



Spécifications techniques: Définitions de messages des services BCSS

Historique des révisions

Date	Version	Description	Author
29/11/2011	0.1	Version initiale pour validation interne	PVDB
05/12/2011	0.2	Adaptations après validation interne	PVDB
14/12/2011	1.0	Version pour envoi aux partenaires	PVDB
09/02/2012	1.1	- Le numéro d'ordre (messages en mode batch) a été ajouté aux données standard, mise à jour des exemples, adaptation « informationNotified » dans le CommonV3.xsd; - précisions des tickets, codes de statut et d'erreur; - ajout de la représentation des dates et des timestamps; - suppression des critères de bloc comme données standard; - ajout d'un exemple d'utilisation du contexte légal	PVDB
12/10/2012	1.2	Explication du choix de langue dans la description des codes statut et codes erreur.	PVDB
20/01/2013	1.3	Adaptation pour les vouchers : Vouchers, les messages Batch in volgorde ophalen (to translate)	TWU
14/08/2013	1.4	Clarification : fileSequenceNumber in voucher section 3.3	TWU
14/10/2014	1.5	Ajouté SenderReceiverType	TWU
27/11/2014	1.6	Changements restants pour SenderReceiverType	BST
22/12/2014	1.7	IntegrationContextType -> InscriptionContextType	
22/01/2015	1.8	Modification de SequenceNumberType	BST
10/02/2015	1.9	Modification de TicketType	PME
02/10/2015	1.10	Précisions complémentaires sur le contexte légal	JDM
15/10/2015	1.11	Ajout point de contact central pour les problèmes	JDM
15/02/2016	1.12	Ajout codes d'erreur génériques	JDM
19/10/2016	1.13	Mise à jour fichier batch	MBU
30/03/2018	1.14	Explication du filtrage de données (« dataFilters »)	SBO
12/06/2018	1.15	Liste des codes techniques	MDB
18/12/2018	1.16	Modification listes des codes business et techniques	JDM
22/02/2019	1.17	Ajout code technique MSG00016 et modification pour MSG00014	BST
15/03/2019	1.18	Modification description pour MSG00012, MSG00021	WDE
17/04/2020	1.19	Ajout de codes business MSG00017 et MSG00018	JDM



Documents y afférents

Document	Author
[1] CommonV3.xsd CBSSCommonExample.xsd	BCSS
[2] Documentation générale relative aux services web BCSS (plate-forme SOA) Documentation relative à l'architecture orientée service	BCSS
[3] Documentation générale relative aux fichiers batch/lot de messages LDM (plate-forme SOA) Projet "Lot de messages" (LDM)	BCSS
[4] Spécifications techniques du projet BatchSOAP TSS BatchSOAP	BCSS
[5] Liste des actions pour accéder à l'infrastructure SOA de la BCSS https://ksz.fgov.be/sites/default/files/assets/services_et_support/11soa_accesinfrastructurebcss.docx	KSZ



Table des matières

Spécifications techniques: Définitions de messages des services BCSS	1
Historique des révisions	1
Documents y afférents	2
Table des matières	3
1 Portée	4
2 Services web	5
2.1 Format	5
2.2 Idempotence	5
2.3 Sécurité	5
2.4 Exemples	5
2.4.1 Consultation sur base du NISS	5
2.4.2 Consultation sur base du NISS et de la période	6
3 Fichiers batch/lot de messages	7
3.1 Format	7
3.2 Voucher	8
3.3 Message batch recherché en séquence	8
3.4 BatchSOAP: Faire appel à des services web via le fichier batch	9
3.5 Exemples	10
3.5.1 Service type de requête/réponse	10
3.5.2 Service type de notification ou de mutations	10
3.5.3 Fichier BatchSOAP	11
4 Données standard	12
4.1 Legal context (contexte légal)	12
4.2 Filtres de données (dataFilters)	12
4.3 Inscriptions	12
4.4 Exemple de détermination du legal context et du inscription context	13
4.5 InformationCustomer and informationCBSS pour les services online	14
4.6 Identification de l'institution pour les services online	15
4.7 Identification d'une institution pour des services batch	15
4.8 Ticket de message et UUID	17
4.9 Des services online auprès desquels la BCSS est cliente	17
4.10 NISS	18
4.11 Critères de recherche	18
4.12 Statut de la réponse	19
4.13 Des messages d'erreur techniques	21
4.14 Messages en batch ou lot de messages	25
4.15 Numéro d'ordre	26
4.16 Dates	26
4.17 Timestamps	27
4.18 Résumé des données standard	27
4.19 Comparaison d'un header A1 avec les données standard XML	29
5 Routage par les institutions de gestion: FAQ	30
6 Point de contact central pour les problèmes	32



1 Portée

Ce document présente la méthodologie lors de la définition de nouveaux messages en format XML telle qu'elle est suivie par la BCSS, tant pour les services web que pour les services batch.

Ce document présente:

- ✓ le format utilisé pour les services web et les fichiers batch, y compris quelques exemples de définitions de message
- ✓ les données standard présentes au sein de chaque service
- ✓ les données standard souvent présentes
- ✓ le routage des données par une institution de gestion dans un réseau secondaire

Des exemples de définitions de message sont aussi disponibles dans le fichier CBSSCommonExample.xsd ([1]).

La méthodologie au sein des services web SSDN et au sein des fichiers batch en XMLD n'est plus suivie lors du développement de nouveaux services. Elle n'est donc pas décrite dans ce document.

La BCSS suit une architecture orientée services lors du développement de nouveaux services. Cette méthodologie ne met pas l'accent sur le développement de formulaires avec un header A1.

Formulaire: Demander des données auprès d'un fournisseur relatives à un seul dossier, donc à une phase code qualité, à la période du dossier et à celle du message.

La procédure suivie par la BCSS consiste à vérifier l'existence d'un dossier requis et à éventuellement passer à un filtrage des données, selon les contrôles imposés par le Comité sectoriel.

Service: Afin de répondre aux besoins spécifiques d'un client dans le cadre de sa mission au sein de la sécurité sociale. La réalisation de cet objectif par la BCSS consiste à fournir au client des informations ciblées provenant du réseau de la sécurité sociale, mais dans une structure indépendante du protocole ou de la façon dont le fournisseur met les données à disposition.

La procédure de la BCSS consiste à agréger des informations provenant des sources disponibles et à les transformer dans un format permettant un progrès auprès des partenaires. La BCSS veille à offrir un service stable, même si des changements se produisent auprès des sources des données.



2 Services web

2.1 Format

Les messages d'un service web sont définis à l'aide d'un WSDL et des XSD y afférents. Un WSDL regroupe plusieurs opérations d'un service ayant chacune une définition de la requête et de la réponse. Un SOAP-Fault standard est utilisé pour des erreurs techniques contenant, si d'application, des informations d'erreur détaillées spécifiques au service (cfr. [2]).

2.2 Idempotence

Les services web sont développés afin de présenter un comportement idempotent. C'est-à-dire que l'exécution multiple d'une requête aboutit au même résultat que l'exécution unique d'une requête.

Cette méthodologie évite des problèmes lors d'une nouvelle exécution d'une requête après une erreur. Lors d'un message d'erreur ou d'un timeout, l'utilisateur n'est, en effet, pas en mesure de savoir dans quelle mesure la requête a été exécutée.

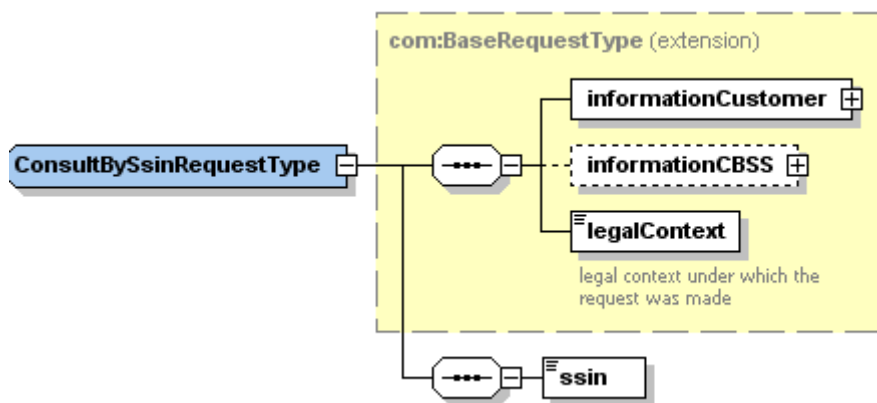
2.3 Sécurité

La protection des services web est prévue de manière standard à l'aide d'une connexion https 2-way SSL et de la possibilité de cryptage à l'aide de WS-Security. Pour plus d'informations, nous vous renvoyons au [2].

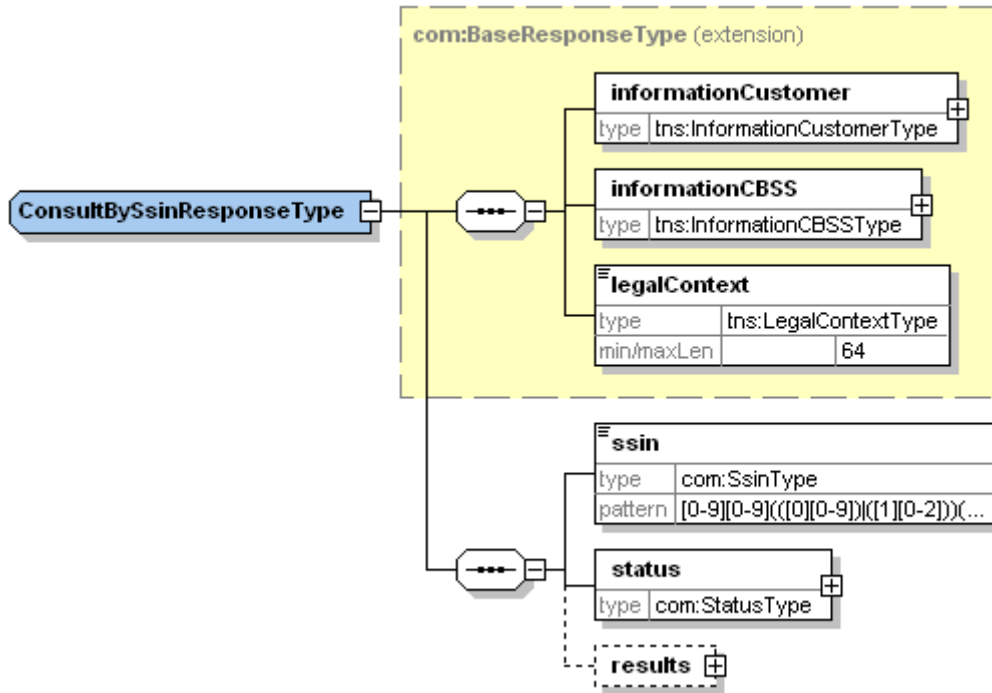
2.4 Exemples

2.4.1 Consultation sur base du NISS

Requête

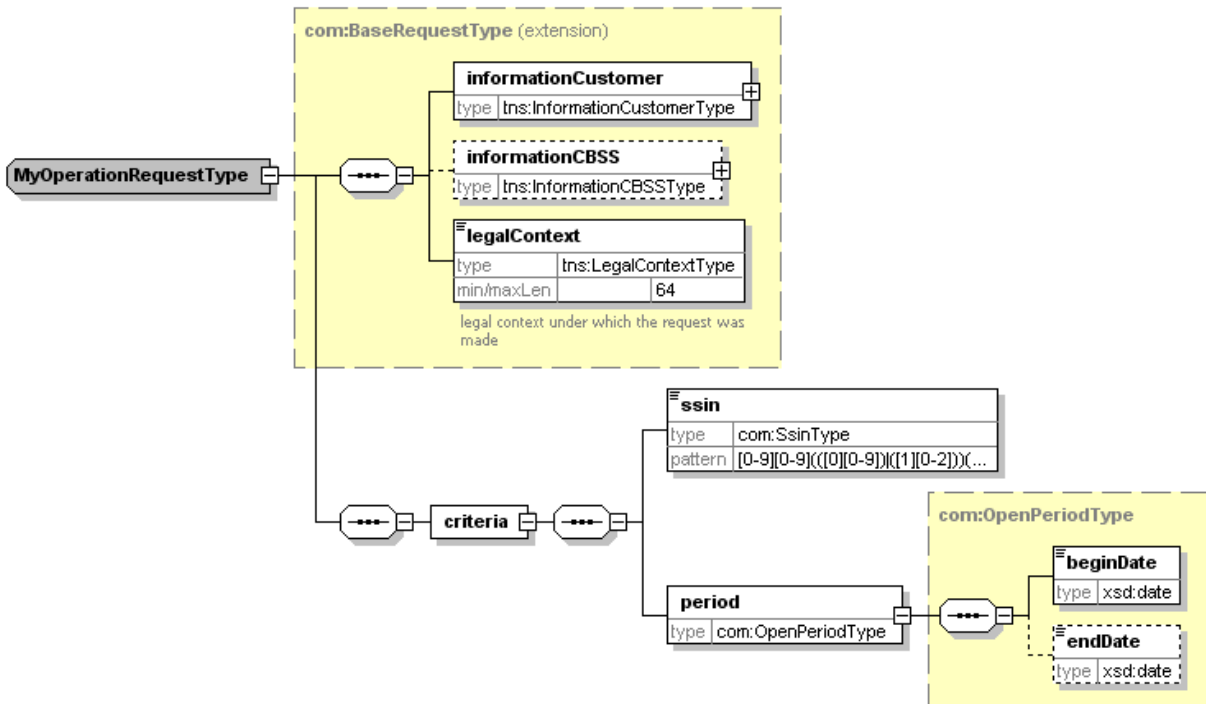


Réponse



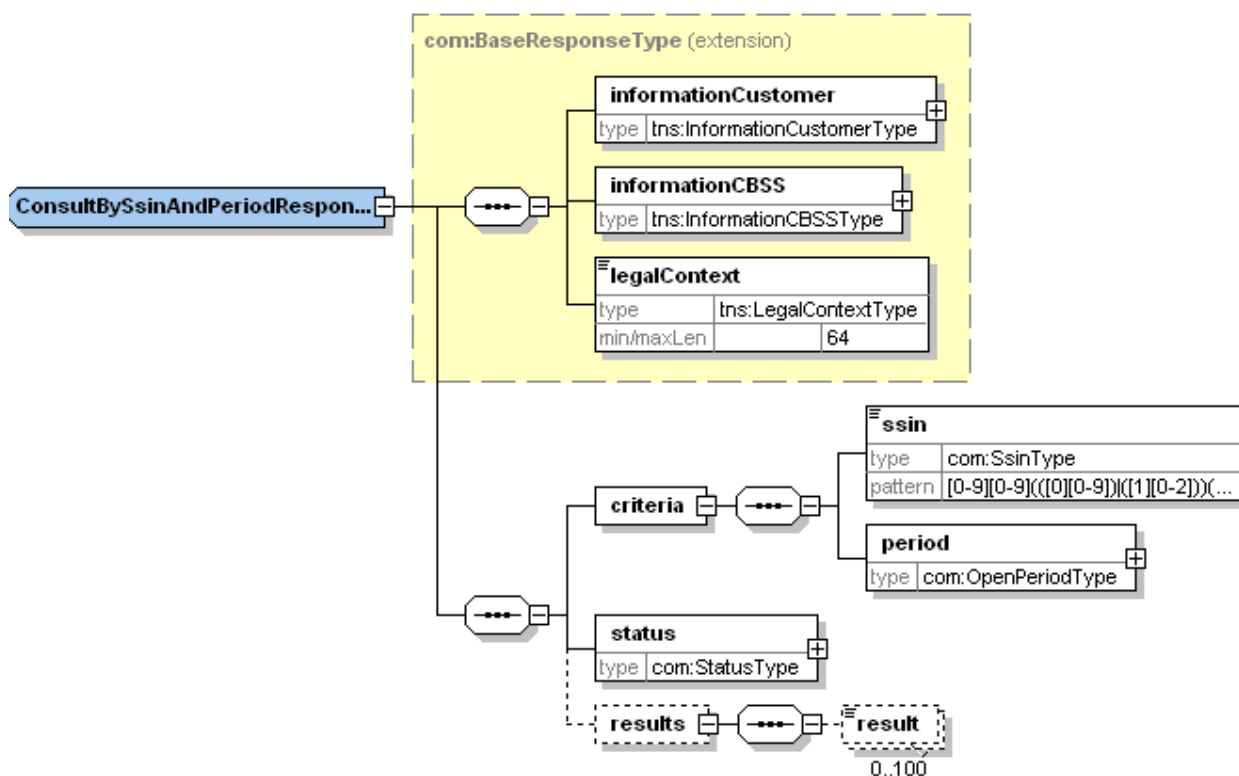
2.4.2 Consultation sur base du NISS et de la période

Requête





Réponse



3 Fichiers batch/lot de messages

3.1 Format

Les définitions de message pour des fichiers ressemblent fortement à celles pour des services web. Un fichier est considéré comme un message répondant à un XSD spécifique à un service.

Par exemple: les mutations du Registre national et du Registre BCSS d'une journée constituent un message répondant à un XSD. Ce message contient l'identification de l'expéditeur et du destinataire, un contexte légal et une liste de références.

Différences avec les services web:

- ✓ Pas de WSDL, seul un ou plusieurs XSD définissant les messages
- ✓ Il n'y a pas forcément un message de réponse
- ✓ Aucune enveloppe SOAP, mais bien un fichier voucher (cfr [3])
- ✓ Souvent des messages contenant un nombre élevé d'occurrences d'un ou plusieurs blocs de données (= lot de message)
- ✓ Généralement, un délai de traitement élevé

Trois types de services sont distingués:

1. Requête/Réponse
2. Notification (one way push): informer une organisation lors d'un évènement



3. BatchSOAP: Ceci est un service batch spécifique offert par la BCSS qui permet de faire appel à des services web en mettant une liste de requêtes service web dans un fichier batch.

Au sein des services du type requête/réponse et du type BatchSOAP, le lien entre la demande et la réponse peut être établi à l'aide du ticket dans les blocs *sender* et *receiver*. Les données du bloc *sender*, telles que figurant dans la demande, sont reprises dans le bloc *receiver* de la réponse.

Un service batch du type requête/réponse est préféré à l'utilisation de BatchSOAP, lorsque le service est exclusivement utilisé avec un nombre élevé de données. Le service BatchSOAP est préféré, lorsqu'un service web répondant aux besoins existe déjà.

3.2 Voucher

Les fichiers batch sont accompagnés d'un fichier voucher. Le fichier voucher est un fichier qui contient les méta données des fichiers batch liés à ce voucher. Il indique le service et les opérations pour lesquels le fichier est destiné. Ce fichier est également utilisé pour vérifier l'intégrité des fichiers de données, et peut contenir des statistiques sur les fichiers de données. Voir [3] pour plus d'informations.

3.3 Message batch recherché en séquence

Pour certains services, comme les changements de données personnelles, il peut être nécessaire que les messages soient traités dans un ordre séquentiel. Le traitement de l'ordre des fichiers de données est décrit dans la section 4.13.

Afin de faciliter les traitements, les messages batch doivent être recherchés à l'aide du voucher.

Pour cela, les principes suivants sont utilisés:

Détecter les vouchers manquants et les fichiers de données correspondants

Les institutions qui souhaitent pouvoir détecter les vouchers manquants et les fichiers de données correspondants, ont intérêt à utiliser l'élément « uniqueIdentifier », en s'assurant qu'il comporte des valeurs incrémentales.

Dans l'exemple ci-dessous, on ne peut donc commencer à récupérer les fichiers de données X et Y que si toutes les données du fichier correspondant au voucher avec la valeur 000000000010764 dans l'élément « uniqueidentifier » sont récupérées.

Si une institution (BCSS, fournisseur, client) détecte un voucher manquant, cette institution doit contacter l'institution qui a envoyé le voucher. Le service desk de la BCSS est joignable par mail à l'adresse servicedesk@ksz-bcss.fgov.be ou par téléphone au numéro 02-741 84 00 entre 08h et 16h30 les jours ouvrables.

Déterminer l'ordre des fichiers de données dans un voucher

L'ordre des fichiers de données dans un voucher est déterminé par l'élément « fileSequenceNumber ».

Dans l'exemple ci-dessous, on doit donc d'abord récupérer les données du fichier X et ensuite les données du fichier Y. Le fileSequenceNumber représente l'ordre des fichiers fournis dans le voucher, celui-ci commence à partir de 1.



Remarque: Le fileSequenceNumber est utilisé pour l'ordre des fichiers, et par conséquent il ne correspond pas avec le sequenceNumber qui peut être présent dans les fichiers de données.

Voici un exemple du contenu d'un fichier voucher pour XML notifications.

```
<?xml version="1.0" encoding="UTF-8" ?>
<tns:lotPackageVoucher
xmlns:tns="http://kszbcss.fgov.be/types/Batch/External/LotPackageVoucherObjects/20090716">
  <metaData>
    <voucherName> ... </voucherName>
    <version>1</version>
    <uniqueIdentifier>000000000010764</uniqueIdentifier>
    ...
  </metaData>
  <packagedLotFiles>
    <packagedLotFile>
      <lotFileName> X </lotFileName>
      <fileSequenceNumber>1</fileSequenceNumber>
      ...
    </packagedLotFile>
    <packagedLotFile>
      <lotFileName> Y </lotFileName>
      <fileSequenceNumber>2</fileSequenceNumber>
      ...
    </packagedLotFile>
  </packagedLotFiles>
</tns:lotPackageVoucher>
```

3.4 BatchSOAP: Faire appel à des services web via le fichier batch

BatchSOAP est un service batch spécifique offert par la BCSS qui permet de faire appel à des services web en intégrant une liste de requêtes service web dans un fichier batch et en les envoyant à la BCSS. La BCSS fournit en retour un ou plusieurs fichiers batch contenant les réponses des services web. Il est possible de combiner des requêtes destinées à plusieurs services web ou à plusieurs opérations dans un fichier.

Le fichier batch est uniquement considéré comme un moyen de transmission des requêtes. Les blocs sender et receiver déterminent les partenaires entre lesquels ont lieu les échanges. Chaque webservice possède aussi les blocs informationCustomer, informationCBSS qui correspondent aux sender et receiver.

Le service ne garantit pas que le traitement s'effectue dans un certain ordre, ni que toutes les réponses seront communiquées dans un fichier batch. Le lien entre le fichier requête et le fichier réponse peut être établi à l'aide du ticket que le client communique dans le bloc sender de chaque fichier.

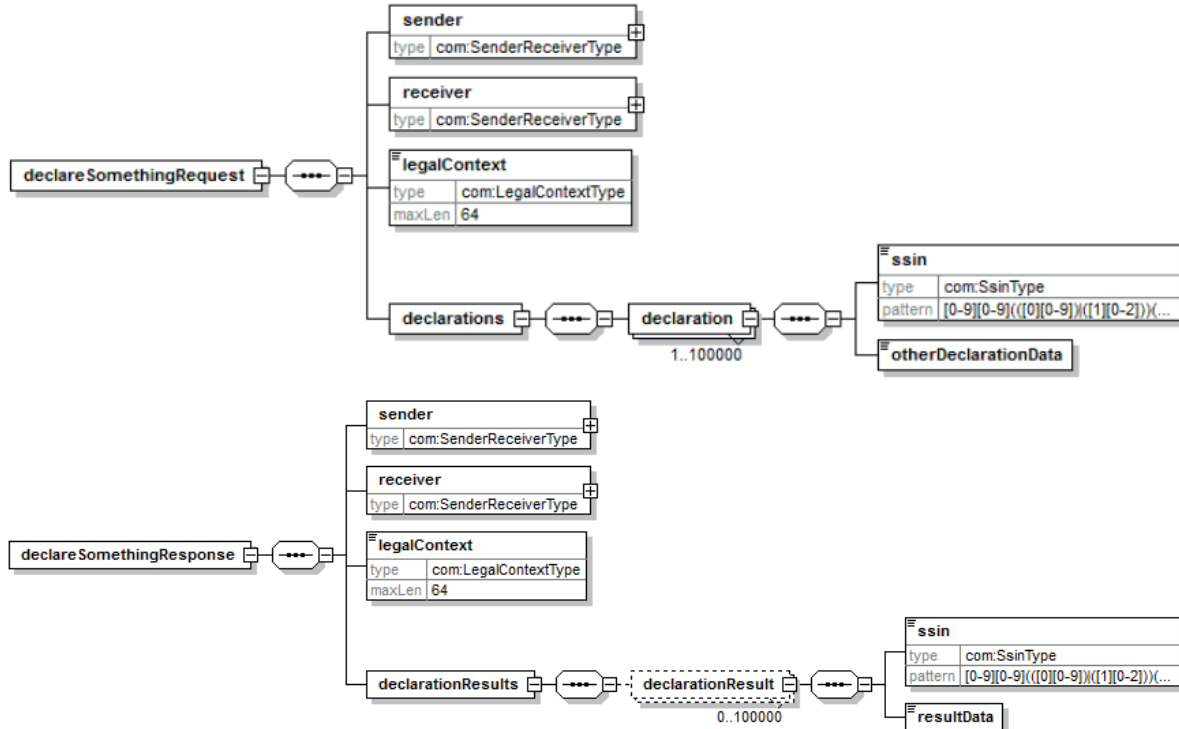
Le lien entre chaque requête et la réponse du webservice sera effectué à l'aide du ticket présent dans le bloc informationCustomer de la requête.



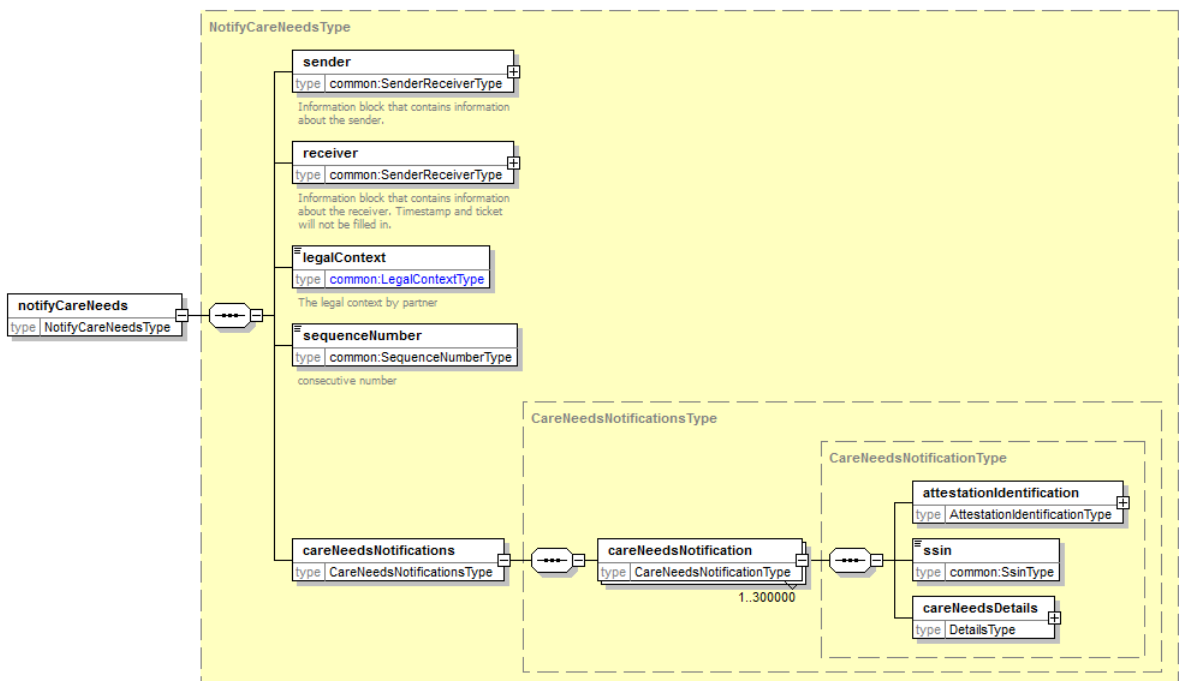
Pour les spécifications du service BatchSOAP, nous vous renvoyons à ce document [4].

3.5 Exemples

3.5.1 Service type de requête/réponse



3.5.2 Service type de notification ou de mutations





3.5.3 Fichier BatchSOAP

```

<?xml version="1.0" encoding="UTF-8"?>
<n1:batchSOAPRequest
  xmlns:n1="http://kszbcss.fgov.be/types/Batch/External/BatchSoapObjects/20141210/"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <sender>
    <ticket>EXAMPLE_TICKET</ticket>
    <timestampSent>2001-12-17T09:30:47.0Z</timestampSent>
    <organizationIdentification>
      <sector>11</sector>
      <institution>0</institution>
    </organizationIdentification>
  </sender>
  <receiver>
    <organizationIdentification>
      <sector>25</sector>
      <institution>0</institution>
    </organizationIdentification>
  </receiver>
  <batchSoapEntries>
    <soapenv:Envelope
      xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
      xmlns:wsa="http://www.w3.org/2005/08/addressing"
      xmlns:v1="http://kszbcss.fgov.be/intf/ConsultPensionRegisterService/v1">
      <soapenv:Header>
        <wsa:To>https://b2b.ksz-
bcss.fgov.be:4520/ConsultPensionRegisterService/ConsultPensionRegister</wsa:To>

        <wsa:Action>http://kszbcss.fgov.be/ConsultPensionRegisterService/getRightsAnd
MaximalPayments</wsa:Action>
      </soapenv:Header>
      <soapenv:Body>
        <v1:getRightsAndMaximalPaymentsRequest>
... request body...
        </v1:getRightsAndMaximalPaymentsRequest>
      </soapenv:Body>
    </soapenv:Envelope>
  </batchSoapEntries>
</n1:batchSOAPRequest>
  
```

4 Données standard

Une nouvelle définition de message est établie pour chaque nouveau service web ou service batch. Cette définition est établie en fonction des besoins spécifiques de ce service. Il est prévu de placer les données devant revenir dans chaque service dans une position standard dans la définition de message.

4.1 Legal context (*contexte légal*)

Le cadre légal (*legal context*) indique les finalités pour lesquelles un service de la BCSS peut être utilisé par un partenaire. L'indication renvoie donc à la finalité de l'accès aux données, telle que spécifiée dans l'autorisation accordée par le Comité sectoriel. Si l'autorisation reste vague sur la finalité, le contexte légal sera aussi vague (ex. ANY_USE).

En fonction du contexte légal indiqué, de l'opération invoquée et de l'organisation faisant appel au service, la BCSS peut imposer une série de conditions concernant l'utilisation du service. Ainsi, une condition peut consister dans l'existence d'une intégration pour la personne pour laquelle des données sont consultées.

Le champ *legal context* est un champ obligatoire dans chaque requête de service et est repris dans chaque réponse. Le champ est présenté au moyen d'une chaîne ayant une longueur maximale de 64 caractères. Les valeurs sont des représentations textuelles, généralement en anglais et en majuscules, et dans lesquelles les mots sont séparés par des tirets de soulignement. Celles-ci sont précédées par l'abréviation de l'organisation en anglais et par un double point. Chaque valeur est définie de manière bilatérale par la BCSS et l'organisation concernée au cours de la phase d'analyse et est finalement reprise dans les spécifications du service (TSS document).

4.2 Filtres de données (*dataFilters*)

Une institution de sécurité sociale pourrait obtenir un ensemble de données dont certaines peuvent être inutile ou non autorisée pour elle. Dans ce cas, un filtrage doit être effectué sur ces données. C'est pourquoi un champ « dataFilters » est rajouté afin d'avertir l'institution quels sont les filtres qui ont eu lieu sur les données renvoyées. Par conséquent, lorsqu'il y a un filtrage ou non, cette information sera toujours statique pour un client pour un service.

4.3 Inscriptions

Une institution de sécurité sociale peut enregistrer auprès de la BCSS qu'elle est en train de traiter un dossier concernant une certaine personne. Cet enregistrement est dénommé une intégration. La façon dont la personne a affaire à l'organisation est dénommée une qualité (*inscription context/qualité*).

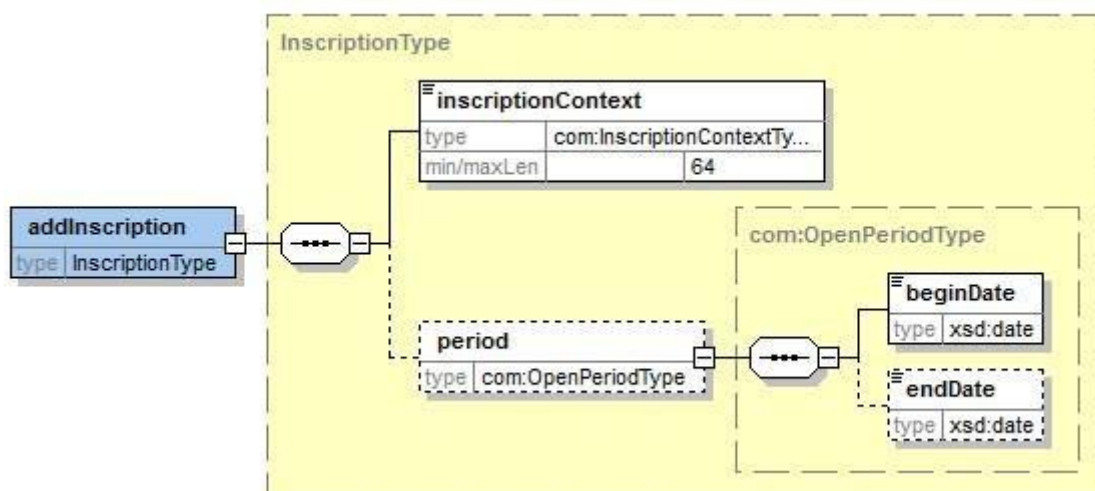
Une inscription est caractérisée par:

- ✓ une qualité
- ✓ une institution de sécurité sociale
- ✓ une personne
- ✓ la période dans laquelle la personne est associée à l'institution

Le champ *inscription context* est représenté au moyen d'une chaîne ayant une longueur maximale de 64 caractères. Les valeurs sont des représentations textuelles, généralement en anglais et en minuscules, et dans lesquelles les mots sont séparés par des tirets de soulignement. Celles-ci sont précédées par l'abréviation de l'organisation en anglais et par un double point. Chaque valeur est définie au préalable par la BCSS et par l'organisation concernée. L'utilisation d'une représentation textuelle au lieu d'une représentation numérique facilite l'interprétation de ce champ.

Notez que le contexte d'intégration remplace l'ancienne combinaison composée d'un code qualité et d'un code phase (le code phase étant souvent égal à zéro). Si le code phase n'est pas utilisé, un mapping 1 à 1 existe entre le contexte inscription et le code qualité.

En fonction des besoins pour un service, un service offre parfois l'enregistrement d'une intégration, ainsi que la fonctionnalité primaire du service. Dans ce cas, un bloc d'enregistrement est prévu dans la requête du service.



Notez que le champ *inscriptionContext* n'est pas présent de manière standard dans les définitions de message de chaque service, contrairement au champ *legalContext*.

4.4 Exemple de détermination du *legal context* et du *inscription context*

La définition d'un *legal context* correspond à un cadre légal. Il ne décrit ni qui est consulté (il s'agit de la qualité, p.ex. pour le personnel, l'*inscription context* *personnel* existe), ni ce qui est consulté (il s'agit du nom du service, p.ex. pour la consultation du cadastre des pensions, le service « *ConsultPensionRegister* » existe). Il décrit plutôt pourquoi une consultation est réalisée (finalité).

L'exemple suivant sert à clarifier la signification du champ *legal context* et à expliquer comment celui-ci peut être relié au contrôle de l'existence d'inscriptions.

Les CPAS demandent des données à caractère personnel auprès de la BCSS afin d'examiner à quels avantages sociaux une personne a droit et dans le cadre de l'administration du personnel



d'un CPAS. Un CPAS peut uniquement examiner les avantages sociaux des personnes pour lesquelles le CPAS a un dossier 'en cours d'examen' et des personnes qui perçoivent déjà un revenu d'intégration social du CPAS.

L'autorisation est plus stricte pour la finalité administration du personnel que pour l'examen d'avantages sociaux. Moins de données peuvent être consultées et, par conséquent, moins de services peuvent être utilisés. Ainsi, la composition du ménage ne peut pas être demandée pour un membre du personnel.

Le secteur des CPAS et la BCSS définissent en concertation deux contextes légaux:

- ✓ PCSA : PERSONNEL_ADMINISTRATION
- ✓ PCSA : BENEFITS_INVESTIGATION (PCSA = public center of social aid = CPAS)

Sont aussi définis plusieurs contextes d'inscription à l'aide desquels un CPAS peut enregistrer des inscriptions auprès de la BCSS.

- ✓ pcsa:personnel (collaborateur d'un CPAS)
- ✓ pcsa:under_investigation (personne avec un dossier 'en cours d'examen' auprès d'un CPAS)
- ✓ pcsa:living_wage_beneficiary (personne qui perçoit un revenu d'intégration social d'un CPAS)

Lorsqu'un CPAS fait appel à un service pour la consultation de la composition du ménage ayant le contexte légal PERSONNEL_ADMINISTRATION, la consultation sera toujours refusée par la BCSS. Lors de l'utilisation du contexte légal PCSA_BENEFITS_INVESTIGATION, la BCSS donnera accès lorsqu'elle connaît une inscription ayant un des contextes d'inscription pcsa:under_investigation ou pcsa:living_wage_beneficiary pour cette personne.

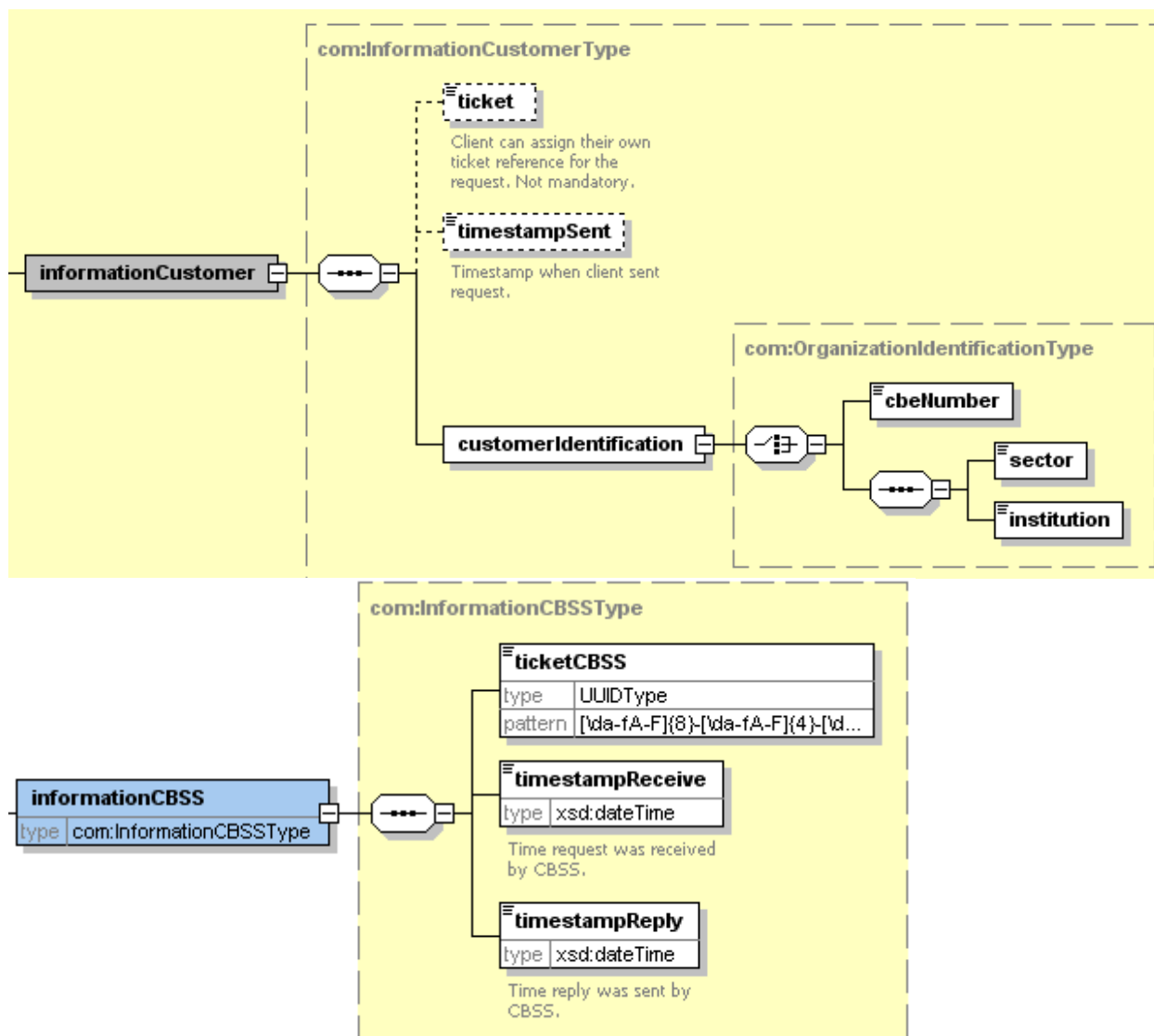
Notez que les contextes d'inscription ne sont pas repris dans les définitions de message du service pour la consultation de la composition du ménage, contrairement à un contexte légal.

4.5 InformationCustomer and informationCBSS pour les services online

Requête:

- ✓ informationCustomer : obligatoire;
- ✓ informationCBSS : facultatif, mais ne sera pas utilisé dans la requête. Ce bloc est présent pour des raisons techniques en interne (BCSS).

Réponse: informationCustomer et informationCBSS obligatoires



4.6 Identification de l'institution pour les services online

L'institution est définie dans un message:

- ✓ soit à l'aide de la combinaison secteur/institution pour les institutions au sein de la sécurité sociale.
- ✓ soit à l'aide du numéro BCE

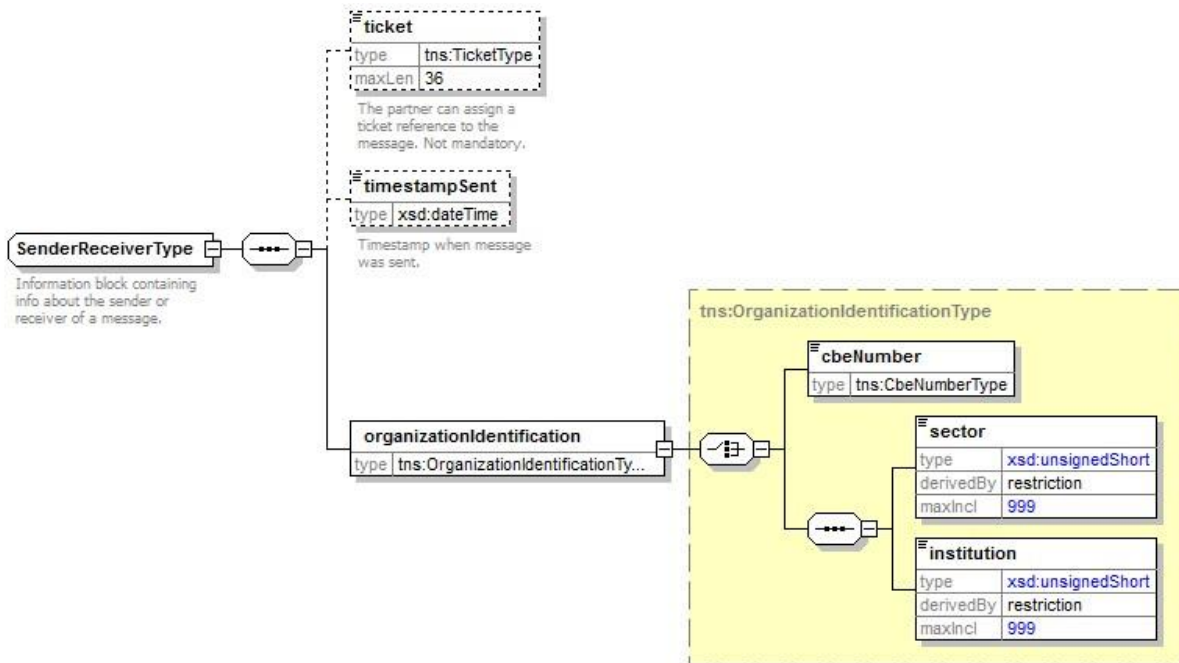
Le type XSD réutilisable `OrganizationIdentificationType` est défini en [1]. Ce type est utilisé dans les blocs `informationCustomer` et `informationSupplier`.

4.7 Identification d'une institution pour des services batch

Les données de l'expéditeur ou du destinataire se trouvent dans le bloc `sender` ou `receiver` pour la demande et/ou la réponse des messages batch. L'expéditeur ou le destinataire se trouve dans la requête et/ou la réponse des messages batch avec le même type : `SenderReceiverType`. Ce type contient le type XSD réutilisable



OrganizationIdentificationType (comme ci-dessus dans le point 4.5 pour les services online) pour identifier les institutions.



Message du client

Request:

- sender
 - ticket: optionnel
 - timestampSent : optionnel
 - organizationIdentification: le client doit être identifié par un numéro CBE ou le secteur et l'institution.
- receiver
 - ticket: absent
 - timestampSent : absent
 - organizationIdentification: identification de la BCSS

Reply:

- sender
 - ticket: optionnel
 - timestampSent: optionnel
 - organizationIdentification: le client doit être identifié par un numéro CBE ou le secteur et l'institution.
- receiver
 - ticket est copié du bloc sender de la requête
 - timestampSent est copié du bloc sender de la requête (donc c'est la date et l'heure de la création de la requête par la BCSS)
 - organizationIdentification: identification de la BCSS

Message de la BCSS

Request:



- sender
 - ticket: rempli
 - timestampSent: rempli
 - organizationIdentification: identification de la BCSS
- receiver
 - ticket: absent
 - timestampSent : absent
 - organizationIdentification: le client doit être identifié par un numéro CBE ou le secteur et l'institution.

Reply:

- sender
 - ticket: rempli
 - timestampSent: rempli
 - organizationIdentification: identification de la BCSS
- receiver
 - ticket est copié du bloc sender de la requête
 - timestampSent est copié du bloc sender de la requête (donc c'est la date et l'heure de la création de la requête par le client)
 - organizationIdentification: le client doit être identifié par un numéro CBE ou le secteur et l'institution.

Pour plus d'information sur le ticket, voir point 4.7, le tag timestampSent est le timestamp correspondant à la création du message par l'expéditeur.

4.8 Ticket de message et UUID

Dans les web service, la BCSS utilise toujours des tickets de message selon le standard UUID (Universal Unique Identifier¹) afin d'identifier les messages créés par la BCSS de manière unique. Un ticket unique est attribué à cet effet à chaque nouveau message. Lorsqu'un message est envoyé plusieurs fois suite à une erreur technique, le même ticket *peut* être réutilisé.

Il est conseillé à chaque partenaire de communiquer le ticket de message créé par la BCSS lors de l'examen d'un problème. Ceci permet à la BCSS de consulter les logs rapidement afin de résoudre le problème au plus vite.

Les partenaires de la BCSS peuvent aussi utiliser des UUID comme un ticket de message, mais ceci n'est pas obligatoire. Outre une UUID les partenaires peuvent aussi utiliser une chaîne avec une longueur maximale de 32 caractères pour le champ de ticket.

Dans les messages batch la BCSS utilise des tickets de 15 caractères. Ce ticket commence par un caractère qui indique l'environnement ('T', 'A', 'P'). Ce premier caractère est suivi par un file id numérique (précédé de zéros) de longueur 14. Exemple: 'A00000123893995'.

4.9 Des services online auprès desquels la BCSS est cliente

Lors du développement d'un nouveau service auprès duquel la BCSS est cliente, la BCSS dispose parfois de la possibilité de définir le WSDL et les XSD. Ceci augmente la cohérence

¹ Voir aussi <http://en.wikipedia.org/wiki/Uuid>



entre les définitions de message et peut ainsi influencer favorablement le délai de développement.

Dans ce cas, les données standard suivantes sont utilisées:

Requête: informationCustomer et informationSupplier obligatoires

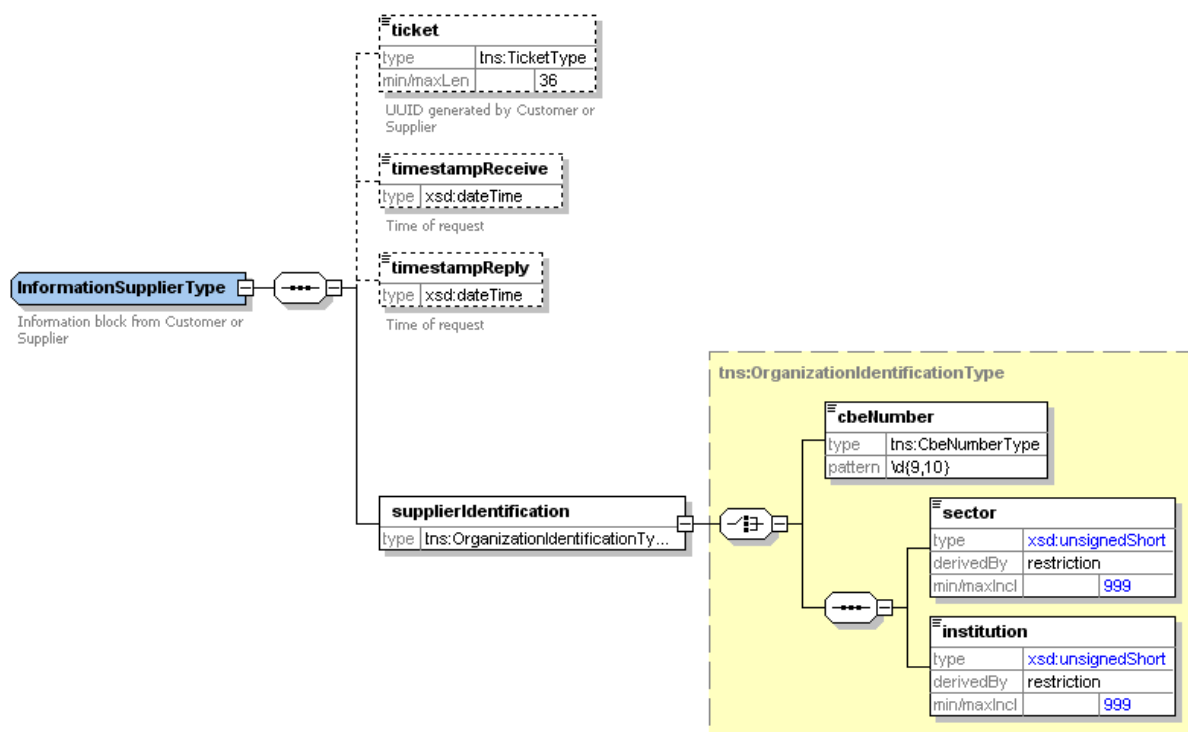
→ informationCustomer est complété par l'identification de la BCSS (numéro CBE 0244640631), par le ticket UUID et par la requête timestamp

→ informationSupplier: est uniquement complété par l'identification de l'organisation offrant le service

Réponse: informationCustomer et informationSupplier obligatoires

→ informationCustomer: repris de la requête

→ informationSupplier: en plus de l'identification, une réponse de ticket et de timestamp est de préférence complétée par le fournisseur



4.10 NISS

La présence d'un champ pour un NISS dépend des besoins du service. Lorsqu'un service traite un seul NISS, un élément dénommé 'ssin' est présent. Le champ n'est pas présent dans une définition de message pour un service effectuant une recherche sur base du numéro BCE.

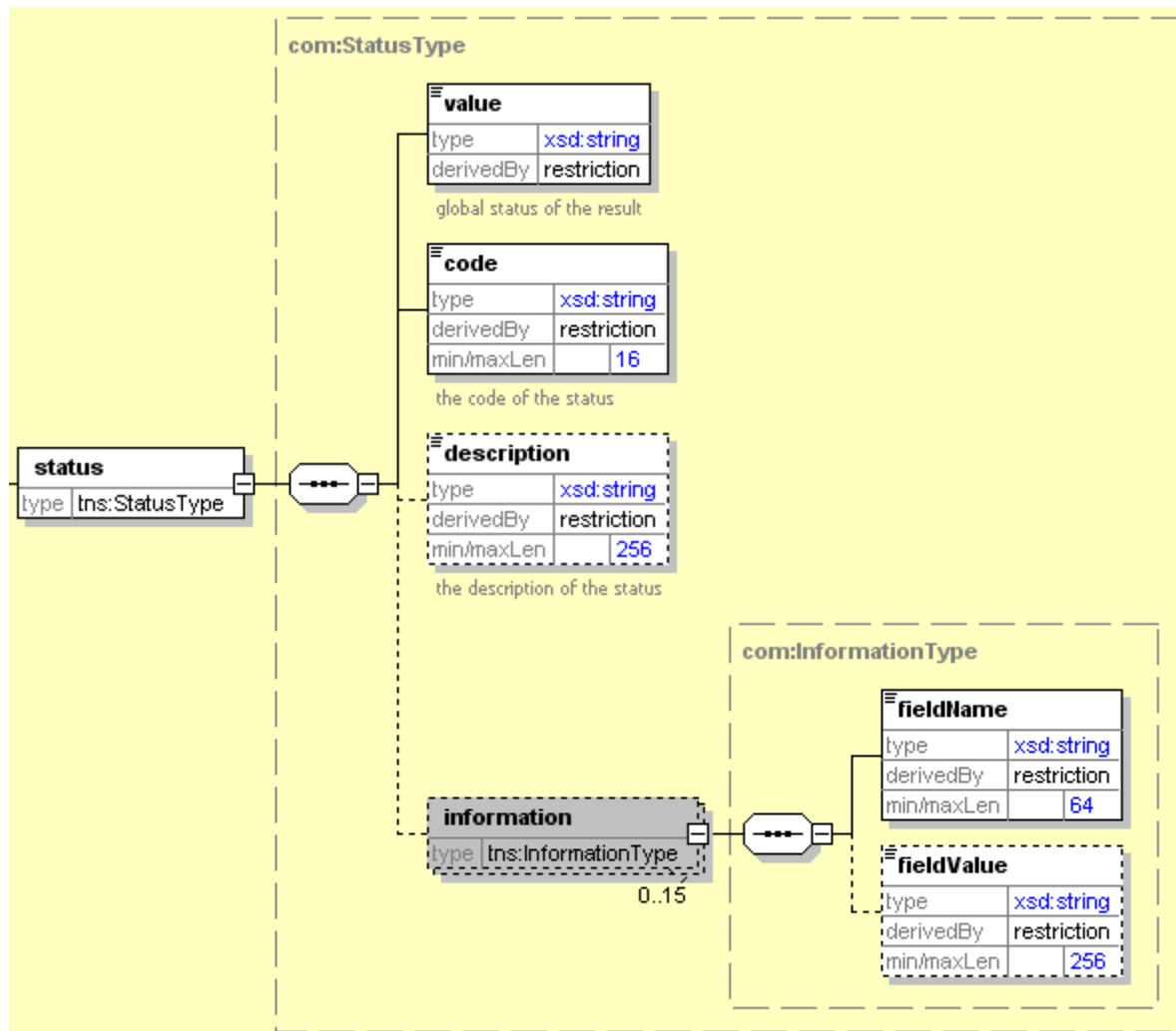
L'élément ssin est aussi repris dans la réponse.

Un type 'SsinType' est prévu dans un XSD commun [1].

4.11 Critères de recherche

Les critères de recherche se trouvent généralement dans un bloc d'une requête, par exemple sous 2.4.2. dans le bloc *criteria*. Les critères de recherche sont toujours repris au début de chaque réponse.

4.12 Statut de la réponse



Chaque réponse définit un bloc *statut* qui représente le statut global de la réponse. Il présente la structure suivante:

- *value* est une énumération de valeurs spécifiques à un service. Donne une indication générale sur le statut de la réponse. Le StatusType dans CommonV3.xsd définit les valeurs d'énumération courantes (DATA_FOUND, NO_DATA_FOUND, NO_RESULT)
- *code* donne plus de détails que *value*. Une liste des codes généraux courants est établie ci-dessous. Une liste des codes de statut possibles est reprise dans les spécifications (document TSS) de chaque service.
- *description* est un champ de texte libre qui donne plus d'explication, par exemple: "The requested ssin does not exist."
- *information* les paires *fieldName/fieldValue* peuvent être ajoutées comme des informations supplémentaires à certains messages afin de communiquer des données d'une manière structurée. Pour les use cases requérant des structures de données plus complexes, des blocs spécifiques, en plus du bloc statut, sont ajoutés à la réponse.



Les énumérations possibles pour le champ *value* dépendent des besoins d'un service. Un type *StatusType* est repris dans les types de données communs de CommonV3.xsd. Ce type convient pour les services communs du type consultation. Dans les services, où les valeurs possibles pour le champ *value* de ce *StatusType* ne conviennent pas, un autre *StatusType* est défini dans un XSD spécifique à ce service. A l'exception des valeurs d'énumération possibles, la même structure est maintenue à travers de l'ensemble des définitions de message. Les valeurs possibles des champs *value* et *code* sont documentées avec leur signification pour chaque service.

Ces codes sont établis et gérés par la BCSS. Les codes fréquemment utilisés qui sont réutilisés dans plusieurs services commencent par 'MSG', tandis que les codes spécifiques à un service commencent par d'autres caractères (p.ex. dans le service *LivingWages* le code LVWG0001 est utilisé).

Codes statut généraux, fréquemment utilisés :

Code (champ code)	Description (EN)	Valeurs possibles pour value	Description (FR)
MSG00000	Treatment successful	DATA_FOUND / OK	Traitement réussi
MSG00100	Treatment successful, but no data found at the supplier	NO_DATA_FOUND	Traitement réussi mais pas de données trouvé
MSG00005	The SSIN given in request does not exist	NO_RESULT / NOK	Le NISS est inconnu
MSG00006	The SSIN given in request has been replaced	NO_RESULT / NOK	Le NISS est remplacé
MSG00007	The SSIN given in request is cancelled	NO_RESULT / NOK	Le NISS est annulé
MSG00008	The request contains invalid data. Please check your message content.	NO_RESULT / NOK	Requête invalide
MSG00009	Person deceased	NO_RESULT / NOK	Personne décédée
MSG00010	The person is no longer inscribed in the National Register	NO_RESULT / NOK	Personne n'est plus inscrite dans le Registre National (le NISS est radié)
MSG00011	The structure of the SSIN given in request is invalid	NO_RESULT / NOK	Le NISS a un format invalide (checksum)
MSG00012	The SSIN is not sufficiently integrated for your organization	NO_RESULT / NOK	Le NISS est inconnu pour le demandeur dans ce contexte légal
MSG00013	Access to this operation is not allowed with the given legal context	NO_RESULT / NOK	Accès à l'opération refusé avec le contexte légal donnée
MSG00017	The person is inscribed in the BIS Register	NO_RESULT / NOK	Personne est inscrite dans le registre Bis
MSG00018	The person is inscribed in the National Register	NO_RESULT / NOK	Personne est inscrite dans le Registre National
MSG00021	The SSIN is not sufficiently integrated for the provider of the data	NO_DATA_FOUND	Le NISS est inconnu pour le destinataire dans ce contexte légal
MSG00026	The cbe number given in request is invalid	NO_RESULT / NOK	Numéro CBE invalide
MSG00027	The client is not authorized to consult the	NO_RESULT / NOK	Pas d'autorisation pour la donnée demandé



	requested data		
--	----------------	--	--

La BCSS a fait le choix de fournir la description du code statut dans une seule langue, à savoir l'anglais. Cette description figure dans le bloc « statut » du champ *description*. La traduction n'est pas fournie dans le message, étant donné que les descriptions sont principalement destinées à faciliter la détection d'erreurs dans le logging. Cette méthode de travail est d'ailleurs plus performante, ce qui permet d'améliorer le temps de réponse de la BCSS.

Il appartient à la logique de programmation des partenaires d'interpréter les codes retour et, sur cette base, de poursuivre ou d'interrompre la logique métier. Cette méthode de travail s'inscrit dans le cadre de la vision de la BCSS en matière d'offre de services web au lieu d'applications web. Ce choix est aussi conforme à l'ancienne plate-forme SSDN (qui ne prévoyait également qu'une description en une seule langue) et va au-delà des flux A1 où aucune description n'est prévue.

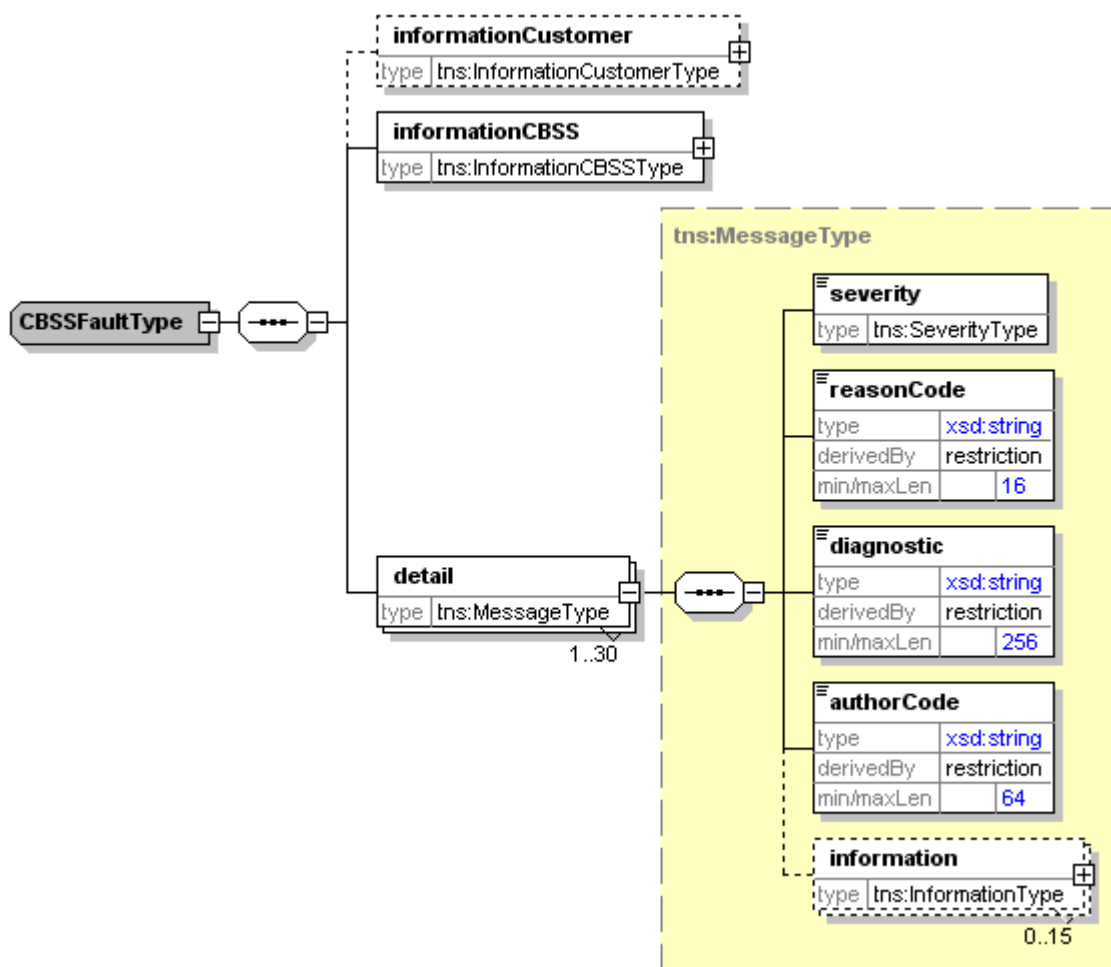
Conjointement avec les services dans le cadre d'un projet, la BCSS s'engage à fournir un document technique ("Technical Service Specifications") comprenant les codes possibles pour ces services avec une description pour chaque code.

A terme, tous les codes retour et leurs descriptions seront ajoutés dans l'application centrale des listes de codes existante (CTMS) et un ou plusieurs services web supplémentaires seront prévus pour la consultation de CTMS.

4.13 Des messages d'erreur techniques

Lorsqu'une erreur technique se produit lors de l'affichage d'un service web, comme en cas d'indisponibilité d'un service, d'une structure invalide vis-à-vis WSDL/XSD, la BCSS répond toujours par un soap-fault. Dans tous les autres cas, dans lesquels le service fonctionne sans problèmes et le schéma de la requête est valide, une réponse soap est donnée. Les erreurs de validation qui ne sont pas contrôlées par le WSDL ou le XSD sont aussi reprises dans la réponse, par exemple un checksum NISS invalide, la date de fin antérieure à la date de début, etc.

La BCSS envoie un bloc détail dans le soap-fault si le schéma de la requête est valide et le service a été contacté, mais qu'une autre erreur s'est produite (p.ex. ressource indisponible, erreur interne inattendue). Le bloc détail est défini dans le WSDL et fait appel à une structure fixe qui est définie dans CommonV3.xsd comme le *CBSSFaultType*. Le nom de l'élément ayant ce type de données correspond, conformément à la convention, à *<nom d'opération>Fault* ou *<nom de service>Fault*, tel que spécifié dans [2].



Les codes d’erreur techniques peuvent provenir tant de la BCSS que d’une autre institution. La source du code est indiquée au moyen du champ *authorCode*. Pour la BCSS, la valeur de ce champ est le suivant <http://www.ksz-bcss.fgov.be>. Les codes de la BCSS suivent la même structure que les codes statut (les codes généraux commencent par ‘MSG’), mais n’ont pas les mêmes valeurs. Un code erreur technique ne peut donc pas être affiché comme code statut ou vice versa.

Tout comme pour les codes statut, la BCSS opte pour une description des codes erreur en une seule langue, à savoir l’anglais. Cette description figure dans le champ *diagnostic*. À terme les codes erreur établis par la BCSS seront aussi mis à disposition via CTMS. Les codes possibles seront aussi insérés dans la documentation de chaque service.

La liste des codes d’erreur techniques généraux et fréquemment utilisés sont les suivants :

Code (champ <i>reasoncode</i>)	Signification	Actions
MSG00002	Problème de communication avec nos partenaires/destinations.	Veillez contacter notre service desk.
MSG00003	Erreur interne dans l'application BCSS	Veillez contacter notre service desk.
MSG00004	Problème avec le message reçu : message non conforme au schéma, pas de message XML valide, pas de message SOAP valide,	Validez votre message avant de l'envoyer à la BCSS.
MSG00014	Problème avec l’authentification.	Vérifiez si un certificat valide est utilisé, connu à la BCSS et conforme à [5], pour authentification TLS-client,



		et, si applicable, pour signature numérique. Vérifiez si le message contient la bonne identification client.
MSG00015	Pas d'accès au service demandé.	Pour les nouveaux accès, contactez le chef de projet concerné à la BCSS. Pour les échanges de données existants, veuillez contacter notre service desk.
MSG00016	Signature numérique invalide	Vérifiez si un certificat valide est utilisé pour signature numérique, connu à la BCSS et si la signature numérique est conforme à [5].
MSG00025	Vous utilisez une fonctionnalité qui n'est pas encore/plus supportée.	Corrigez votre message avant de l'envoyer à la BCSS.

Quelques exemples de messages d'erreur techniques:

- 1) Le nom d'hôte et la porte de l'url auquel la requête est envoyée sont valides, mais le nom de chemin de l'url ne l'est pas.

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <env:Fault>
      <faultcode>env:Client</faultcode>
      <faultstring>Internal Error</faultstring>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

- 2) Le message de la requête est invalide vis-à-vis du WSDL

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <soapenv:Fault>
      <faultcode>soapenv:Client</faultcode>
      <faultstring>Validation error</faultstring>
      <faultactor>http://www.ksz-bcss.fgov.be</faultactor>
      <detail>
        <ns:faultInfo xmlns:ns="http://kszbcss.fgov.be/inf/faultinfo">
          <informationCustomer>
            <ticket>alb2c3d4e5</ticket>
            <timestampSent>2010-08-25T13:38:56.854Z</timestampSent>
            <customerIdentification>
              <cbeNumber>000000000</cbeNumber>
            </customerIdentification>
          </informationCustomer>
          <informationCBSS>
            <ticketCBSS>00000000-0000-0000-0000-000000000000</ticketCBSS>
            <timestampReceive>0001-01-01T00:00:00-00:00</timestampReceive>
            <timestampReply>0001-01-01T00:00:00-00:00</timestampReply>
          </informationCBSS>
          <detail>
            <severity>FATAL</severity>
            <reasonCode>MSG00014</reasonCode>
            <diagnostic>Validation error</diagnostic>
            <authorCode>http://www.ksz-bcss.fgov.be/</authorCode>
            <information>
              <fieldName>error message</fieldName>
              <fieldValue>https://10.104.128.16:4520/PensionAuthorizedActivityService/consult:
cvc-particle 2.1: in element
{http://kszbcss.fgov.be/intf/PensionAuthorizedActivity/v1}stopSubscriptionNotificationRequest
of type
{http://kszbcss.fgov.be/types/PensionAuthorizedActivity/v1}StopSubscriptionNotificationRequest
Type, found &lt;ssin> (in default namespace), but next item should be
legalContext</fieldValue>
            </information>
          </detail>
        </ns:faultInfo>
      </detail>
    </soapenv:Fault>
  </soapenv:Body>
</soapenv:Envelope>
```



```

</detail>
</soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
  
```

3) Le service a été contacté, mais une erreur interne s'est produite

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <soapenv:Fault>
      <faultcode>soapenv:Server</faultcode>
      <faultstring>Internal error</faultstring>
      <faultactor>http://www.ksz-bcss.fgov.be</faultactor>
      <detail>
        <svc:livingWagesFault xmlns:svc="http://kszbcss.fgov.be/intf/livingwagesservice/v1">
          <informationCustomer>
            <ticket>alb2c3d4e5</ticket>
            <timestampSent>2010-08-25T13:38:56.854Z</timestampSent>
            <customerIdentification>
              <cbeNumber>000000000</cbeNumber>
            </customerIdentification>
          </informationCustomer>
          <informationCBSS>
            <ticketCBSS>00000000-0000-0000-0000-000000000000</ticketCBSS>
            <timestampReceive>0001-01-01T00:00:00-00:00</timestampReceive>
            <timestampReply>0001-01-01T00:00:00-00:00</timestampReply>
          </informationCBSS>
          <detail>
            <severity>FATAL</severity>
            <reasonCode>MSG00003</reasonCode>
            <diagnostic>Internal error</diagnostic>
            <authorCode>http://www.ksz-bcss.fgov.be</authorCode>
          </detail>
        </svc:livingWagesFault>
      </detail>
    </soapenv:Fault>
  </soapenv:Body>
</soapenv:Envelope>
  
```

4) Un problème s'est produit lors du contact du fournisseur

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <soapenv:Fault>
      <faultcode>soapenv:Server</faultcode>
      <faultstring>Error in communication with the destination/supplier</faultstring>
      <faultactor>http://www.ksz-bcss.fgov.be</faultactor>
      <detail>
        <svc:livingWagesFault xmlns:svc="http://kszbcss.fgov.be/intf/livingwagesservice/v1">
          <informationCustomer>
            <ticket>alb2c3d4e5</ticket>
            <timestampSent>2010-08-25T13:38:56.854Z</timestampSent>
            <customerIdentification>
              <cbeNumber>000000000</cbeNumber>
            </customerIdentification>
          </informationCustomer>
          <informationCBSS>
            <ticketCBSS>00000000-0000-0000-0000-000000000000</ticketCBSS>
            <timestampReceive>0001-01-01T00:00:00-00:00</timestampReceive>
            <timestampReply>0001-01-01T00:00:00-00:00</timestampReply>
          </informationCBSS>
          <detail>
            <severity>FATAL</severity>
            <reasonCode>MSG00002</reasonCode>
            <diagnostic>Error in communication with the destination/supplier</diagnostic>
            <authorCode>http://www.ksz-bcss.fgov.be</authorCode>
          </detail>
        </svc:livingWagesFault>
      </detail>
    </soapenv:Fault>
  </soapenv:Body>
</soapenv:Envelope>
  
```

5) La fonctionnalité demandé n'est pas disponible

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  
```




```
<soapenv:Body>
  <soapenv:Fault>
    <faultcode>soapenv:Server</faultcode>
    <faultstring>This functionality is not available</faultstring>
    <faultactor>http://www.ksz-bcss.fgov.be</faultactor>
    <detail>
      <svc:livingWagesFault xmlns:svc="http://kszbcss.fgov.be/intf/livingwagesservice/v1">
        <informationCustomer>
          <ticket>alb2c3d4e5</ticket>
          <timestampSent>2010-08-25T13:38:56.854Z</timestampSent>
          <customerIdentification>
            <cbeNumber>000000000</cbeNumber>
          </customerIdentification>
        </informationCustomer>
        <informationCBSS>
          <ticketCBSS>00000000-0000-0000-0000-000000000000</ticketCBSS>
          <timestampReceive>0001-01-01T00:00:00-00:00</timestampReceive>
          <timestampReply>0001-01-01T00:00:00-00:00</timestampReply>
        </informationCBSS>
        <detail>
          <severity>FATAL</severity>
          <reasonCode>MSG00025</reasonCode>
          <diagnostic>This functionality is not available</diagnostic>
          <authorCode>http://www.ksz-bcss.fgov.be/</authorCode>
        </detail>
      </svc:livingWagesFault>
    </detail>
  </soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
```

4.14 Messages en batch ou lot de messages

Les blocs suivants sont présents pour un scénario de requête/réponse:

- requête: sender, receiver, et legalContext obligatoires
- réponse: sender, receiver, et legalContext obligatoires

En batch, un équivalent d'un soap fault n'est pas prévu pour les messages d'erreur techniques, parce qu'il n'est pas requis. Lors de problèmes de ressource, l'opérateur batch reprend le traitement du message dès que ces problèmes sont résolus. Lors de problèmes relatifs à la structure des messages ou d'autres problèmes de blocage, l'opérateur batch établit un contact afin de garantir un bon suivi.

Pour des scénarios de notification/de push, tels que des mutations, les trois blocs suivants doivent être présents:

- sender
- receiver
- legalContext

Note: Un bloc statut peut être ajouté au niveau du root d'une définition d'un fichier de lot, mais n'est normalement pas ajouté. En général, un lot n'a pas de statut global. Des métadonnées peuvent, si nécessaire, être ajoutées à un fichier voucher.

Les lots de messages maintiennent, dans la mesure du possible en fonction des besoins du service, une structure, telle que celle affichée dans les exemples sous 3.5., notamment:

- ✓ un tag XML obligatoire contenant une liste de plusieurs occurrences.
- ✓ telle qu'affichée ci-dessous, la liste des occurrences de 0 à n d'un bloc de données
- ✓ tel que dans ce bloc de données, un champ NISS



4.15 Numéro d'ordre

Lors de services de notification en batch, l'ordre des messages est souvent important. Dans ce cas, un élément *sequenceNumber* est prévu dans la définition de message. Cet élément est du type `xsd:long` (valeur minimale 1, nombre de chiffres maximaux : 15). Voir aussi 3.5.2 pour un exemple d'une définition de message contenant cet élément.

Le numéro d'ordre est croissant et continu, de sorte que les messages puissent être mis dans le bon ordre et que des messages manquants puissent être détectés. Cette valeur n'est pas valable à travers les différents services, mais uniquement au sein d'un même service et d'une opération.

4.16 Dates

Pour la définition d'un champ spécifiant une date, le type `xsd:date` du schéma standard XML est utilisé.

Ce type de données contient un fuseau horaire optionnel. La BCSS n'impose pas de limitations supplémentaires à la spécification d'un fuseau horaire dans les définitions de message. Une telle limitation causerait des problèmes lors de l'utilisation d'outils et de libraires de tiers qui créent des messages XML.

Si le fuseau horaire manque, celui d'application en Belgique est utilisé, c'est-à-dire, UTC+01:00 (CET) pour des dates qui tombent dans la saison d'hiver et UTC+02:00 pour des dates dans la saison d'été.

Si le fuseau horaire est bien spécifié, le fuseau horaire correct d'application en Belgique pour cette date doit être utilisé, soit +01:00 pour des dates qui tombent dans la saison d'hiver et +02:00 pour des dates dans la saison d'été.

Corrects:

- 2012-01-01
- 2012-01-01+01:00
- 2012-07-01
- 2012-07-01+02:00

Erronés:

- 2012-01-01+02:00 (saison d'été au lieu de saison d'hiver)
- 2012-07-01+01:00 (saison d'hiver au lieu de saison d'été)
- 2012-01-01+01:15 (fuseau horaire pas d'application en Belgique)
- 2012-01-01+03:00 (fuseau horaire pas d'application en Belgique)
- 2012-01-01Z (fuseau horaire pas d'application en Belgique)

L'utilisation d'un fuseau horaire erroné produit souvent des résultats inattendus.

Parfois une définition de message requiert l'admission d'une date incomplète, c'est-à-dire une date ne spécifiant pas le jour ni le mois. Pour ce faire, le type `IncompleteDateType` de [1] est utilisé. Ce type ne permet pas de spécifier un fuseau horaire. Les valeurs admises suivent une des séquences suivantes:

- yyyy-mm-dd
- yyyy-mm-00
- yyyy-00-00



4.17 Timestamps

Pour la définition d'un champ représentant un timestamp, le type `xsd:dateTime` du schéma standard XML est utilisé.

Ce type de données contient un fuseau horaire optionnel. Si le fuseau horaire manque, celui d'application en Belgique est utilisé, c'est-à-dire, UTC+01:00 (CET) pour des dates qui tombent dans la saison d'hiver et UTC+02:00 pour des dates dans la saison d'été. Si le fuseau horaire est bien spécifié, celui doit être pris en compte.

L'utilisation du caractère 'Z' comme fuseau horaire représente le fuseau horaire universel ('Zulu time' ou UTC). Ce fuseau horaire est souvent utilisé dans les timestamps générés par la BCSS.

Quelques exemples

2012-01-01T17:00:00+01:00 représente 17.00 h selon le fuseau horaire belge normal

2012-01-01T17:00:00+02:00 représente 16.00 h selon le fuseau horaire belge normal

2012-01-01T17:00:00Z représente 18.00 h selon le fuseau horaire belge normal
(saison d'hiver)

2012-07-01T17:00:00Z représente 19.00 h selon le fuseau horaire belge normal
(saison d'été)

4.18 Résumé des données standard

Services web

Requête:

- ✓ `informationCustomer`
- ✓ `legalContext`
- ✓ données de la requête en fonction des besoins du service:
 - seul un NISS
 - bloc contenant plusieurs critères de requête (ex. `NISSet period`)
 - ...

Un bloc optionnel *information CBSS* est présent pour des raisons techniques internes à la BCSS, mais n'est pas utilisé.

Réponse:

- ✓ `informationCustomer`, repris de la requête
- ✓ `informationCBSS`
- ✓ `legalContext`, repris de la requête
- ✓ données de requête (repris de la requête)
- ✓ statut

Lot de messages (type de requête/réponse) - requête

Requête:

L'élément root de la requête est suffixé par « Request ».

Bloc standard :

- ✓ `sender`
- ✓ `receiver`
- ✓ `legalContext`
- ✓ Critères en fonction des besoins :



- Un champ NISS (le cas échéant pour le service)
- Un bloc avec plusieurs critères de recherche (par exemple période)

Réponse:

L'élément root de la réponse est suffixé par « Response ».

Bloc standard :

- ✓ sender
- ✓ receiver, données copiées de la requête
- ✓ legalContext, repris de la requête
- ✓ Un tag XML contenant une liste de plusieurs occurrences
 - Voici ci-dessous la liste des occurrences de 0 à n d'un bloc de données
 - Dans ce bloc de données un champ NISS (si d'application pour un service)

Lot de messages (type notification ou mutation)

Notifications du fournisseur à BCSS:

L'élément root de la réponse commence par « notify' » et si une réponse est attendue, elle sera suffixée par « Request ».

Bloc standard :

- ✓ sender
- ✓ receiver
- ✓ legalContext
- ✓ sequenceNumber (si l'ordre de traitement est important)
- ✓ Un tag XML contenant une liste de plusieurs occurrences
 - Voici ci-dessous la liste des occurrences de 0 à n d'un bloc de données
 - Dans ce bloc de données un champ NISS (si d'application pour un service)

Réponse de la BCSS au fournisseur (le cas échéant):

L'élément root de la réponse commence par « notify' » et suffixée par « Response ».

Bloc standard :

- ✓ sender
- ✓ receiver
- ✓ legalContext
- ✓ sequenceNumber (si l'ordre de traitement est important)
- ✓ Un tag XML contenant une liste de plusieurs occurrences
 - Voici ci-dessous la liste des occurrences de 1 à n d'un bloc de données
 - Dans ce bloc de données un champ NISS (si d'application pour un service) et le résultat du traitement des notifications. (Par ex. Vérification de l'intégration Affichage, validation NISS, la distribution de l'information, ...)

Distribution des notifications de la BCSS:

L'élément root de la réponse commence par « notify' ».

Bloc standard :

- ✓ sender
- ✓ receiver
- ✓ legalContext
- ✓ sequenceNumber (si l'ordre de traitement est important)
- ✓ Un tag XML contenant une liste de plusieurs occurrences
 - Voici ci-dessous la liste des occurrences de 1 à n d'un bloc de données
 - Dans ce bloc de données un champ NISS (si d'application pour un service)



4.19 Comparaison d'un header A1 avec les données standard XML

Le tableau ci-après donne un aperçu des données d'un header A1, ainsi que les champs correspondants dans une définition de service XML.

<i>header A1</i>	<i>définition de service XML standard</i>
secteur+institution	informationCustomer/customerIdentification: avec secteur+institution ou pour batch : sender/customerIdentification : avec secteur + institution
Code qualité (+ code phase = 000)	/
/	legalContext
Période du message	/ (Pas prévue à une position standard.)
Période d'inscription	/
User ID (= numéro de programme)	/
	Services web: Authentification du client à l'aide de 2-way SSL Le partenaire est responsable pour l'authentification de ses utilisateurs et pour les autoriser à accéder aux services web.
	Batch: authentification via le protocole utilisé (FTP, SFTP, WebDAV, ...)
Formulaire et type de traitement	Nom d'opération: Premier élément en soap body Service web: Nom du service+ opération Batch: Code d'application+ code d'opération
NISS	ssin (présent si applicable au service)
Référence du demandeur	informationCustomer/ticket ou pour batch : sender/ticket ou receiver/ticket
Référence de la réponse	informationCBSS/ticketCBSS ou pour batch : sender/ticket ou receiver/ticket
Code retour application	statut
Code retour réseau	soap fault (voir messages d'erreur techniques)
Timestamps de l'envoi / des messages reçus	informationCustomer/timestampSent informationCBSS/timestampReceive informationCBSS/timestampReply ou pour batch : sender/timestampSent receiver/timestampSent



5 Routage par les institutions de gestion: FAQ

Étant donné que les besoins des institutions relatifs au routage sont divergents et difficiles à généraliser, nous essaierons de répondre aux questions fréquemment posées.

Q: Les données sur la base desquelles le routage d'un message devrait s'effectuer, ne se situent pas dans un bloc isolé, mais dans la partie données. Ceci rend l'implémentation d'un mécanisme de routage générique impossible, requérant un nouveau développement pour chaque service.

A: Plusieurs données standard (informationCustomer, informationCBSS, legalContext) se situent obligatoirement dans une position fixe dans chaque définition de message. Elles peuvent aussi être extraites du message par un programme, par exemple à l'aide des expressions XPath.

Un NISS est uniquement présent dans la définition de message lorsqu'il est utile à la fonctionnalité du service. Par exemple, il ne sert à rien de prévoir un NISS dans la requête lors d'une consultation de l'ensemble des collaborateurs d'une entreprise. Si le NISS est présent, le champ est dénommé *ssin*. Il peut aussi être extrait du message de manière générique, par exemple à l'aide d'une expression XPath.

Q: Les données standard sont insuffisantes pour effectuer le routage des données: le UserId manque.

Un UserId de 11 chiffres, tel qu'affiché dans le préfixe A1, n'est, en effet, pas prévu. Pour la protection des services web, des mécanismes plus sûrs sont prévus, tels que l'utilisation de certificats. Si cette donnée a été utilisée pour le routage par une institution de gestion, le champ ticket de message (longueur max. de 36 caractères ou UUID) peut être utilisé afin d'offrir une solution équivalente.

Q: Les données standard sont insuffisantes pour effectuer le routage des données: le code qualité manque.

A: Le header code qualité n'est, en effet, plus présent. Cependant, une nouvelle donnée est présente: le cadre légal (legal context). Cette donnée indique la finalité d'une consultation envoyée à la BCSS ou d'une notification envoyée par la BCSS.

L'introduction de ce concept dissipe l'ambiguïté qui s'était installée entre les codes qualité et les codes institution. Le contexte légal indique le cadre dans lequel un accès est donné à certaines données, conformément à l'autorisation. En fonction du contexte légal, la BCSS applique les contrôles attendus, tels que l'existence d'une inscription avec un ou plusieurs codes qualité admis.

Un contexte légal peut donner lieu à un contrôle sur plusieurs codes qualité. Un seul message peut aussi être envoyé plusieurs fois à un destinataire pour deux différents contextes légaux.

Q: Les données standard sont insuffisantes pour effectuer le routage de données: la période d'intégration manque.

A: La période d'intégration n'est, en effet, pas prévue de manière standard dans les définitions de message. Les besoins concrets de prévoir cette période dans une position standard dans le message ne nous sont pas clairs actuellement. Veuillez contacter la BCSS à ce sujet.



Q: Les données standard sont insuffisantes pour effectuer le routage des données: le service ou le formulaire manque.

A:

Un service, tant dans le cas d'un service web que dans le cas d'un service batch, regroupe une ou plusieurs fonctionnalités dans les opérations.

Service web:

Chaque service web est défini par un WSDL et accessible via un URL unique. La BCSS utilise aussi un namespace XML unique pour chaque définition de service.

Le nom d'opération peut être déterminé comme le premier élément d'un soap body.

De manière alternative, le http soapAction header peut aussi désigner un service et une opération de façon unique.

Batch:

Les noms de service et d'opération sont mentionnés dans les champs *applicationCode* et *operationCode* du voucher. Il est convenu que le nom d'opération est aussi utilisé comme l'élément root dans un fichier batch en XML.

Q: Les données standard sont insuffisantes pour effectuer le routage des données: la période de message manque.

Une période de message n'est, en effet, pas prévue de manière standard dans les définitions de message.

Notez que la 'période de message' n'est pas toujours univoque. Ainsi, un message relatif à une allocation de chauffage est associé, tant à une saison de chauffage qu'à une date de décision quant à l'allocation. En fonction du partenaire, le contrôle d'intégration est requis sur la base de la saison de chauffage ou sur la base de la date de décision.

Lors d'une consultation, les besoins peuvent aussi varier: Parfois une période de recherche s'avère utile et à d'autres instants, seule une date de recherche, un trimestre ou aucune indication de temps suffit.

Si une période de message s'avère nécessaire pour le routage des messages, une extraction de celle-ci à partir du message pourrait être paramétrée par service.

Q: Comment les valeurs du champ `legalContext` sont-elles déterminées lors du passage à un service dans un nouveau format?

Les valeurs du contexte légal sont déterminées par la BCSS en concertation avec les organisations partenaires.

Q: Les données de routage ne sont pas séparées d'autres données sensibles auxquelles l'institution de gestion n'a pas accès. Ceci rend le routage par l'institution de gestion impossible, étant donné que celle-ci ne peut pas avoir accès aux données sensibles.

A: D'un point de vue technique, ce procédé a le même niveau de confidentialité que les messages A1:

- Pour les messages A1, une extraction d'un nombre de caractères fixe s'effectue au début du message. Le reste du message doit rester caché.



- ✓ Pour les messages XML, une extraction de tags XML fixes s'effectue dans le message. Le reste du message doit rester caché.

Les données sensibles ne sont pas chiffrées dans les deux cas. Par conséquent, il a lieu de faire confiance à la bonne programmation par l'institution de gestion afin de continuer à cacher les données.

Si la confidentialité doit être garantie, une solution de chiffrement peut être discutée avec la BCSS.

6 Point de contact central pour les problèmes

En cas de problèmes, veuillez contacter le service desk

- ✓ par téléphone au numéro 02-741 84 00 entre 8h00 et 16h30 les jours ouvrables,
- ✓ par courriel à l'adresse suivante: servicedesk@ksz-bcss.fgov.be,

Veuillez communiquer les informations suivantes concernant le problème:

- ✓ Pour les services en ligne
 - Message question et réponse, ou si impossible
 - Ticket du message, il s'agit du ticket BCSS (de préférence) ou de la référence du message que le client a lui-même ajouté au message
 - Date et l'heure du request
 - L'environnement dans lequel le problème se manifeste (acceptation ou production)
 - Le nom du service (tel que fourni par la BCSS)
- ✓ Pour les flux en mode batch
 - L'environnement dans lequel le problème se manifeste (acceptation ou production)
 - Nom du fichier
 - Nom du flux ou du projet
 - Éventuellement, la date et l'heure de l'envoi, le nom du flux ou du projet, et le répertoire dans lequel le fichier a été placé ou le serveur sur lequel il a été placé

Si vous souhaitez obtenir de plus amples informations relatives au service desk, nous vous invitons à consulter notre site web.