

SFTI Item Availability Check Service Specification

Date: 2021-06-21

Version: 1.1.0

Executive Summary

This document describes and defines SFTI Item Availability Check Service from business, semantical and technical points of view.

The purpose of the SFTI Item Availability Check Service is to enable customers, in the buyer role, to check the availability and possible delivery date(s) of items/articles in a seller's catalogue. After receiving the information based on the submitted query, the buyer can place an order knowing that the seller is in a better position to effectuate it.

It should be stressed that the SFTI Item Availability Check Service represents a pre-order process. The benefits of this process includes fewer roundtrips in the ordering phase that follows (i.e. less order – order response calls) and thus less time spent on executing an order.

The SFTI Item Availability Check Service is built on top of the Peppol and UBL standards and, for technical reasons, it is designed to complement the Peppol BIS Ordering 3 and the Order, Order Response messages¹.

¹ In this specification the words message and transaction are used as synonyms

Table of Content

Executive Summary	2
Table of Content	3
1 Introduction	4
1.1 Structure of specification.....	4
1.2 Intended Audience	4
2 Governance and Management	7
2.1 Versions and Releases	7
2.2 Change and Release Log.....	7
2.3 Terms of Use and Statement of Copyright.....	7
2.4 References	7
3 Business Specification	9
3.1 Benefits.....	9
3.2 Business Services	9
3.3 Parties and Roles	9
3.4 Business Requirements and Agreements	10
3.5 Business Security and Risks	10
4 SFTI Item Availability Check Service	11
4.1 Service interactions	11
4.2 Service Requirements	11
4.3 Access points / Service discovery / Address Lookup.....	11
4.4 Use cases	12
5 Semantic Model	14
5.1 Message: Check Request	14
5.2 Message: Check Response.....	18
5.3 Data Models	21
6 Technical API Specification	23
6.1 Description	23
6.2 Technical Security and Risk.....	23
6.3 Open API Specification.....	23
6.4 Technology Bindings	25
7 Release Log	26

1 Introduction

This specification is the result of work within SFTI and is published as part of the SFTI collection of recommendations, and specifications. The document provides a specification for implementing the *SFTI Item Availability Check Service*.

The purpose of the SFTI Item Availability Check Service is to enable customers, in the buyer role, to check the availability and possible delivery date(s) of items/articles in a seller's catalogue. When the potential buyer has received the information based on the submitted query, the user can plan the ordering according to availability of items. The benefits of using this pre-order process includes fewer roundtrips in the ordering process (i.e. less order – order response calls) and thus less time spent on executing an order.

The SFTI Item Availability Check Service requires that the parties, buyer and seller, have established a business agreement before the process is performed.

The *technical API*² part of the Specification adheres to the Open API 3.0 specification (OAS) and is documented and published in the Swagger Hub³. The Open API Specification (OAS) defines a standard, programming language-agnostic interface description for REST⁴ APIs. Swagger HUB is an open source tool used to design, build and document API's based on the Open API. See section 6, Technical API Specification, for details.

1.1 Structure of specification

The structure of this specification consists of five substantial parts.

1. Governance and management part (section 2) that describes how this document is managed, version handled, etcetera.
2. Business specification part (section 3) that describe the process, parties and roles, and business requirements from business and organisational interoperability perspectives.
3. Technical part that describes the Availability Check Service, with use cases, (section 4) from an ICT interoperability perspective.
4. Semantical model (section 5) that describes Request and Response messages with bindings to syntax and code lists.
5. Technical API Specification, comprising of an introductory description (section 6 of this document) and of the formal Open API 3.0 Specification for the SFTI Item Availability Check Service found in Swagger Hub.

1.2 Intended Audience

The intended audience for this document is organizations wishing to use SFTI specifications, recommendations and standards to exchange electronic business documents.

² An application programming interface, API, is a computing interface which defines interactions between multiple software components, for example it defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc.

³ <http://swagger.io>

⁴ Representational State Transfer, REST, is a set of rules/constraints followed when creating APIs.

These organizations may be:

- Service providers
- Contracting Authorities
- Economic Operators
- Software Developers

More specifically the document is addressed towards the following roles:

- SFTI
- Business Analyst and EDI administrator
- ICT Architect and ICT Developer

This document aims to support the interested parties in the following work they do and questions they need answered.

Business Analyst and EDI administrator

Work to be done:

- a) Evaluate business specifications and interoperability from legal, business concept, organisational, semantic, and business security point of views.
- b) Achieve interoperability from a legal, business concept, organisational, semantic, and business security point of views.
- c) Assure functional quality of eProcurement and the ordering process.
- d) Ensure handling of business errors and exceptions.
- e) Evaluate and manage business agreements and risks.
- f) Develop business requirements for development of EDI and business systems.
- g) Manage processes, services and business specifications that are version handled.
- h) Decide if the specification, its services and business rules deliver benefits, incur reasonable risks, and sufficiently fit with the organisations security policy.

ICT Architect and Developer

Work to be done:

- a) Evaluate technical API specifications and interoperability from technical, technology, data and technical security point of views.
- b) Achieve interoperability from a technical, technology, data and technical security point of views.
- c) Assure functional and technical quality of eProcurement and ordering process systems.
- d) Ensure handling of technical errors and exceptions.
- e) Manage technical agreements and risks.
- f) Develop technical requirements for development of the technical systems.
- g) Develop software using the technical API.
- h) Automatically generate software based on formal API specification.
- i) Test technical system.
- j) Automatically generate software for testing.
- k) Manage technical specifications that are version handled.
- l) Decide if the specification, its services and technical rules incur reasonable technical risks, and sufficiently fit with the organisations security policy.

SFTI

Work to be done:

- a) Improve eProcurement and the ordering process for buyers and sellers.

- b) Develop and manage guidelines, recommendations and standards that cover legal, business concept, organisational, semantical and technical point of views.
- c) Provide quality assurance support for implementors of SFTI guidelines, recommendations and standards.
- d) Manage version handled specifications that depend on underlying recommendations, standards and specifications.

2 Governance and Management

The specification of the SFTI Item Availability Check Service comprises two parts: this document and the technical API specification located in Swagger Hub. Both are needed to understand the Service and they are to be seen as an integral document.

The documentation is governed, and lifecycle managed, by Single Face to Industry (SFTI). Questions relating to this specification can be directed to SFTI Technical Secretariat, tekniskt.kansli@skr.se.

2.1 Versions and Releases

The specification of the Service is based on major, minor and revision versioning principles, expressed as major.minor.revision, for example “Version 1.1.1”. The two parts of documentation (i.e. this document and the technical API specification located in Swagger Hub) are dependently versioned at the level of major.minor, but at the level of “revision” they may differ.

This document and API specification are released according to SFTI release cycle principles.

2.2 Change and Release Log

Changes to this document, technical API specification located in Swagger Hub, and to the underlying standards and specifications are documented in the Release Log (section 0) in this document.

No life cycle management efforts exist between this specification and referenced business and technical documents. Thus, any significant changes in the underlying sources are not automatically reflected in changes that are being implemented in this specification.

2.3 Terms of Use and Statement of Copyright

© 2020, *Sveriges Kommuner och Regioner (The Swedish Association of Local Authorities and Regions)*.

2.4 References

Link to main site of documentation: <http://www.sfti.se/>

Reference to integral parts: the technical API specification

Item	Link / Source	Comment
API specification	http://app.swaggerhub.com/apis-docs/sfti-se/check-item-availability/1.1	The Swagger Hub part of the technical API

References to syntax binding used for this specification

Item	Link / Source	Comment
Peppol BIS Ordering 3	<p>Peppol BIS Ordering 3: http://docs.peppol.eu/poacc/upgrade-3/profiles/28-ordering/</p> <p>Peppol BIS Order transaction 3: http://docs.peppol.eu/poacc/upgrade-3/syntax/Order/tree/</p> <p>Peppol BIS Order response transaction 3: http://docs.peppol.eu/poacc/upgrade-3/syntax/OrderResponse/tree/</p>	Defines the Order and Order Response activities on which the Item Availability Check is aligned with.
UBL 2.1	http://docs.oasis-open.org/ubl/UBL-2.1.html	The basis for data model of this specification.
UNCL 1229 code list	http://www.unece.org/fileadmin/DAM/trade/untdid/d18a/tred/tred1229.htm	Response codes used in the API Response message

References to technical API parts that this specification depends on

Item	Link / Source	Comment
HTTP error codes	http://developer.mozilla.org/en-US/docs/Web/HTTP/Status	Provides a list of the technical error codes that can be used to explain why a call fails due to technical issues.
Open API	http://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md	Explains the OpenAPI concept
Swagger Hub	http://swagger.io	Describes the Swagger API concept.

Other references

Item	Link / Source	Comment
SFTI Technical Secretariat	tekniskt.kansli@skr.se	Email to SFTI Technical Secretariat, may be used for questions about this specification.

3 Business Specification

The primary purchasing process, advocated by SFTI for procurement of standardised articles (products and services) under framework agreement, presumes that the buyer maintains a copy of the seller's catalogue. The catalogue data, as appropriate adapted to the requirements of the framework agreement, are transferred from the supplier to the buyer's purchase system by means of a transaction called Peppol BIS Catalogue without response (or some similar format). Through this, the buyer can ensure that the catalogue is consistent with the procurement specifications, that articles are presented in a supplier-neutral way, and that purchasers observe the framework agreement(s) – all of these vital elements for sound procurement practice. However, certain challenges exist when it comes to maintaining an updated catalogue in the buyer's system, examples are lack of data on supply status and availability.

This is addressed through the present document, introducing specifications for an application programming interface (API) for item availability checking. A request-response pair of messages is introduced through which the buyer catalogue can be supplemented with up-to-date details on inventory and availability status for specific items/articles. This is pre-order process, targeting the select items/articles being contemplated for a specific order. Status information has limited lifespan and, as no reservation is implied by the availability check, the order (in the form of Peppol BIS ordering format) should follow without delay.

Note the similarities to working with the Peppol BIS Punch Out 3. One main difference with Punch Out is that the buyer is then working with the catalogue in the seller's system.

3.1 Benefits

The benefits of using this pre-order service includes fewer roundtrips in the ordering process (i.e. less order – order response calls) and thus less time spent on executing an order.

3.2 Business Services

This specification defines the following business services:

- SFTI Item Availability Check Service

3.3 Parties and Roles

The table below gives the definitions of the parties and roles of this specification.

Business parties	Description
Customer	The customer is the legal person or organization who is in demand of a product or service. Examples of customer roles: buyer, consignee, delivery party, debtor, contracting authority, originator. [Source: Peppol BIS Ordering 3]

Supplier	The supplier is the legal person or organization who provides a product or service. Examples of supplier roles: seller, consignor, creditor, economic operator. [Source: Peppol BIS Ordering 3]
Role / actor	Description
Buyer	The buyer is the legal person or organization acting on behalf of the customer and who buys or purchases the goods or services. [Source: Peppol BIS Ordering 3]
Seller	The seller is the legal person or organization acting on behalf of the supplier and who sells goods or services to the customer. [Source: Peppol BIS Ordering 3]

3.4 Business Requirements and Agreements

The SFTI Item Availability Check Service requires that:

- a) The buyer has access to the seller's catalogue over products and/or services, customised to his needs, either in the own purchasing system or through a third-party catalogue provider.
- b) The buyer and seller have an agreement to use the API service to complement the catalogue with availability information before the service is being performed.

3.5 Business Security and Risks

Those implementing the API Specification should consider risk of unintentionally exposing business data to competitors.

4 SFTI Item Availability Check Service

The SFTI Item Availability Check Service is a service that enables customers, in the buyer role, to check the availability and possible delivery date(s) of items/articles in a seller's catalogue.

4.1 Service interactions

This SFTI Item Availability Check Service is a pre-order process. When planning for an order, the buyer may wish to first verify that the items contemplated are available from the seller, before placing the actual order. Information on availability is needed notably for time-critical supplies and where there are dependencies between items, calling for joint and delivery, but it may also be a means to reduce administrative hassle due to order change and part deliveries. The Availability Check Service gives the buyer assurance of a more efficient and effective outcome of the ordering process that follows, but the ordering process itself is not affected.

From a technical perspective the pre-order and ordering processes have many similarities. The actors and party roles described in Peppol BIS Ordering 3 apply to this pre-order process, and subsets of the Peppol Order and Order Response transactions function as data models for the Availability Check Request and Availability Check Response.

The business service supports the following request types:

- Full Check (see section 4.4.1)
- Simple Check (see section 4.4.2).

4.2 Service Requirements

Implementors of SFTI Item Availability Check Service can support one or both request types specified for this service. Based on the capabilities of the supplier, the buyer should configure the application to only invoke/use the appropriate type.

The type of request (full/simple) is indicated by the structure of the resource URL (see section 6.3.2 Resources). The seller responds with HTTP Status code 403 if a request type is used which isn't supported (see section 6.3.4 Responses).

4.3 Access points / Service discovery / Address Lookup

There is currently no dynamic discovery of end point addresses. The location of the end point (URL) for the API must be communicated bilaterally.

4.4 Use cases

4.4.1 Use case - Full Check

Use case: Full Check	
Description	The buyer checks the availability of one or more items, listed in a catalogue. This enables the buyer to ensure that the items are available in the seller's system before ordering the items.
Parties /roles involved	Buyer, Seller
Assumptions	A copy of the seller's catalogue over products or services is available in the buyer's purchasing system (or through a third-party catalogue provider).
Flow	<p>1. The buyer creates and sends an Availability Check Request (using the request type <i>full</i> in the URL) with the following information entities:</p> <ul style="list-style-type: none"> • <i>Requested delivery location</i> (on the Request message level) • <i>Quantity</i> (on the line level, i.e. per item) • <i>Requested delivery period</i> (on the message level or on the line level, i.e. per item). <p>2. The Seller replies, using an Availability Check Response, with availability information about each requested item. If an item is out of stock the seller may propose a substitute item.</p>
Result	The response received from the seller's system contains either the confirmation that the item is available or a rejection that the item is not available as requested; if applicable a substitute item may be proposed. Note that the seller makes no reservation of items based on the Availability Check Request.
Example documents	-

4.4.1.1 Business Rules

If *Requested delivery period* is stated on the level of Check Request message the period applies to all items in the Check Request. If a deviating *Requested delivery period* is given for an item, period on the line item level applies only to that item.

Requested delivery period can be narrowed down to *Requested delivery date* by setting end date equal to start date.

A *Response code* is to be present for each item in the Check Response, either confirming that the item is available as requested or rejecting the request. Depending on the nature of rejection an explanatory text may be provided in *Response clarification* and, in the case the seller can propose a substitute item, details of the proposal are given in the group of business terms named *Seller proposed substitution item*.

If any the elements *Requested delivery location*, *Quantity* or *Requested delivery period* are missing, the seller returns *Reason code 70* (Pending because of incomplete information) to indicate that more information needs to be supplied.

The Response code values, and their meanings, are described in section 5.3.2.

4.4.2 Use case – Simple Check

Use case: Simple Check	
Description	The buyer wants to know if some specific products/items are available for immediate delivery. An Availability Check Request is created based on the respective item identifications in the catalogue. In this case, both <i>Quantity</i> and <i>Requested delivery period</i> are excluded from items listed in the Availability Check Request. An estimate of how much the supplier currently has in stock for the requested item(s) is returned.
Parties /roles involved	Buyer, Seller
Assumptions	A copy of the seller’s catalogue over products or services is available in the buyer’s purchasing system (or through a third-party catalogue provider).
Flow	1. The buyer creates and sends an Availability Check Request (using the request type <i>simple</i> in the URL) listing the item(s) for query. If applicable, the request may be restricted to a <i>Requested delivery location</i> . 2. The Seller replies with an Availability Check Response showing availability of each Item in the <i>Response clarification</i> (“Note”).
Result	The exact or estimated quantity (at the seller’s discretion), showing what is currently in stock for each item in the Availability Check Request. Note that the seller makes no reservation of items based on the Availability Check Request.
Example documents	-

4.4.2.1 Business Rules

If an Item request is sent without specifying *Quantity* and *Requested delivery period*, the Quantity for the Item is entered into the *Response clarification* element in the Response. Reference to quantity should include unit of measure, as applicable. Please note that the quantity in *Response clarification* is not necessarily expressed as a numeric figure.

The elements *Quantity* or *Requested delivery period* are not used in request type simple and if present, can be ignored by the Seller.

For details on the Response code, see section 5.3.2.

Table 1 Semantic model of the Availability Check Request bound to a subset of Peppol Order 3 (T01)

Card	UBL 2.1 Term name/ Order	Business Term – Name and Description
1..1	ubl:Order	SFTI Item Availability Check Request (short form: Check Request) <i>The Availability Check Request contains the information that the buyer provides through the API when inquiring about availability of articles in a pre-order process.</i>
1..1	• ID	Identifier <i>A transaction instance must have an identifier. The identifier enables referencing the transaction for various purposes, such as from other transactions that are part of the same process.</i> Example value: 123
1..1	• IssueDate	Issue date <i>The date on which the transaction instance was issued.</i> Example value: 2019-12-19
1..1	• IssueTime	Issue time <i>The time when the transaction instance was issued.</i> Example value: 14:39:00
1..1	• BuyerCustomerParty	Buyer information <i>Description of buyer</i>
1..1	• • Party	Party information
1..1	• • • PartyIdentification	Party identification
1..1	• • • • ID	Buyer party identification <i>An identification for the buyer party.</i> Example value: 7300010000001
1..1	• SellerSupplierParty	Seller information <i>Description of seller</i>
1..1	• • Party	Party information
1..1	• • • PartyIdentification	Party identification
1..1	• • • • ID	Seller party identification <i>Identifies the seller party.</i> Example value: 7300010000001
0..1	• Delivery	Delivery information
0..1	• • DeliveryLocation	Requested delivery location

Card	UBL 2.1 Term name/ Order	Business Term – Name and Description
1..1	• • • ID	Delivery location ID <i>Identification of the delivery location to which the whole Availability Check Request refer.</i> Example value: 7300010000001
0..1	• • RequestedDeliveryPeriod	Requested delivery period
1..1	• • • StartDate	Period start date. <i>Identification of the start date to which the Availability Check Request refer. The start date counts as part of the period. Format = "YYYY-MM-DD"</i> Example value: 2019-12-19
1..1	• • • EndDate	Period end date <i>Identification of the end date to which the Availability Check Request refer. The end date counts as part of the period. Format = "YYYY-MM-DD"</i> Example value: 2019-12-19
1..n	• OrderLine	Check Request line <i>Specification of Check Request line</i>
1..1	• • LineItem	Line item <i>Description of line item</i>
1..1	• • • ID	Line item identifier <i>Identifies the Check Request line item assigned by the buyer; the identifier must be unique within the Check Request.</i> Example value: 1
0..1	• • • Quantity	Quantity <i>The quantity of an item inquired about in the Check Request line item.</i> <i>Relevant for request type: full</i> Example value: 5.7
M	• • • • @unitCode	Quantity unit of measure <i>The unit of measure that applies to the Check Request line item quantity.</i> Example value: LTR
0..1	• • • Delivery	Line delivery information <i>Delivery information applicable for the Check Request line item. When stated, this information supersedes any corresponding information on document level for the line item in question.</i>

Card	UBL 2.1 Term name/ Order	Business Term – Name and Description
		<i>Relevant for request type: full</i>
1..1	• • • • RequestedDeliveryPeriod	Requested delivery period <i>Requested delivery period for the Check Request line item.</i>
1..1	• • • • • StartDate	Period start date <i>The date on which the period starts. The start date counts as part of the period. Format = "YYYY-MM-DD"</i> Example value: 2019-12-19
1..1	• • • • • EndDate	Period end date <i>The date on which the period ends. The end date counts as part of the period. Format = "YYYY-MM-DD"</i> Example value: 2019-12-19
1..1	• • • Item	Item information
0..1	• • • • SellersItemIdentification	Sellers item identification
1..1	• • • • • ID	Seller's item identifier <i>An identifier, assigned by the seller, for the item. Associates the item with its identification according to the seller's system.</i> Example value: 3249834
0..1	• • • • StandardItemIdentification	Standard item identification
1..1	• • • • • ID	Item standard identifier <i>An item identifier based on a registered scheme. Associates the item with its identification according to a standard system, such as GSI GTIN.</i> Example value: 05704066204093

5.2 Message: Check Response

The SFTI Item Availability Check Response, used in both the Full Check and Simple Check use cases, comprises the following information. For the representation as a transaction, the format is bound to a subset of Order Response defined in Peppol BIS Ordering 3.

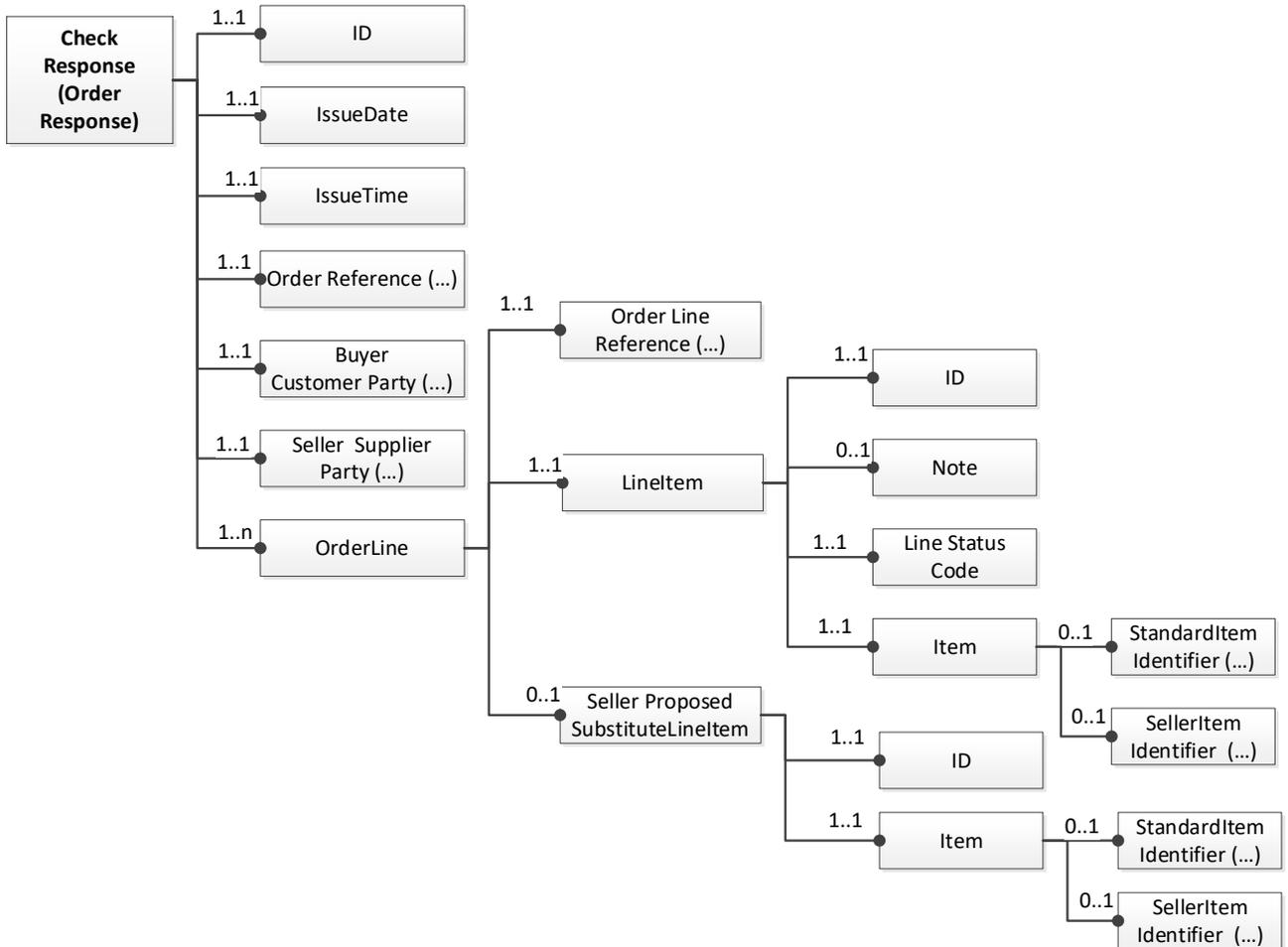


Figure 3 Semantic of model for Availability Check Response

Table 2 Semantic model of Availability Check Response bound to a subset of Peppol Order Response 3 (T76)

Card	UBL 2.1 Term name/Order response	Business Term – Name and Description
1..1	ubl:OrderResponse	SFTI Item Availability Check Response (short form: Check Response) <i>The response to the Item Availability Check Request, in a pre-order process, submitted to the API by the seller's system.</i>
1..1	• ID	Identifier <i>A transaction instance must have an identifier. The identifier enables referencing the transaction for various</i>

Card	UBL 2.1 Term name/Order response	Business Term – Name and Description
		<i>purposes, such as from other transactions that are part of the same process.</i> Example value: 123
1..1	• IssueDate	Issue date <i>The date on which the transaction instance was issued.</i> Example value: 2019-12-19
1..1	• IssueTime	Issue time <i>The time on which the transaction instance was issued</i> Example value: 14:39:00
1..1	• OrderReference	Check Request reference
1..1	• • ID	Check Request document reference <i>Used to reference the transaction that is being responded to.</i> Example value: 123
1..1	• SellerSupplierParty	Seller information
1..1	• • Party	Description of seller
1..1	• • • PartyIdentification	Seller identification
1..1	• • • • ID	Seller party identification <i>Identifies a seller party.</i> Example value: 123456
1..1	• BuyerCustomerParty	Buyer information
1..1	• • Party	Description of buyer
1..1	• • • PartyIdentification	Buyer identification
1..1	• • • • ID	Buyer party identification <i>An identification for the buyer party.</i> Example value: 98765
1..n	• OrderLine	Check Response line
1..1	• • OrderLineReference	Reference to Check Request line
1..1	• • • LineID	Check Request line reference <i>Used to reference the Check Request line that corresponds to the response line.</i>

Card	UBL 2.1 Term name/Order response	Business Term – Name and Description
		Example value: 34
1..1	• • LineItem	Description of line item
1..1	• • • ID	Document line identifier <i>Identifies the response line. Each Check Response line MUST be unique within the Check Response.</i> Example value: 3
0..1	• • • Note	Response clarification <i>Clarification of the supplier's item status intended to be presented to a user in a clear and understandable way. It may contain quantity information (for request type Simple Check) or an explanation linked to the Response code.</i> <i>Common rejection/deviation reasons can be found in GSI codelist "T0211 Avvikelseorsak".</i> Example value (Simple check): >50 units currently available Example value (Full check): Stock shortage, 90 units available Example value (Full check): Item not in stock, available for order 2021-06-30
1..1	• • • LineStatusCode	Response code <i>A code that indicates the availability of the item identified in the referenced Check Request line.</i> <i>See actions codes from the subset of UN/CEFACT UNCL 1229 in section 5.3.1.</i> Example value: 5
1..1	• • • Item	Item information
0..1	• • • • SellersItemIdentification	Sellers item identification
1..1	• • • • • ID	Seller's item identifier <i>An identifier, assigned by the seller, for the item.</i> <i>Associates the item with its identification according to the seller's system.</i> Example value: 3249834
0..1	• • • • StandardItemIdentification	Standard item identification
1..1	• • • • • ID	Item standard identifier <i>An item identifier based on a registered scheme.</i> <i>Associates the item with its identification according to a standard system.</i> Example value: 07310865010087

Card	UBL 2.1 Term name/Order response	Business Term – Name and Description
0..1	• • SellerProposedSubstituteLineItem	Seller proposed substitution item <i>Item proposed as substitute by the seller if the requested item is unavailable.</i>
1..1	• • • ID	Substituted Line Item Identifier <i>Identifier of the substituted line item</i> Example value: 123
1..1	• • • Item	Item information
0..1	• • • • SellersItemIdentification	Sellers item identification
1..1	• • • • • ID	Sellers item identifier <i>An identifier, assigned by the seller, for the item. Associates the item with its identification according to the seller's system.</i> Example value: 3249834
0..1	• • • • StandardItemIdentification	Standard item identification
1..1	• • • • • ID	Item standard identifier <i>An item identifier based on a registered scheme. Associates the item with its identification according to a standard system, such as GSI GTIN.</i> Example value: 17300015200205

5.3 Data Models

5.3.1 Relation to technical formats

The semantic model for the messages forming the SFTI Item Availability Check Service is technically aligned to subsets of Peppol BIS Ordering 3 – the Order transaction (T01) and the Order Response Transaction (T76) – using UBL version 2.1 as syntax. The detailed representation of this alignment is shown in Tables 1 and 2 (above).

5.3.2 Response codes

The Availability Check Response message uses response codes that are drawn from a subset of code list UN/CEFACT UNCL 1229 to indicate the seller's judgement of the availability request for the specific item. The business term *Response code* is placed in the *LineStatusCode* data element and, as needed, the *Response code* can be further qualified (or explained) by text in the *Response clarification* (i.e. the *Note* data element).

Response code (LineStatusCode)	Meaning	Remark
5	Accepted without amendment	Reference: Peppol BIS Ordering 3, UNCL1229. The item of the Check Request is accepted without changes. This means that the item is available and can be delivered, as applicable, to the <i>Requested delivery location</i> and within the <i>Requested delivery period</i> (or delivery date, if narrowed down).
6	Accepted with amendment	Reference: UNCL1229. This item of the Check Request is accepted but amended by the seller by a proposed substitute item. This <i>Response code</i> requires the the business terms for <i>Seller proposed substitution item</i> to be provided in the Check Response.
7	Not accepted	Reference: Peppol BIS Ordering 3, UNCL1229. The item of the Check Request is rejected. Reasons can be that the item is not in stock, or cannot be delivered to the <i>Requested delivery location</i> , or not available within the <i>Requested delivery period</i> (or delivery date, if narrowed down).
10	Not found	Reference: UNCL1229. This item is not found in the supplier's catalogue.
70	Pending, incomplete	Reference: UNCL1229. Pending because of incomplete information. To be used when more information is expected, such as <i>Requested delivery location</i> , or if the buyer's <i>ID</i> is invalid. The seller will take no further action but would expect a complete/ corrected Availability Check Request.

6 Technical API Specification

6.1 Description

This section of the specification provides technical details that enable ICT architects and developers to design, develop and test ICT solutions and products based on the functional business aspects in described in this specification.

The technical API specification consists of two parts, this document and an Open API 3.0 interface that is documented and published in the Swagger Hub. When working with the two document parts, please verify that the version and release numbers of the documentation is consistent, c.f. section 2.1 above.

The OpenAPI Specification (OAS) defines a standard, programming language-agnostic interface description for REST APIs, which allows both humans and computers to discover and understand the capabilities of a service without requiring access to source code, additional documentation, or inspection of network traffic. [Source: Open API]

The Swagger HUB is an open source tool used to design, build and document API's based on the Open API specification and JSON syntax.

6.2 Technical Security and Risk

See section 3.53.5 above.

6.3 Open API Specification

This section provides additional Open API specific details of the technical API. See Swagger Hub for full Open API specification.

6.3.1 Operations

API Timeout

The suppliers (buyers) determine and set the requested API timeout.

Recommendation: Set the API timeout dynamically, given the characteristics of the actual data exchanged through the API, such as the number of items requested.

6.3.2 Resources

This technical API specifies one single resource structured in the following way:

1. 2. 3. 4. 5. 6.
<https://www.example.com/any/structure/sfti-api/check-item-availability/{RequestType}/1.1>

1. The domain name
2. Optional: Any folder/sub structure
3. Required: sfti-api

- 4. Required: API-name
- 5. Required: RequestType – full | simple
- 6. Required: Version number

The purpose of a pre-defined URL-structure is that only first parts of the (section 1 and 2) needs to be configured by the buyer for a specific supplier. A pre-defined URL can eliminate errors and simplify the setup.

6.3.3 Parameters

Not applicable for this specification.

6.3.4 Responses

The API provides two types of return messages:

- a) The first type is HTTP response status codes that indicate status of a message request when using the API at HTTP level. The HTTP error status codes defined for this API are documented in the API documentation at Swagger Hub.
 - 400 Bad Request (error in the JSON not parsable)
 - 422 Unprocessable Entity (Not according to JSON semantical schema or the semantical model)
 - 403 Forbidden (unknown trading party or partner agreement). Also used to respond that the indicated RequestType (full/simple check) is not supported
 - 404 Not Found (incorrect URL)
 - 500 Internal Server Error
 - 503 Service Unavailable

- b) The second type is the messages that are entered from the seller's system. This message consists of a *Response code* (data element *LineStatusCode*) and, as needed, a *Response clarification* (data element *Note*). The codes are based on the Peppol BIS Order Response 3 and UNCL1229, as described in the section 5.3.2. Additionally, the supplier may propose an substitute item if the one requested is out of stock.

6.3.5 Definitions

Not applicable for this specification.

6.3.6 Security

6.3.6.1 Authenticity

Authentication method is described in SFTI API Authentication specification.

6.3.6.2 Confidentiality

This API requires the use TLS/SSL.

6.3.7 Tags

Not applicable for this specification.

6.3.8 External Documentation

No additional external documentation.

6.4 Technology Bindings

This section of the specification provides additional technology specific details that enable seamless interoperability between implementors of this specification.

6.4.1 Syntax bindings

6.4.1.1 *JSON*

The data for the messages in this specification are transferred using JSON syntax according to the Open API principles and Technical API details specified in Swagger Hub.

7 Release Log

This section provides documentation of changes to this document and underlying standards and other specifications.

Date	Version	Changes
2020-08-25	1.0	First version
2021-06-21	1.1	Changed authentication method from basic auth to OAuth. New method described in separate specification. Clarification that a API provider can support Simple and/or Full. Instructions added for how to respond when a non-supported request type is used.