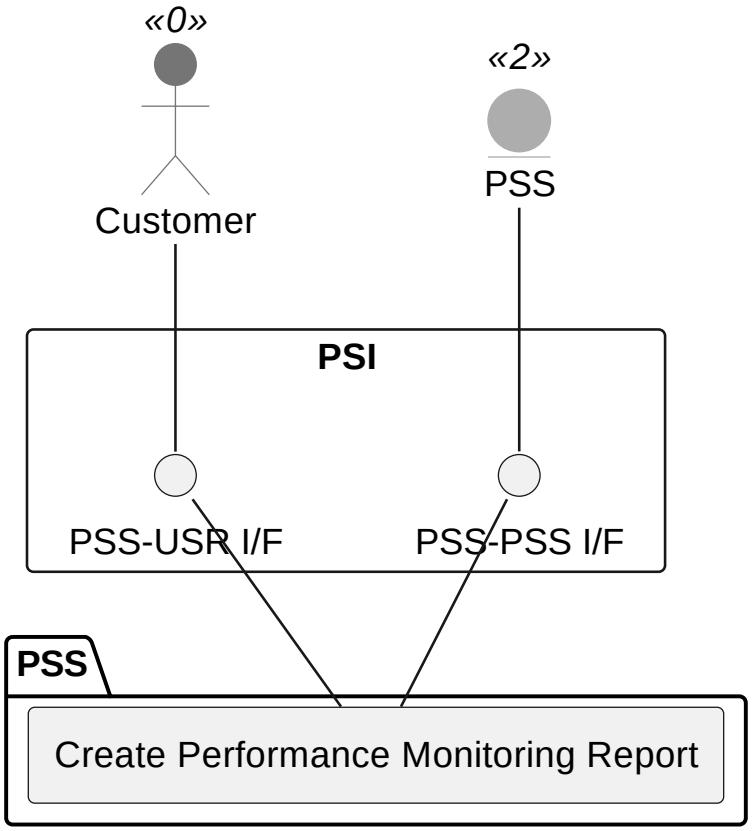

Figure 5.145: **TOD-06-05-01**: Create Performance Monitoring Report

Prerequisites

Service for which PM Report needs to be generated exists in the PSS datastore. Applicable key indicators are defined in the PSS datastore.

Main operation

Creates a new Performance Monitoring Report instance via a standard interface.

Some properties of a Performance Monitoring Report are:

- *granularity* - Sampling rate of the collection of performance indicators, e.g. 10 milliseconds, 1 second, 5 minutes, 24 hours
- *outputFormat* - Output formats for the Performance Report: json, xml, avro, csv
- *reportingTimeframe* - Defines the interval for data to be included in the report
- *resultFormat* - Defines how the provider will deliver the Performance Report to the PSS:
- payload data will be delivered directly in the “Get Report” API response
- attachment data will be stored in the file and the API response will contain a URL to the file

- *scheduleDefinition* - The schedule definition for running jobs
- *servicePayloadSpecificAttributes* - an extension point for service-specific performance monitoring configuration. It includes a definition of observed service and applicable performance monitoring objectives.

REST Endpoints

- POST /performanceMonitoring/v1/performanceReport

Post Conditions

The PM report is successfully created in the PSS datastore.

Applicable Requirements

- PSI-06-05-01-01
- PSI-06-05-01-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.5.2 TOD-06-05-02-View_Performance_Monitoring_Report

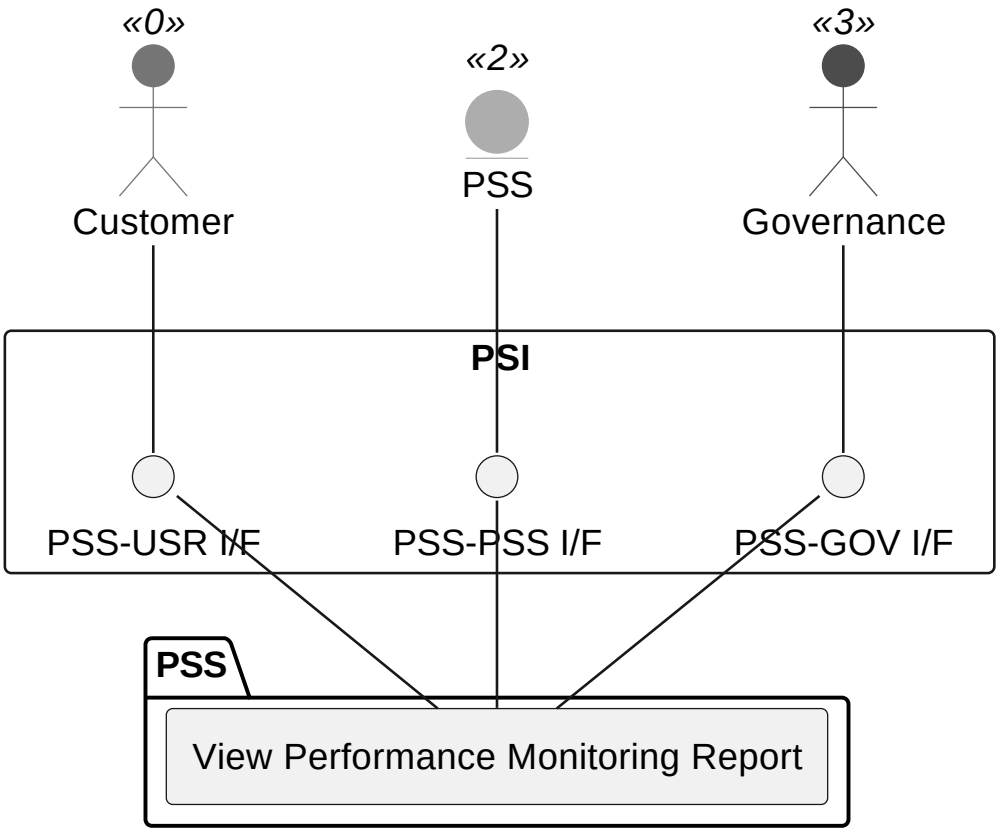

Figure 5.146: **TOD-06-05-02**: View Performance Monitoring Report

Prerequisites

The Performance Monitoring Report exists in the PSS datastore.

Main operation

Gets a Performance Monitoring Report instance via a standard interface.

REST Endpoints

- GET /performanceMonitoring/v1/performanceReport/{id}

Post Conditions

The Performance Monitoring Report is successfully returned for viewing.

Applicable Requirements

- PSI-06-05-02-01
- PSI-06-05-02-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.5.3 TOD-06-05-03-View_All_Performance_Monitoring_Report

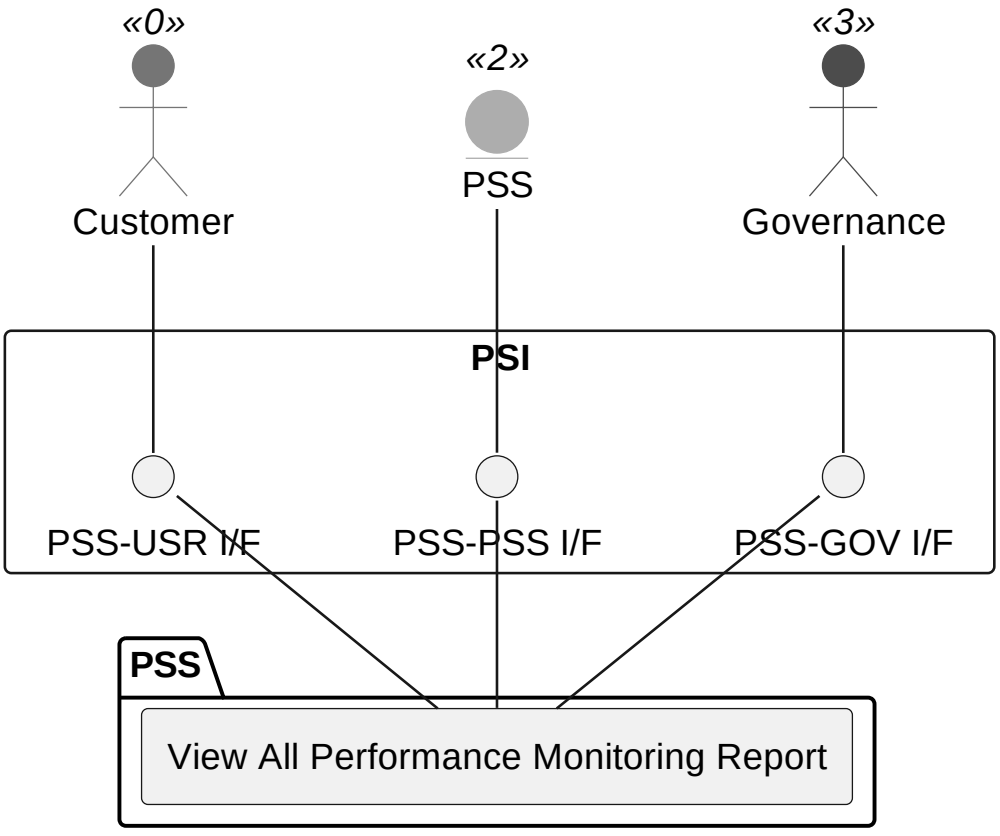

Figure 5.147: **TOD-06-05-03**: View All Performance Monitoring Report

Prerequisites

The Performance Monitoring Report exists in the PSS datastore.

Main operation

Gets all Performance Monitoring Report instances via a standard interface.

REST Endpoints

- GET /performanceMonitoring/v1/performanceReport
- POST /performanceMonitoring/v1/performanceReportComplexQuery

Post Conditions

All visible Performance Monitoring Reports are successfully returned for viewing.

Applicable Requirements

- PSI-06-05-03-01
- PSI-06-05-03-02

eTOM Reference

The operation is based on the 1.4.7 process identifier from the eTOM.

5.6.6 TOD-06-06-Alarm_Management

The Alarm Management task takes care of the maintenance of alarms in the PSS. Alarms are generated by a service provider as a consequence of crossing thresholds defined by the service level objectives defined for the service (see TOD-06-01). The following diagram illustrates a typical sequence leading to the creation of an alarm.

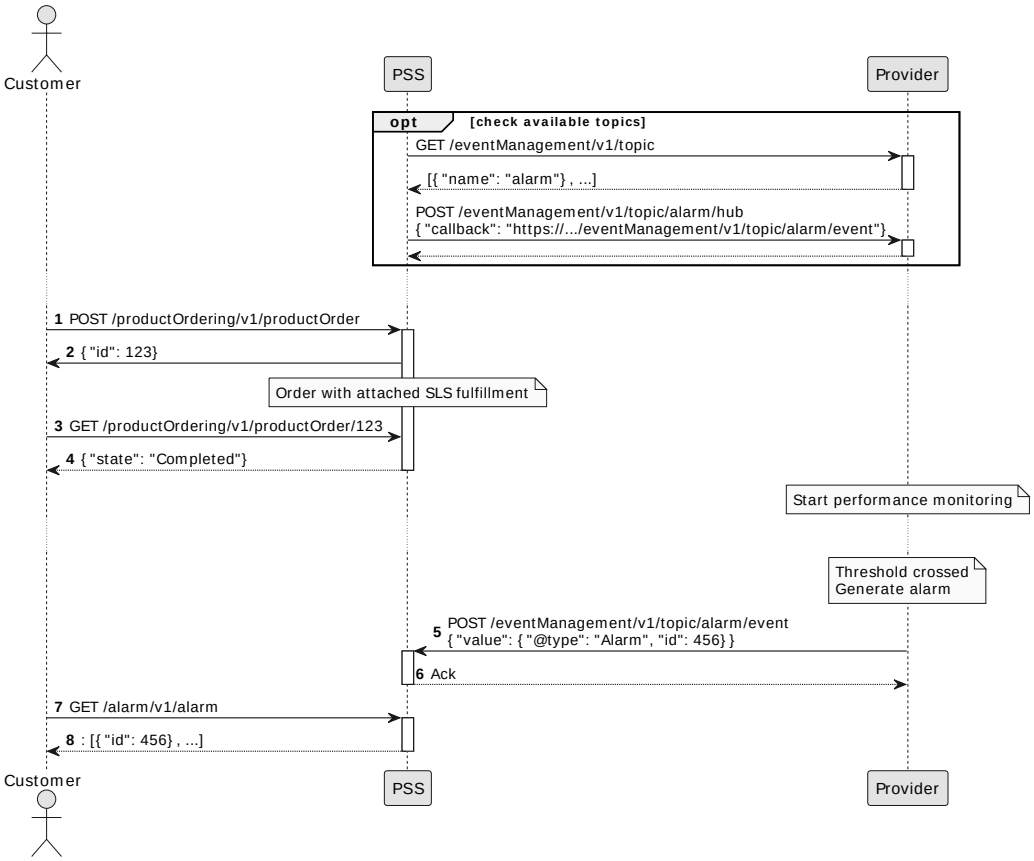

Figure 5.148: **TOD-06-06**: Alarm Management Sequence

The PSS is subscribing for events related to alarm generation. This provides a callback address to which service provider should send alarms related to services (see TOD-01-02). When customer of the PSS sends an order for a product with SLS attached, provisioning of the service will also start performance monitoring on the service provider side. In case service provider detects that service level objective threshold was crossed and the prescribed action requires an alarm, it will generate one. This will in turn cause creation of a relevant event that will be pushed to PSS with all details of the alarm.

Steps related to alarms are further described in the following operations:

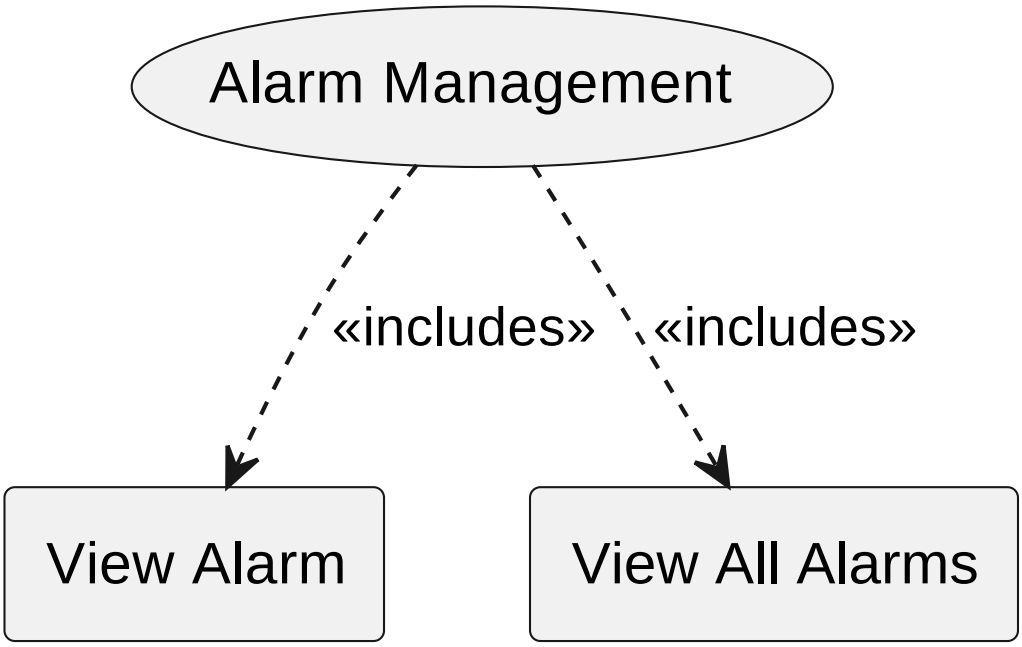

Figure 5.149: **TOD-06-06**: Alarm Management

	Customer	Provider	Other PSS	Governance
View Alarm	✓	✓	✓	✓
View All Alarms	✓	✓	✓	✓

Table 5.25: Alarm Management Matrix.

eTOM Reference

The task is based on the 1.4.6 and 1.5.8 process identifier from the eTOM.

5.6.6.1 TOD-06-06-01-View_Alarm

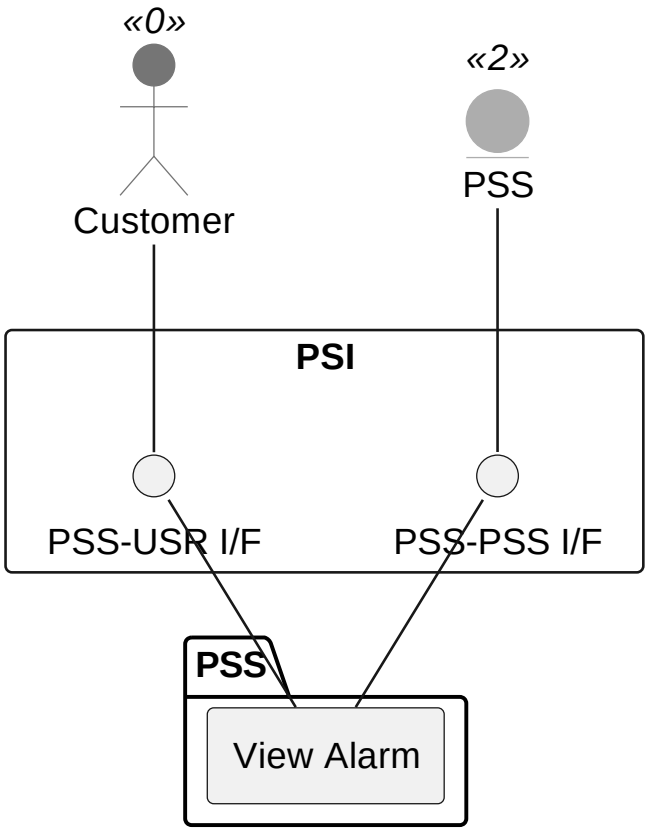

Figure 5.150: **TOD-06-06-01**: View Alarm

Prerequisites

The Alarm exists in the PSS datastore.

Main operation

Gets an Alarm instance via a standard interface.

REST Endpoints

- GET /alarmManagement/v1/alarm/{id}

Post Conditions

The Alarm is successfully returned for viewing.

Applicable Requirements

- PSI-06-06-01-01
- PSI-06-06-01-02

eTOM Reference

The operation is based on the 1.4.6 and 1.5.8 process identifier from the eTOM.

5.6.6.2 TOD-06-06-02-View_All_Alarm

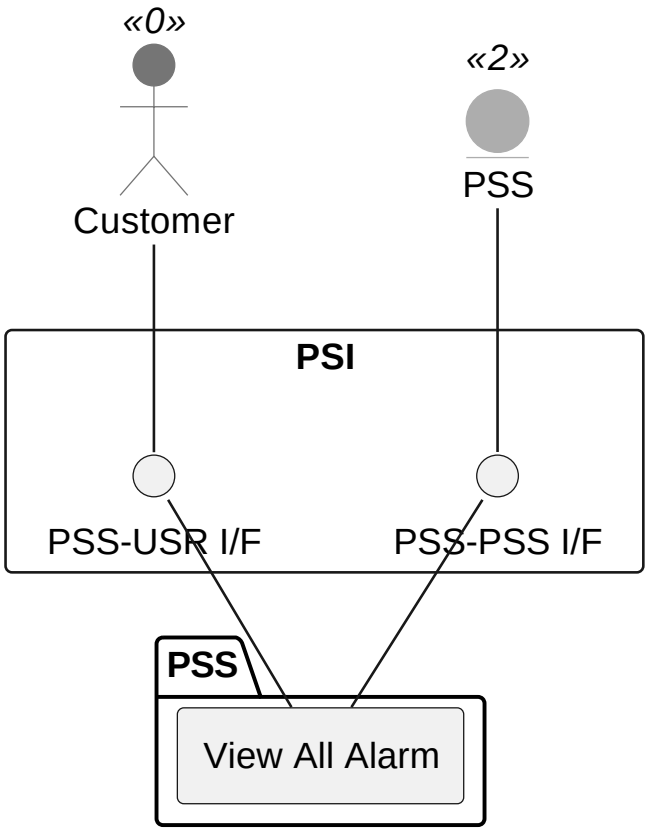

Figure 5.151: **TOD-06-06-02**: View All Alarm

Prerequisites

The Alarm exists in the PSS datastore.

Main operation

Gets all Alarm instances via a standard interface.

REST Endpoints

- GET /alarmManagement/v1/alarm

Post Conditions

All visible Alarms are successfully returned for viewing.

Applicable Requirements

- PSI-06-06-02-01

eTOM Reference

The operation is based on the 1.4.6 and 1.5.8 process identifier from the eTOM.

Last Page of Document