

# Directive XX.2023: Système de surveillance et de signalement des paris

Ce document contient les exigences auxquelles les bookmakers doivent se conformer lorsqu'ils communiquent les données de paris à l'Autorité nationale des paris de Chypre.

## Exigences en matière de données de l'API — Guide de l'utilisateur

## Historique des versions

Version	Date	Description
1.0	7.8.2023	-

# Table des matières

## Sommaire

Table des matières.....	3
<b>Directive XX.2023: Système de surveillance et de signalement des paris.....</b>	<b>4</b>
<b>Annexe A: Exigences techniques.....</b>	<b>5</b>
1.0 Introduction.....	5
<b>2.0 Le modèle de données du BMRS.....</b>	<b>6</b>
2.1 Bookmakers, représentants autorisés et magasins (titulaires de licence).....	7
2.2 Joueurs et comptes.....	7
2.3 Le bulletin de pari.....	8
<b>3.0 Les processus de l'API du BMRS.....</b>	<b>11</b>
3.1 Comptes des joueurs.....	13
3.1.1 Méthode SaveAccounts.....	13
3.2 Transactions de compte.....	19
3.2.1 Méthode CreateAccountTransactions.....	19
3.3 Restrictions de compte.....	25
3.3.1 Méthode SaveAccountRestrictions.....	26
3.4 Activité de pari.....	31
3.4.1 Méthode CreateBetSlips.....	32
3.4.3 Méthode UpdateBetSlips.....	43
3.4.5 Méthode SaveBetLines.....	52
3.5 Appels système.....	56
3.5.1 Méthode Heartbeat.....	56
3.5.2 Méthode GetErrorLogs.....	57
<b>4.0 Protocoles de communication BMRS.....</b>	<b>61</b>
4.1 Protocole de communication primaire — L'API BMRS.....	61
4.2 Protocole de communication secondaire — Fichiers XML.....	63

## **Directive XX.2023: Système de surveillance et de signalement des paris**

La présente directive est rédigée et émise par l'Autorité nationale des paris (ci-après la «NBA») conformément aux dispositions de l'article 15, paragraphe b) de la loi de 2009 sur les paris (L.37(I)/2019), telle que modifiée ou remplacée, et énonce toutes les exigences et procédures techniques que les titulaires de licence de classe A et B doivent adopter pour se connecter et transmettre des informations à la NBA via le système de surveillance et de signalement des paris de la NBA (ci-après «BMRS»). Elle prescrit les types de données qui doivent être transmises ainsi que les instances ou les délais qui déclenchent la transmission.

2. Tous les titulaires de licence de classe A et B doivent:

- a) adopter et mettre en œuvre toutes les exigences et procédures techniques de l'annexe A;
- b) se connecter au BRMS conformément à l'annexe A;
- c) transmettre des données aux moments et dans les cas prévus à l'annexe A.

3. La présente directive est contraignante et sa mise en œuvre est obligatoire pour tous les titulaires de licence des classes A et B.

# Annexe A: Exigences techniques

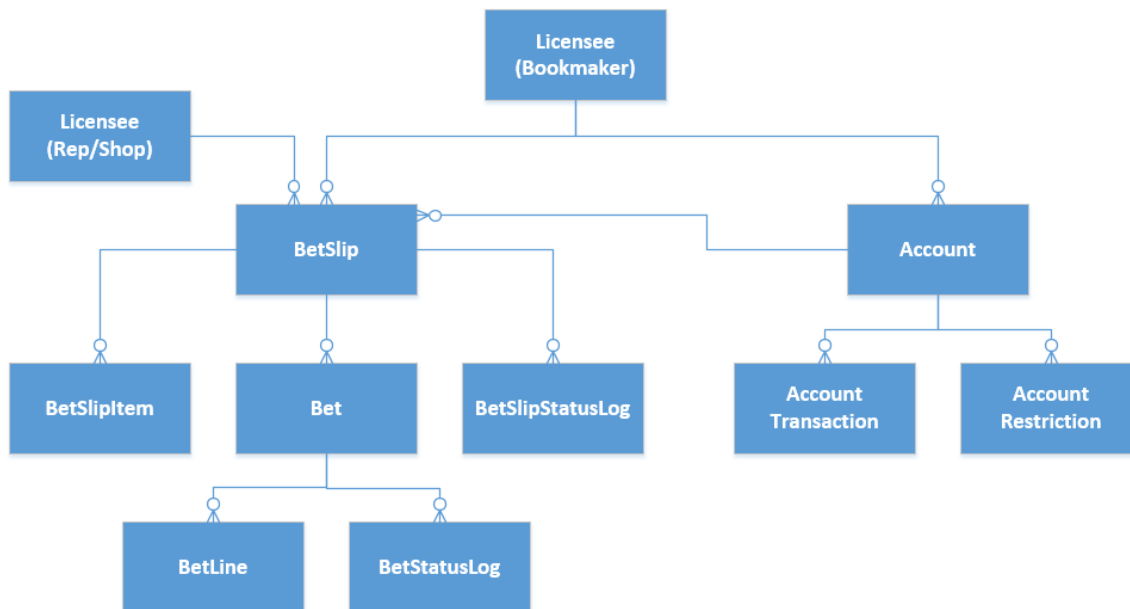
## 1.0 Introduction

Le BMRS est une plateforme de surveillance développée par la NBA dans le cadre de l'amélioration de son cadre de régulation centré sur les données en suivant de près l'activité de tous les participants du marché des paris en temps quasi réel. Les bookmakers qui facilitent les activités de paris en ligne (titulaires de licence de classe B) ou par l'intermédiaire de représentants autorisés et magasins de détail (titulaires de licence de classe A) sont censés s'engager dans une communication continue avec la plateforme par l'intermédiaire d'une API conçue appropriée et transmettre toutes les informations que la NBA juge nécessaires pour remplir son mandat réglementaire.

Ce document décrit en détail les différents processus que le système met à la disposition des utilisateurs finaux (les titulaires de licence), le but et les champs de données requis pour chacun de ces processus, la forme attendue à laquelle les transmissions de données doivent se conformer (schémas XML), les réponses et les erreurs potentielles résultant de la validation des données ainsi qu'une description générale des protocoles de communication utilisés.

## 2.0 Le modèle de données du BMRS

Le BMRS collecte et traite les données d'une manière qui permet à la fois une vue d'ensemble pertinente des tendances et des comportements du marché et, le cas échéant, une ventilation détaillée des activités de paris au niveau individuel. Sa conception est basée sur la modélisation de certaines entités clés et les processus par lesquels elles interagissent les unes avec les autres pour façonner collectivement les opérations de l'entreprise de paris. Un aperçu du modèle est illustré à la figure 2.1.



**Figure 2.1:** Aperçu du modèle de données du BMRS.

Pour chaque pari, il y a deux contreparties, à savoir le bookmaker (titulaire de licence) qui offre certaines cotes pour un résultat particulier d'un événement sous-jacent futur, et le joueur qui accepte ces cotes et décide de placer un certain montant de mise sur celles-ci. Toutes les informations relatives à un pari particulier sont contenues sur un bulletin de pari, qui devient une forme valide d'accord une fois envoyé par le joueur et accepté par le bookmaker. Par la suite, le bulletin de pari suit un certain cycle de vie, qui se termine soit une fois que tous les événements sur lesquels le joueur a misé sont conclus sans paiement, soit sur le paiement de tout gain réalisé (ou remboursement).

## 2.1 Bookmakers, représentants autorisés et magasins (titulaires de licence)

Les **bookmakers** facilitent les activités de paris en ligne (classe B) ou par l'intermédiaire de **représentants autorisés** et **magasins** de détail (classe A). Dans le premier cas, ils fournissent des plateformes sur lesquelles les joueurs peuvent enregistrer un **compte** et soumettre ensuite des **bulletins de paris**. Dans les sites de paris physiques, les joueurs n'enregistrent pas leurs coordonnées via un compte, mais soumettent leurs paris par l'intermédiaire des caisses du magasin.

Étant donné que ces trois entités sont réglementées par la loi, les détails concernant leur structure et leur configuration opérationnelle sont suivis par un certain nombre de systèmes en dehors du BMRS. Lors des échanges de données par l'intermédiaire des méthodes énumérées à l'article 3.0, le BMRS identifie de manière unique chaque expéditeur, mais la NBA ne s'appuie pas sur ce canal de communication particulier pour gérer les enregistrements détaillés de ces entités.

## 2.2 Joueurs et comptes

Dans le cas des paris en ligne, une personne ne peut soumettre un pari qu'après avoir ouvert un **compte** auprès d'un bookmaker de classe B. Le compte, qui est géré sur la plateforme du bookmaker, contient certaines données d'identification de son titulaire et conserve un enregistrement des fonds à sa disposition pour effectuer des paris. La plateforme du bookmaker facilite un certain nombre de **transactions** qui affectent le solde du compte, comme le dépôt de fonds supplémentaire par le titulaire du compte, le retrait des fonds et le suivi de tout bonus que le bookmaker aurait pu offrir au joueur sous certaines règles et conditions. À la suite de ce qui précède, un joueur a également la possibilité d'imposer certaines limites à ses activités de paris au moyen de **restrictions**. Dans certains cas, les **restrictions** peuvent également être appliquées sur le compte d'un joueur par le bookmaker pour suspicion de toute violation des règles.

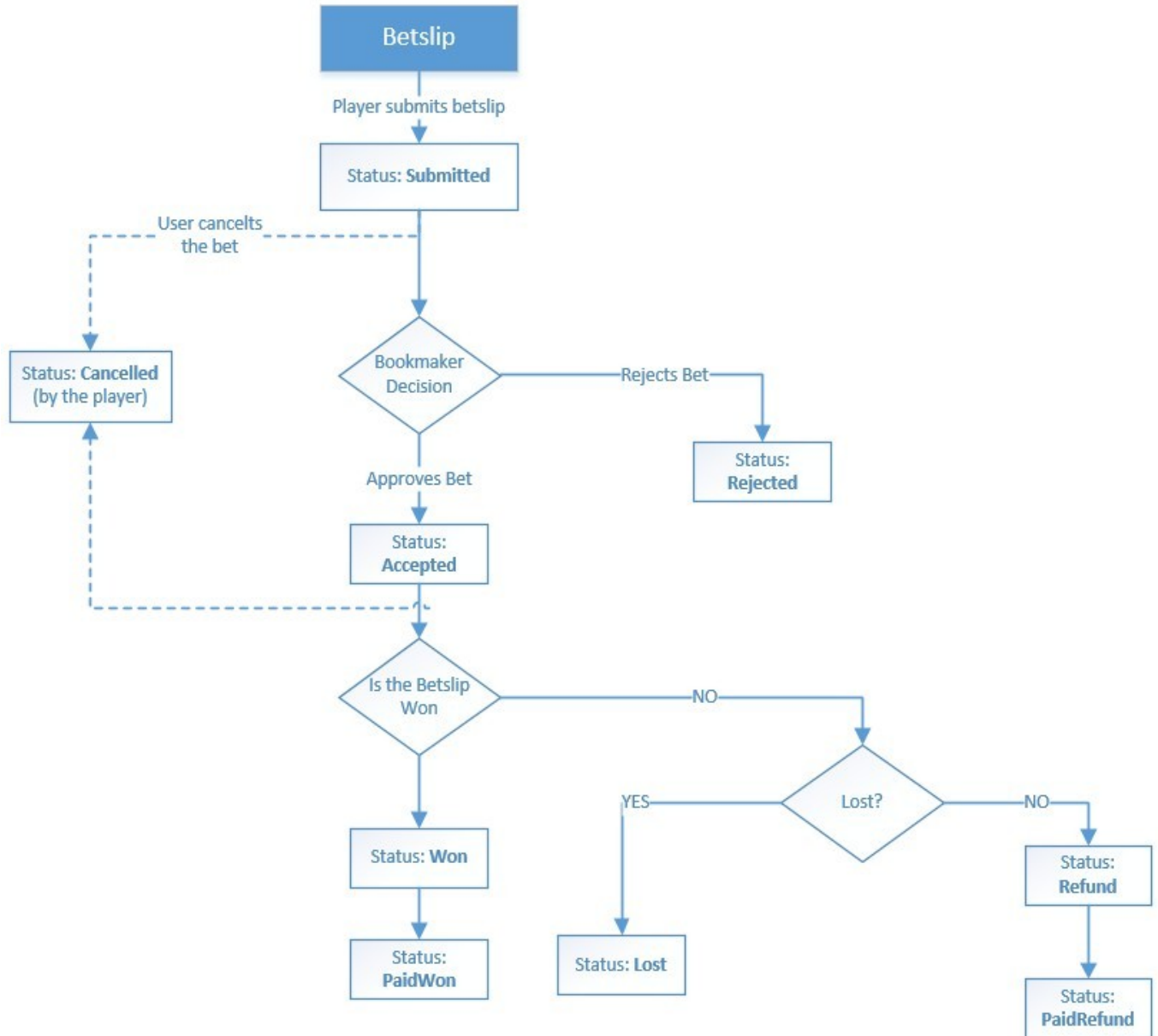
Dans les sites de paris physiques, le joueur n'a actuellement pas besoin d'un compte pour participer aux paris, et la plupart de ce qui a été décrit dans le paragraphe précédent ne sont pas pertinents. Dans le cadre du modèle du BMRS, le compte d'un joueur dans le contexte des paris en magasin représente la caisse du magasin par l'intermédiaire de laquelle un bulletin de

pari a été envoyé.

## 2.3 Le bulletin de pari

La base du modèle de données du BMRS utilisé pour représenter le trafic des paris est le bulletin de pari (**bulletin de pari**). Un bulletin de pari peut être émis dans un magasin dans le cas d'une licence de type classe A ou en ligne dans le cas de la classe B. Il est composé d'un ou de plusieurs **éléments du bulletin de pari**, chacun d'entre eux contenant des informations sur un événement sous-jacent, un marché associé à cet événement et le résultat de ce marché (sélection) sur lequel le joueur souhaite placer un pari contre les cotes offertes par le bookmaker. Un **pari**, sur lequel le joueur place un montant de mise, peut impliquer un ou plusieurs de ces éléments du bulletin de pari (événements, marchés et sélections du joueur) qui forment ensemble un ensemble de combinaisons possibles ou des **lignes de pari**. Étant donné que chacune de ces combinaisons est composée de différents éléments, les cotes diffèrent d'une ligne à l'autre et un pari peut donc se voir attribuer une gamme de cotes offertes.

Une fois créé, un bulletin de pari suit un cycle de vie spécifique, qui est conclu après que tous les paris qu'il contient ont été réglés, c'est-à-dire que tous ses événements sous-jacents sont terminés, et tous les gains réalisés sont payés ou crédités sur le compte du joueur. Un certain nombre d'étapes intermédiaires existent entre ces points de départ et de fin, et ces étapes peuvent varier d'un système de bookmaker à un autre. Le modèle du BMRS pour le cycle de vie du bulletin de pari est représenté à la figure 2.2. Tout système externe d'envoi de données au BMRS doit être ajusté pour suivre ce cycle aussi fidèlement que possible en établissant un lien entre ses propres statuts d'un bulletin de pari et ceux indiqués dans la figure, qui sont expliqués plus en détail dans le tableau 3.18.



**Figure 2.2:** Le cycle de vie du bulletin de pari du BMRS.

Betslip	Bulletin de pari
Player submits betslip	Le joueur soumet le bulletin de pari
Status: <b>Submitted</b>	Statut: <b>Submitted</b>
User cancels the bet	L'utilisateur annule le pari
Status: <b>Cancelled</b> (by the player)	Statut: <b>Cancelled</b> (par le joueur)
Bookmaker Decision	Décision du bookmaker
Rejects Bet	Il rejette le pari
Status: <b>Rejected</b>	Statut: <b>Rejected</b>
Approves Bet	Il approuve le pari
Status: <b>Accepted</b>	Statut: <b>Accepted</b>
Is the Betslip Won	Le bulletin de pari est gagnant

Status: <b>Won</b>	Statut: <b>Won</b>
Status: <b>PaidWon</b>	Statut: <b>PaidWon</b>
YES	OUI
NO	NON
Status: <b>Lost</b>	Statut: <b>Lost</b>
Lost?	Lost?
Status: <b>Refund</b>	Statut: <b>Refund</b>
Status: <b>PaidRefund</b>	Statut: <b>PaidRefund</b>

### 3.0 Les processus de l'API du BMRS

Toute transmission de données par les entités réglementées à la NBA via le BMRS a lieu à l'aide d'un certain nombre de méthodes de l'API sous la forme de requêtes SOAP (les protocoles de communication sont décrits en détail à l'article 4.0). Les modèles de données que le BMRS utilise pour structurer les différents objets qui décrivent collectivement les processus et les activités qu'il est destiné à surveiller sont intégrés dans ces méthodes.

Ces activités peuvent être regroupées en fonction des objets sur lesquels elles agissent:

- Comptes des joueurs:
  - Créer et mettre à jour des comptes;
  - Créer des transactions de compte (dépôts, retraits, etc.);
  - Surveiller l'ajout, la suppression ou l'expiration des restrictions de compte;
- Activité de pari:
  - Créer des bulletins de pari;
  - Mettre à jour les bulletins de pari tout au long de leur cycle de vie;
- Données relatives au système BMRS:
  - Appel à Heartbeat;
  - Journal d'erreurs.

Le BMRS s'attend à recevoir des données pas plus de **90 secondes** après avoir été enregistré dans le système du bookmaker. Les méthodes de l'API permettent l'envoi de telles données par lots, ce qui signifie que plusieurs objets peuvent être créés ou mis à jour en un seul appel. Néanmoins, le choix pour le processus réel et la planification suivie revient à l'expéditeur et est seulement limité par la limite de 90 secondes.

La structure des données d'une méthode du BMRS prend la forme d'une arborescence XML. Certains éléments de cette arborescence sont simplement des conteneurs sous lesquels une collection d'un ou plusieurs sous-éléments de même type peuvent être répertoriés, tandis que d'autres contiennent des sous-éléments qui représentent des données.

L'élément <en-tête> qui est commun à toutes les méthodes, contient des informations de base qui identifient l'expéditeur des données comme indiqué dans le tableau 3.1. Les paramètres figurant dans <en-tête> sont fournis par la NBA avant de démarrer une connexion entre un système externe et le BMRS. Toute modification ultérieure de ces identifiants est communiquée par l'Autorité en temps utile.

Header				
N°.	Nom du champ	Type de données	Description	Commentaires
1	DataEntryKey	Chaîne de caractères	Une clé unique générée par le BMRS et fournie à chaque bookmaker par la NBA (GUID).	Cette valeur peut être remplacée (réémission) si nécessaire.
2	LicenseNumber	Chaîne de caractères	Le numéro de licence du bookmaker.	
3	LicenseeIdentifier	Chaîne de caractères	Une clé unique générée par le BMRS et fournie à chaque bookmaker par la NBA (GUID).	Cette valeur est <u>statique</u> . Une fois émise, elle ne peut pas changer car elle identifie de manière unique le bookmaker.

**Tableau 3.1:** Champs de données (sous-éléments) sous l'élément d'en-tête.

## 3.1 Comptes des joueurs

La majorité des processus contrôlés par le BMRS proviennent d'un compte joueur (par exemple, le placement d'un pari) ou ciblent un compte joueur (par exemple, une transaction résultant du gain d'un pari). En tant que tel, le compte d'un joueur est au cœur du modèle de données utilisé par le système et doit être référencé chaque fois qu'une action impliquant ce compte est appelée. En matière de flux de données, la conséquence de ce qui précède est que la création d'un compte de joueur doit précéder tout signalement impliquant ce compte.

Les comptes des joueurs sont créés et peuvent être mis à jour après leur création par la même méthode, *SaveAccounts*.

**Remarque:** Les comptes sont exigés à la fois pour les opérateurs de classe A et de classe B, mais les renseignements qu'ils contiennent dans le premier cas concernent les caisses des magasins, et non les parieurs.

### 3.1.1 Méthode *SaveAccounts*

#### 3.1.1.1 Description

Les comptes des joueurs créés ou mis à jour dans le système du bookmaker à tout moment doivent être communiqués au BMRS dans un délai de 90 secondes à l'aide de la méthode *SaveAccounts*. Les paramètres transmis par cet appel suivent la structure d'arborescence de type XML illustrée à la figure 3.1.

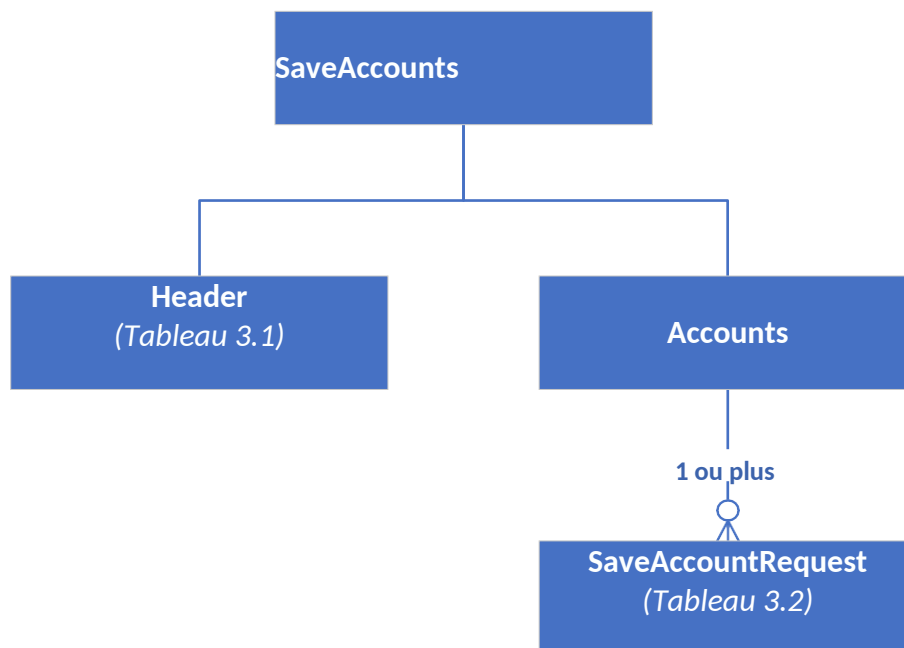
L'élément *en-tête* et son contenu ont déjà été décrits dans le tableau 3.1.

Le nœud *SaveAccountRequest* contient des sous-éléments qui représentent les champs de données liés au compte; ces champs sont énumérés dans le tableau 3.2. Un compte peut être désigné comme actif ou désactivé, comme décrit dans le tableau 3.3.

**Remarque:** Le champ de données **Username** d'un compte doit être un identifiant unique. Il est utilisé lors d'appels ultérieurs pour faire référence à ce compte (par exemple, *CreateBetSlips*, *CreateAccountTransactions* etc.).

À l'exception de l'**Username** (qui est un identifiant unique) et des paramètres **RegisteredOnDate**, tous les autres champs transmis lors de la méthode *SaveAccounts* peuvent être modifiés après la création d'un compte de joueur. Ainsi, si, à un moment donné, un joueur modifie certaines de ces informations dans le système du bookmaker, les modifications doivent être reflétées dans l'appel ultérieur à l'API du BMRS en incluant l'ensemble complet des données du compte dans un élément *SaveAccountRequest* sous le nœud Comptes.

### 3.1.1.2 Méthode Request



**Figure 3.1:** Structure XML de la méthode *SaveAccounts*.

SaveAccountRequest				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Username	Chaîne de caractères	L'identifiant du compte. <b>L'identifiant unique</b> de ce compte.	Cela ne doit pas nécessairement être un nom d'utilisateur réel, par exemple, il peut s'agir d'un numéro de référence.
2	RegisteredOnDate	Date et heure	La date et l'heure UTC à laquelle ce compte a été créé dans le système du bookmaker.	Ce paramètre est défini lorsque le compte est créé et ne peut pas être mis à jour par des appels ultérieurs.
3	FullName	Chaîne de caractères	Nom et prénom du titulaire du compte.	
4	IdentityDocumentType	Chaîne de caractères	Le type de document d'identification du titulaire du compte (carte d'identité nationale ou passeport)	
5	IdentityNumber	Chaîne de caractères	Le numéro d'identité du titulaire du compte (selon le type de document d'identité sélectionné)	
6	IdentificationDocumentIssuingCountry	Chaîne de caractères	Le code du pays émetteur du document d'identification (carte d'identité nationale ou passeport).	<b>ISO-Alpha-2</b> ou <b>ISO-Alpha-3.</b>
7	DateOfBirth	Date et heure	La date de naissance du titulaire du compte.	
8	Status	Énumération [Active] [Disabled]	Le statut actuel du compte.	Voir le tableau 3.3 pour plus de détails.
9	IsVerified	Boolean	Un indicateur précisant si l'identité/les détails du titulaire du compte ont été vérifiés par le bookmaker.	
10	Description	Chaîne de caractères	Toute information complémentaire.	

**Tableau 3.2:** Champs de données (sous-éléments) sous l'élément SaveAccountRequest.

Status	Description
Active	Le compte est ouvert aux transactions (dépôt, retrait, etc.) et/ou peut être impliqué dans des activités de paris.
Disabled	Le compte ne peut pas être utilisé pour des transactions ou des paris.

**Tableau 3.3:** *SaveAccountRequest* — valeurs du champ **Status**.

**Remarque:** Les nœuds multiples *SaveAccountRequest* peuvent être inclus dans un seul appel à la méthode *SaveAccounts*, y compris les nouveaux comptes et les comptes existants qui ont été mis à jour. Un compte ne doit être inclus dans le lot qu'une seule fois; plusieurs occurrences du même identifiant dans le même appel renverraient une erreur.

Un exemple de requête SOAP/XML pour cette méthode est présenté ci-dessous:

**Requête d'échantillon SaveAccount**

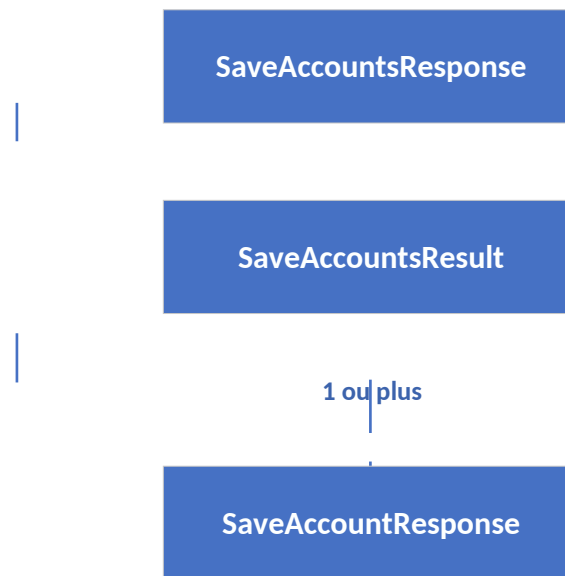
```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Request"
>
  <soapenv:Header/>
  <soapenv:Body>
    <tem:SaveAccounts>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:accounts>
        <!--One or more repetitions:-->
        <int:SaveAccountRequest>
          <int:DateOfBirth>1999-09-27T19:07:15.4993989+03:00</int:DateOfBirth>
          <int:Description>DESC_SPNTOSPIRG</int:Description>
          <int:FullName>First Last</int:FullName>
          <int:IsVerified>1</int:IsVerified>
          <int:IdentityDocumentType>NationalIdentityCard</int:IdentityDocumentType>
          <int:IdentityNumberIssuingCountry>CYP</int:IdentityNumberIssuingCountry>
          <int:IdentityNumber>00112233</int:IdentityNumber>
          <int:RegisteredOnDate>2018-09-27T19:07:15.5004083+03:00</int:RegisteredOnDate>
          <int>Status>Active</int>Status>
          <int:Username>UT-User_XBBQARPETW</int:Username>
        </int:SaveAccountRequest>
      </tem:accounts>
    </tem:SaveAccounts>
  </soapenv:Body>
</soapenv:Envelope>

```

### 3.1.1.3 Méthode Response

À la suite de l'appel à n'importe quelle méthode du BMRS, le système fournit un retour d'information à l'expéditeur sous la forme d'une réponse XML. Dans le cas de la création ou de la mise à jour des comptes des joueurs, cette réponse inclut des retours d'information pour chacun des comptes de l'appel initial, recueillis dans le cadre de l'élément de base *SaveAccountsResult* comme indiqué dans la figure 3.2. Noter que chaque compte est référencé par son identifiant assigné.



Élément	Type de données
ErrorCode	Nombre entier
ErrorMessage	Chaîne de caractères
Success	Boolean
Username	Chaîne de caractères

**Figure 3.2:** Structure XML de *SaveAccountsResponse*.

Les types d'erreurs possibles et leur description sont énumérés dans le tableau 3.4.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber: '{0}' LicenseeIdentifier: '{1}' DataEntryKey: '{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, un code pays non valide, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues)
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, BetSlips, BetSlipItems, BetLines, Accounts, AccountTransactions, Restrictions)
1515	«Le titulaire de licence {0} n'accepte pas les comptes. Type de titulaire de licence: {1}»	Aucune requête relative à un compte n'est autorisée à moins que le titulaire ne soit de classe A ou B.

**Tableau 3.4:** Types d'erreurs dans SaveAccountsResponse.

Voici un exemple de réponse SOAP/XML:

Exemple de réponse SaveAccount:
<p><b>Exemple de réponse:</b></p> <pre> &lt;s:Envelope xmlns:s="http"&gt; &lt;s:Body&gt; &lt;SaveAccountsResponse xmlns="htt"&gt; &lt;SaveAccountsResult xmlns:a=" nse" xmlns:i="htt"&gt; &lt;a:SaveAccountResponse&gt; &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt; &lt;a:ErrorMessage i:nil="true"/&gt; &lt;a:Success&gt;true&lt;/a:Success&gt; &lt;a:Username&gt;UT-User_XBBQARPETW&lt;/a:Username&gt; &lt;/a:SaveAccountResponse&gt; &lt;/SaveAccountsResult&gt; &lt;/SaveAccountsResponse&gt; &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

## 3.2 Transactions de compte

### 3.2.1 Méthode CreateAccountTransactions

#### 3.2.1.1 Description

**Remarque:** Les transactions de compte s'appliquent uniquement aux titulaires de licence de classe B. Aucune donnée de ce genre n'est attendue d'un bookmaker de classe A.

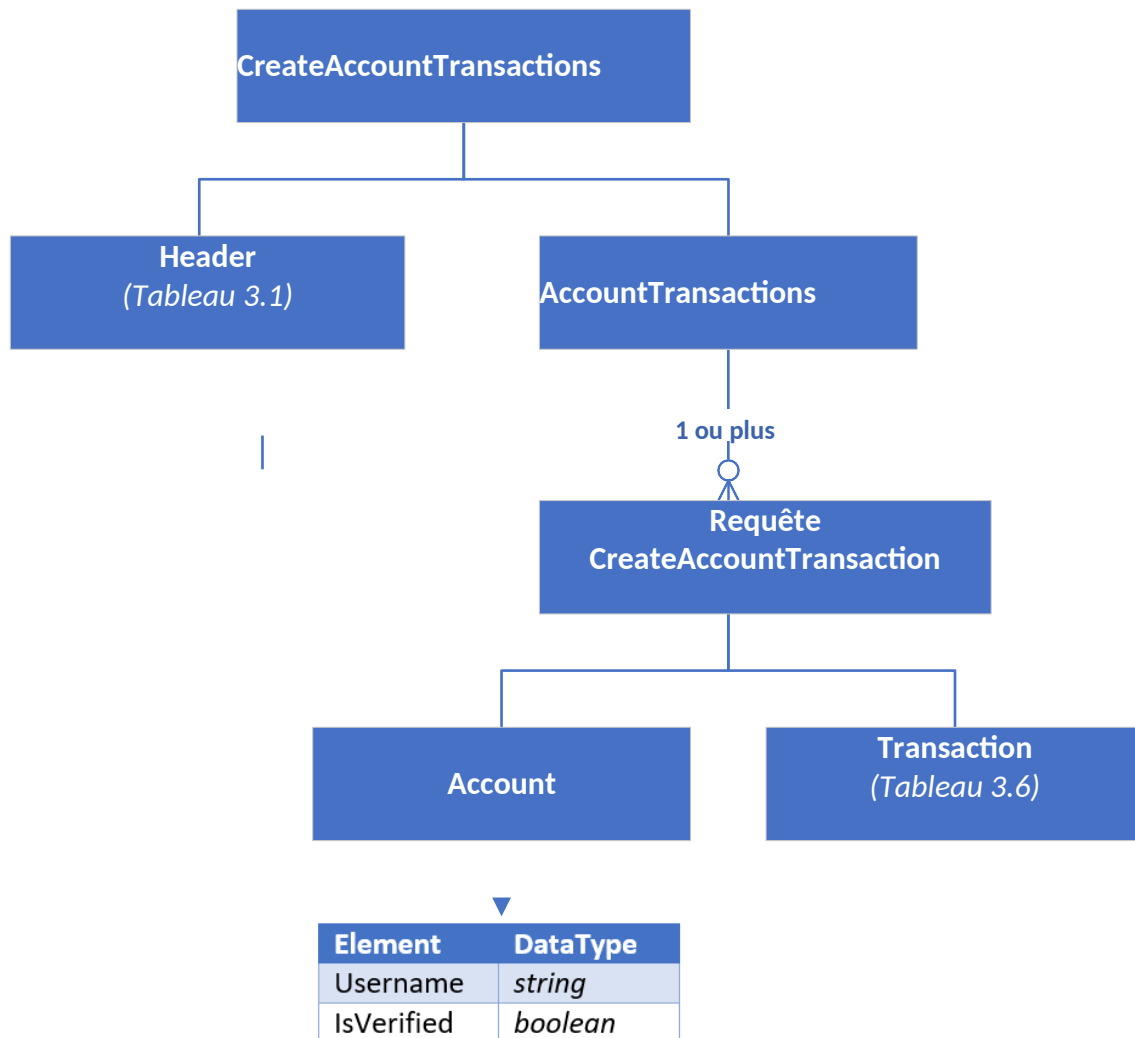
Les transactions des joueurs reflètent tout transfert de fonds dans ou hors du compte d'un joueur, l'engagement de fonds dans les paris et toute conversion de bonus en argent réel. La liste complète des types de transactions est illustrée dans le tableau 3.5 ci-dessous. Avant qu'une transaction de quelque nature que ce soit ne soit démarrée dans le BMRS, le compte en question doit avoir été créé. Une fois envoyée, une transaction de compte ne peut pas être mise à jour.

Type de transaction	Description	Champs obligatoires
Deposit	Des fonds ont été déposés dans le compte.	AmountMoney
Withdraw	Les fonds ont été retirés du compte.	AmountMoney
BonusReal	Un montant de bonus a été converti en argent réel.	AmountBonus ET AmountMoney
BetWinnings	Un montant (argent réel ou bonus ou les deux) a été crédité sur le compte à la suite d'activités de paris.	AmountBonus ET/OU AmountMoney
BetWager	Un montant (argent réel ou bonus ou les deux) a été engagé pour effectuer un pari.	AmountBonus ET/OU AmountMoney
AddBonus	Un montant de bonus a été crédité sur le compte.	AmountBonus
RemoveBonus	Un montant de bonus a été retiré du compte.	AmountBonus
Refund	Un montant (argent réel ou bonus ou les deux) a été remboursé sur le compte.	AmountBonus ET/OU AmountMoney

**Tableau 3.5:** CreateAccountTransactions — valeurs du champ **TransactionType**.

### 3.2.1.2 Méthode Request

Toute transaction des types énumérés dans le tableau 3.5 qui a eu lieu sur la plateforme du bookmaker doit être communiquée à l'Autorité dans une fenêtre de 90 secondes. La méthode *CreateAccountTransactions* facilite la transmission de ces données par lots, où chaque lot peut contenir plusieurs transactions de ce type (chacune avec un *TransactionReferenceNumber* différent) à partir de plusieurs comptes de joueurs. La structure de données pertinente est illustrée à la figure 3.3. Les éléments de la transaction sont décrits dans le tableau 3.6.



**Figure 3.3:** La structure XML de la méthode *CreateAccountTransactions*.

Transaction				
N°.	Nom du champ	Type de données	Description	Commentaires
1	TransactionReferenceNumber	Chaîne de caractères	Un identifiant unique pour cette transaction.	Il est attribué par le système du bookmaker.
2	TransactionType	Énumération [Deposit] [Withdraw] [BonusReal] [BetWinnings] [BetWager] [AddBonus] [RemoveBonus]  [Refund]	Le type de transaction.	Voir le tableau 3.6 pour plus de détails.
3	AmountMoney	Nombre décimal	Le montant de l'argent réel impliqué dans cette transaction.	
4	AmountBonus	Nombre décimal	Le montant du bonus impliqué dans cette transaction.	
5	BalanceMoney	Nombre décimal	Le solde du compte résultant de l'argent réel <u>après</u> cette transaction.	Cela exclut tout argent réel actuellement misé (champ 7).
6	BalanceBonus	Nombre décimal	Le solde du compte résultant du bonus <u>après</u> cette transaction.	Cela exclut tout bonus actuellement misé (champ 8).
7	WageredMoney	Nombre décimal	Le montant total de l'argent réel de ce compte qui est misé sur les paris en cours <u>après</u> cette transaction.	
8	WageredBonus	Nombre décimal	Le montant total du bonus de ce compte qui est misé sur les paris en cours <u>après</u> cette transaction.	
9	CreatedOnDate	Date et heure	La date et l'heure UTC de cette transaction ont été créées dans le système du bookmaker.	

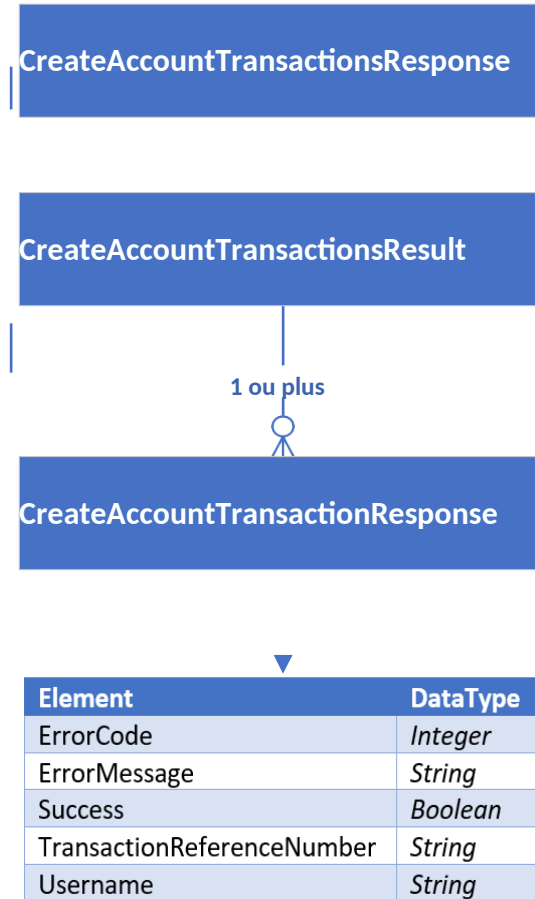
**Tableau 3.6:** Champs de données (sous-éléments) sous l'élément Transaction.

Un exemple de requête SOAP/XML pour cette méthode est présenté ci-dessous:

Requête d'échantillon CreateAccountTransactions
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/" xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ est"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;tem:CreateAccountTransactions&gt;       &lt;tem:header&gt;         &lt;int:DataEntryKey&gt;F2D64583-DFC1-4DF1-A95D-02BCD1CBA489&lt;/int:DataEntryKey&gt;         &lt;int:LicenseNumber&gt;A-UT-0001&lt;/int:LicenseNumber&gt;         &lt;int:LicenseeIdentifier&gt;19666639-C7F8-4942-876D-CA1088B75EFE&lt;/int:LicenseeIdentifier&gt;       &lt;/tem:header&gt;       &lt;tem:accountTransactions&gt;         &lt;!-- One or more repetitions:--&gt;         &lt;int:CreateAccountTransactionRequest&gt;           &lt;int:Account&gt;             &lt;int:IsVerified&gt;1&lt;/int:IsVerified&gt;             &lt;int:Username&gt;UT-User_QVNXIXOLGG&lt;/int:Username&gt;           &lt;/int:Account&gt;           &lt;int:Transaction&gt;             &lt;int:AmountBonus&gt;77&lt;/int:AmountBonus&gt;             &lt;int:AmountMoney&gt;2319&lt;/int:AmountMoney&gt;             &lt;int:BalanceBonus&gt;263&lt;/int:BalanceBonus&gt;             &lt;int:BalanceMoney&gt;4083&lt;/int:BalanceMoney&gt;             &lt;int:CreatedOnDate&gt;2019-09-27T19:11:54.3972355+03:00&lt;/int:CreatedOnDate&gt;             &lt;int:TransactionReferenceNumber&gt;TR-UT_KGLDDUNFMR_122312313&lt;/int:TransactionReferenceNumber&gt;             &lt;int:TransactionType&gt;BetWinnings&lt;/int:TransactionType&gt;             &lt;int:WageredBonus&gt;60&lt;/int:WageredBonus&gt;             &lt;int:WageredMoney&gt;53&lt;/int:WageredMoney&gt;           &lt;/int:Transaction&gt;         &lt;/int:CreateAccountTransactionRequest&gt;       &lt;/tem:accountTransactions&gt;     &lt;/tem:CreateAccountTransactions&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

### 3.2.1.3 Méthode Response

Le BMRS répond à un appel *CreateAccountTransactions* avec la structure XML illustrée à la figure 3.4. La réponse fournit un retour d'information sur chaque transaction reçue par l'intermédiaire de l'API, au moyen de leur champ *TransactionReferenceNumber*. Les types d'erreurs qui pourraient être renvoyés sont décrits dans le tableau 3.7.



**Figure 3.4:** La structure XML de CreateAccountTransactionsResponse.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber:'{0}' LicenseeIdentifier:'{1}' DataEntryKey:'{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, un code de pays non valide, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues)
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, BetSlips, BetSlipItems, BetLines, Accounts, AccountTransactions, Restrictions)
1515	«Le titulaire de licence {0} n'accepte pas les comptes. Type de titulaire de licence: {1}»	Aucune requête relative à un compte n'est autorisée à moins que le titulaire ne soit de classe A ou B.

**Tableau 3.7:** Types d'erreurs dans CreateAccountTransactionsResponse.

Voici un exemple de réponse SOAP/XML retournée par cette méthode:

Exemple de réponse CreateAccountTransactions
<pre> &lt;s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;s:Body&gt;     &lt;CreateAccountTransactionsResponse xmlns="http://tempuri.org/"&gt;       &lt;CreateAccountTransactionsResult xmlns:a="http://schemas.datacontract.org/2004/07/IntelIScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;         &lt;a:CreateAccountTransactionResponse&gt;           &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt;           &lt;a:ErrorMessage i:nil="true"/&gt;           &lt;a:Success&gt;true&lt;/a:Success&gt;           &lt;a:TransactionReferenceNumber&gt;TR-UT_KGLDDUNFMR_122312313&lt;/a:TransactionReferenceNumber&gt;           &lt;a:Username&gt;UT-User_QVNXIXOLGG&lt;/a:Username&gt;         &lt;/a:CreateAccountTransactionResponse&gt;       &lt;/CreateAccountTransactionsResult&gt;     &lt;/CreateAccountTransactionsResponse&gt;   &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

### 3.3 Restrictions de compte

**Remarque:** Les restrictions de compte s'appliquent uniquement aux titulaires de licence de classe B. Aucune donnée de ce genre n'est attendue d'un bookmaker de classe A.

Les restrictions de compte imposent des limites à certaines transactions ou activités de paris autorisées par le compte d'un joueur et ces limites peuvent être auto-imposées ou appliquées par le bookmaker. L'activation, la désactivation et, dans le cas où il y a une dimension temporelle impliquée, l'expiration d'une restriction sont les trois événements surveillés par le BMRS, et si l'un d'entre eux se produit à un moment donné, un appel à la fonction *SaveAccountRestrictions* doit être déclenché dans les 90 secondes qui suivent. Les types de restrictions disponibles dans le BMRS sont décrits dans le tableau 3.8.

Type de restriction	Description
ShortTermSelfExclusion	Le joueur s'est exclu de tout pari pendant une certaine période.
LongTermSelfExclusion	Le joueur s'est exclu de tout pari pour une durée indéterminée.
BettingAmountLimit	Une limite est fixée concernant le montant total autorisé pour les activités de paris sur une certaine période par ce compte.
BettingLossLimit	Une limite est fixée concernant le montant total que le compte peut perdre au cours d'une certaine période.
DepositLimit	Une limite est fixée concernant le montant d'argent qui peut être déposé sur ce compte sur une certaine période.
Other	Tout autre type de restriction non couvert par ce qui précède. Des détails peuvent être fournis dans le champ Description (voir tableau 3.9).

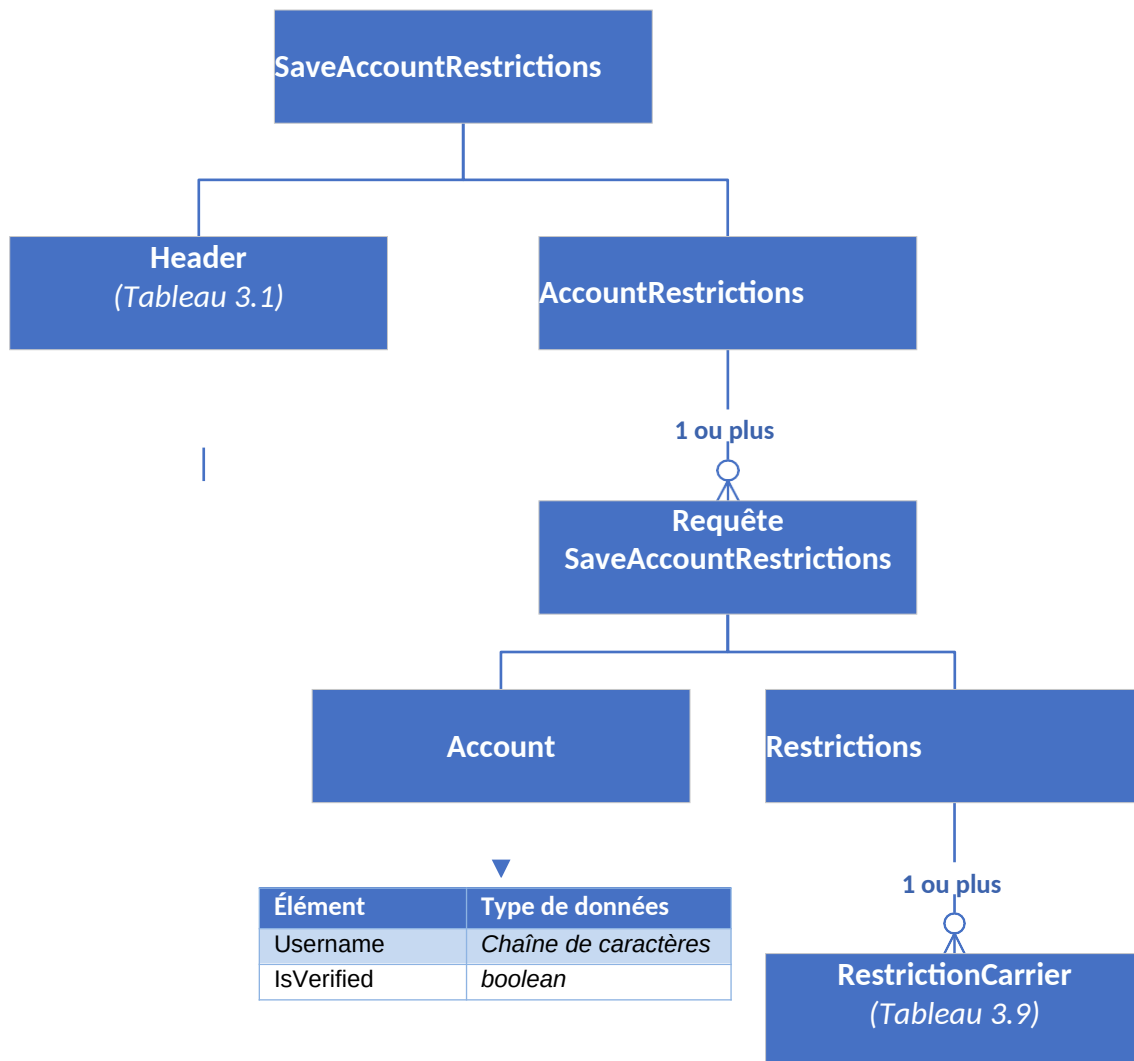
**Tableau 3.8:** *SaveAccountRestrictions* — valeurs du champ **RestrictionType**.

### 3.3.1 Méthode SaveAccountRestrictions

#### 3.3.1.1 Description

La structure XML adoptée par cette méthode est illustrée à la figure 3.5. Les éléments de données uniques figurant dans le nœud *RestrictionCarrier* de l'arborescence sont décrits séparément dans le tableau 3.9. Comme pour toutes les méthodes du BMRS, un seul appel à *SaveAccountRestrictions* peut contenir des lots de données, c'est-à-dire dans ce cas plusieurs événements liés aux restrictions pour un ou plusieurs comptes.

#### 3.3.1.2 Méthode Request



**Figure 3.5:** La structure XML de la méthode *SaveAccountRestrictions*.

RestrictionCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	EventName	Énumération [Activated] [Deactivated] [Expired]	L'événement par lequel cette restriction a été déclenchée.	
2	Type de restriction	Énumération [ShortTermSelfExclusion] [LongTermSelfExclusion] [BettingAmountLimit] [BettingLossLimit] [DepositLimit] [Other]	Le type de restriction.	Voir le tableau 3.8 pour plus de détails.
3	CreatedBy	Énumération [Client] [Bookmaker]	Indique si cette restriction a été imposée par le joueur ou le bookmaker.	
4	CreatedOnDate	Date et heure	Date et heure UTC à laquelle l'événement déclencheur (champ 1) s'est produit.	
5	Rationale	Chaîne de caractères	La raison derrière cette restriction.	Principalement dans les cas où la restriction est imposée par le bookmaker.
6	Description	Chaîne de caractères	Toute information complémentaire.	Par exemple, si RestrictionType est Other, ce champ peut contenir une brève description de ce qui est limité par cette restriction.

**Tableau 3.9:** Champs de données (sous-éléments) sous l'élément RestrictionCarrier.

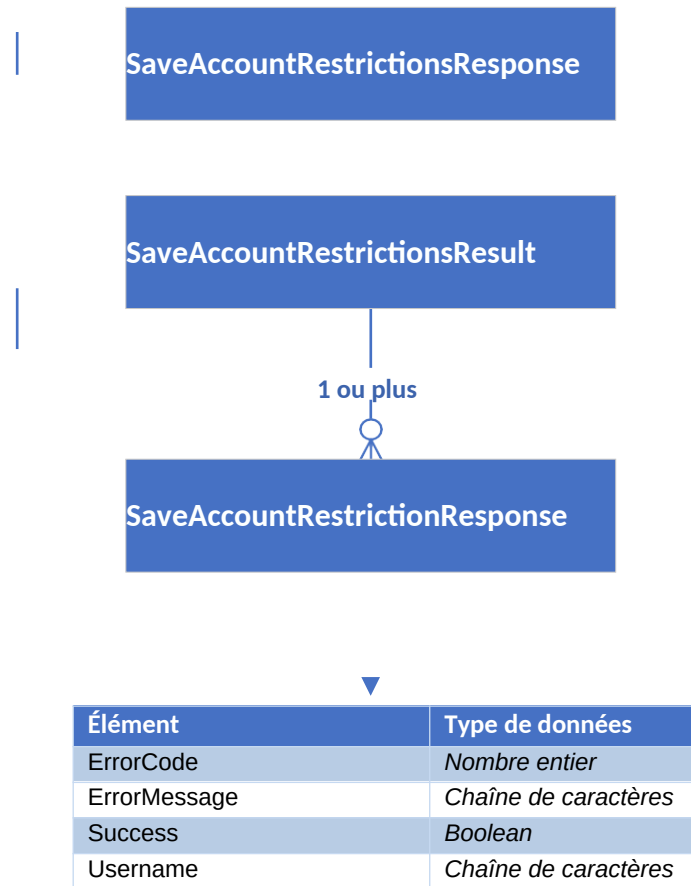
Voici un exemple de cette méthode de requête SOAP/XML:

**Requête d'échantillon SaveAccountRestrictions**

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Req
uest">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:SaveAccountRestrictions>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-JT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:accountRestrictions>
        <!--One or more repetitions:-->
        <int:SaveAccountRestrictionRequest>
          <int:Account>
            <int:IsVerified>true</int:IsVerified>
            <int:Username>UT-User_CFMHOFYMVN</int:Username>
          </int:Account>
          <int:Restrictions>
            <!--One or more repetitions:-->
            <int:RestrictionCarrier>
              <int:CreatedBy>Bookmaker</int:CreatedBy>
              <int:CreatedOnDate>2019-09-27T18:59:56.3519795+03:00</int:CreatedOnDate>
              <int:Description>Description</int:Description>
              <int:EventName>Activated</int:EventName>
              <int:Rationale>Rationale</int:Rationale>
              <int:RestrictionType>BettingLossLimit</int:RestrictionType>
            </int:RestrictionCarrier>
          </int:Restrictions>
        </int:SaveAccountRestrictionRequest>
      </tem:accountRestrictions>
    </tem:SaveAccountRestrictions>
  </soapenv:Body>
</soapenv:Envelope>
```

### 3.3.1.3 Méthode Response

Le BMRS répond à un appel *SaveAccountRestrictions* avec la structure XML illustrée à la figure 3.6. La réponse fournit des commentaires sur chaque événement de restriction reçu via l'API, et ces restrictions sont vaguement référencées par l'identifiant du compte concerné. Les types d'erreurs qui pourraient être renvoyés sont décrits dans le tableau 3.10.



**Figure 3.6:** La structure XML de *SaveAccountRestrictionsResponse*.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber:'{0}' LicenseeIdentifier:'{1}' DataEntryKey:'{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, un code de pays non valide, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues)
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, BetSlips, BetSlipItems, BetLines, Accounts, AccountTransactions, Restrictions)
1515	«Le titulaire de licence {0} n'accepte pas les comptes. Type de titulaire de licence: {1}»	Aucune requête relative à un compte n'est autorisée à moins que le titulaire ne soit de classe A ou B.

**Tableau 3.10:** Types d'erreurs dans SaveAccountRestrictionsResponse.

Voici un exemple de réponse SOAP/XML pour cette méthode:

Exemple de réponse SaveAccountRestrictions Réponse de l'échantillon:
<pre> &lt;s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;s:Body&gt;     &lt;SaveAccountRestrictionsResponse xmlns="http://tempuri.org/"&gt;       &lt;SaveAccountRestrictionsResult xmlns:a="http://schemas.datacontract.org/2004/07/Inteliscape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;         &lt;a:SaveAccountRestrictionResponse&gt;           &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt;           &lt;a:ErrorMessage i:nil="true"/&gt;           &lt;a:Success&gt;true&lt;/a:Success&gt;           &lt;a:Username&gt;UT-User_CFMHOFYMVN&lt;/a:Username&gt;         &lt;/a:SaveAccountRestrictionResponse&gt;       &lt;/SaveAccountRestrictionsResult&gt;     &lt;/SaveAccountRestrictionsResponse&gt;   &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

### 3.4 **Activité de pari**

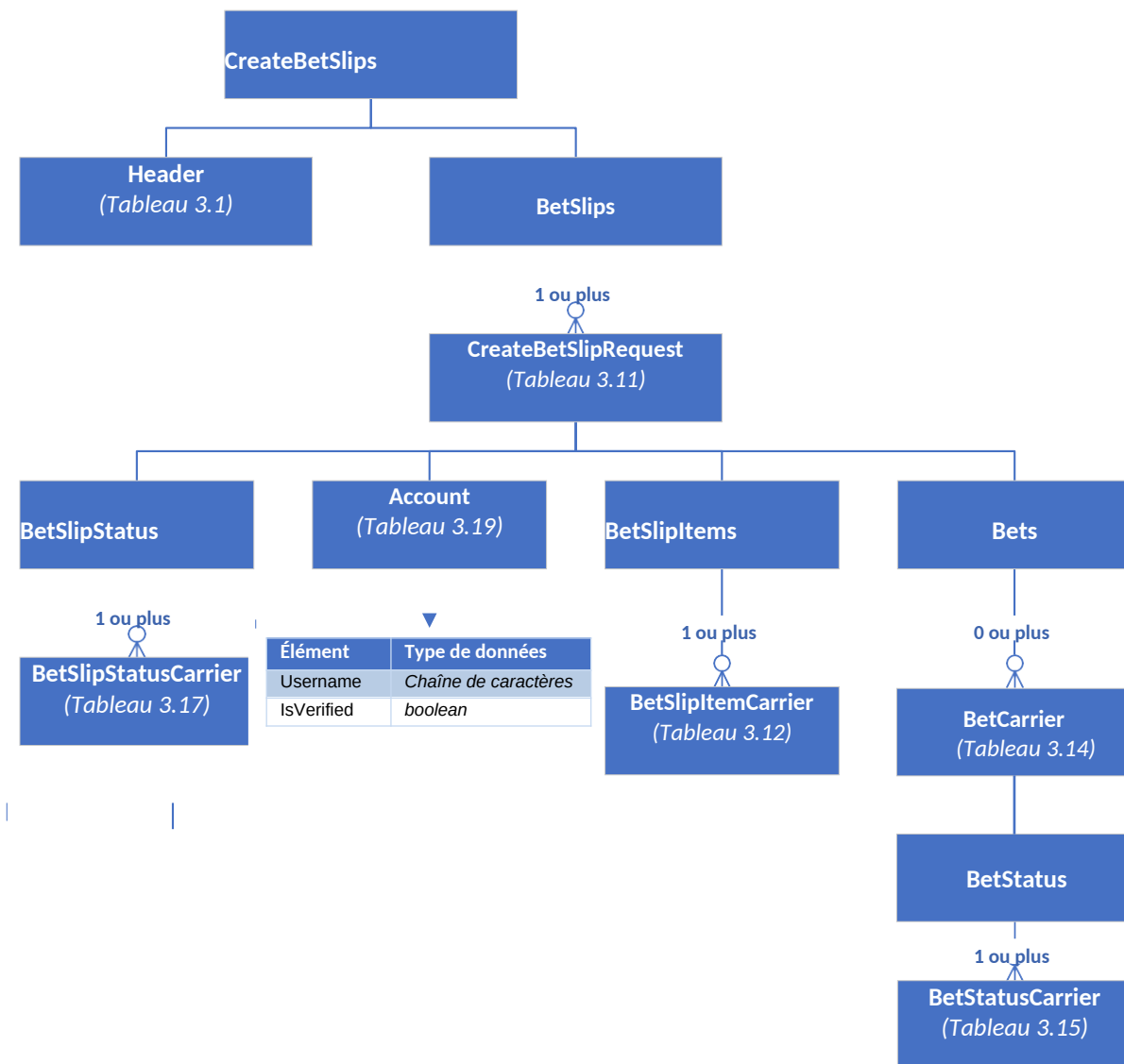
L'activité de pari dans le contexte du BMRS comprend le placement des paris par les comptes des joueurs, une description détaillée de ces paris et le suivi de leur cycle de vie à travers leur statut. Au cœur du modèle de données utilisé pour recevoir les données de pari est l'objet *BetSlip*, qui contient une variété d'informations, y compris les résultats multiples sur lesquels un joueur souhaite parier et les *Bets* réels sur ces résultats. Une vue d'ensemble de la relation entre ces objets est fournie à l'article 2.2.

### 3.4.1 Méthode CreateBetSlips

#### 3.4.1.1 Description

Les nouveaux bulletins de pari créés par un ou plusieurs comptes peuvent être communiqués au BMRS par lots en appelant la méthode *CreateBetSlips*. La structure XML pertinente est illustrée à la figure 3.7. Les différents éléments de données et les valeurs qu'ils peuvent prendre sont décrits dans les tableaux 3.11 à 3.19.

#### 3.4.1.2 Méthode Request



**Figure 3.7:** La structure XML de la méthode *CreateBetSlips*.

CreateBetSlipRequest				
N°.	Nom du champ	Type de données	Description	Commentaires
1	ReferenceNumber	Chaîne de caractères	Identifiant unique attribué par le système du bookmaker. Aucun bulletin de pari ne peut avoir la même valeur.	Identifiant unique. Utilisé pour référencer le bulletin de pari dans les requêtes de mise à jour ultérieures.
2	IssuerLicenseNumber	Chaîne de caractères	Le numéro de licence de l'émetteur.	<u>Classe A</u> : Licence de magasin <u>Classe B</u> : Licence de bookmaker
3	InitialStake	Nombre décimal	La mise initiale totale sur ce bulletin de pari.	<u>Inclut</u> le bonus.
4	InitialStakeBonus	Nombre décimal	Tout montant de bonus sur la mise initiale.	
5	Payout	Nombre décimal	Le paiement final total de ce bulletin de pari après le règlement de tous les paris.	<u>Inclut</u> le bonus.
6	PayoutBonus	Nombre décimal	Tout montant de bonus dans le paiement final total de ce bulletin de pari.	
7	MaxPayout	Nombre décimal	Le paiement total maximum possible.	
8	MinPayout	Nombre décimal	Le paiement total minimum possible.	
9	Description	Chaîne de caractères	Toute information complémentaire.	
10	Commission	Nombre décimal	Montant absolu de la commission versée au représentant autorisé.	
11	TerminalId	Chaîne de caractères	Identifiant du terminal utilisé pour créer ce bulletin de pari.	Classe A seulement.
12	TotalNumberOfCombinations	Nombre entier	Nombre total de combinaisons résultant des paris sous-jacents.	
13	CreatedOnDate	Date et heure	Date et heure de la création du bulletin de pari dans le système du bookmaker.	

**Tableau 3.11:** Champs de données (sous-éléments) sous l'élément CreateBetSlipRequest.

BetSlipltemCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	ItemReferenceNumber	Chaîne de caractères	Numéro de référence attribué à cet élément du bulletin de pari par le système du bookmaker.	Identifiant unique dans ce bulletin de pari uniquement.
2	BetType	Énumération [Outright] [Pregame] [Live] [Other]	Le type de pari de cet élément du bulletin de pari.	
3	Sport	Chaîne de caractères	Le nom du sport de l'événement sous-jacent.	
4	Competition	Chaîne de caractères	Le nom de la compétition au sein de laquelle l'événement sous-jacent a lieu.	
5	Region	Chaîne de caractères	La région géographique dans laquelle se déroule la compétition.	
6	EventName	Chaîne de caractères	Le nom de l'événement sous-jacent.	
7	EventStartDate	Date et heure	La date et l'heure UTC à laquelle l'événement sous-jacent commence.	
8	EventResult	Chaîne de caractères	Le résultat final de l'événement sous-jacent.	
9	EventKey	Chaîne de caractères	L'identifiant unique attribué à l'événement sous-jacent par le système du bookmaker.	
10	CompetitorA	Chaîne de caractères	Le nom du concurrent A.	Si applicable.
11	CompetitorB	Chaîne de caractères	Le nom du concurrent B.	Si applicable.
12	MarketType	Chaîne de caractères	Le nom du type de marché.	Exemples: «Résultat à temps plein» «Under/Over 2.5»
13	MarketTypeKey	Chaîne de caractères	L'identifiant unique attribué à ce type de marché par le système du bookmaker.	
14	MarketTypeKeyBMRS	Chaîne de caractères	L'identifiant unique attribué à ce type de marché par le BMRS.	Fourni par la NBA.
15	SelectionName	Chaîne de caractères	Le nom de la sélection du joueur.	Exemples: «Home To Win» «Over»

16	SelectionKeyBMRS	Chaîne de caractères	L'identifiant unique attribué à la sélection du joueur par le BMRS.	Fourni par la NBA.
17	Odds	Nombre décimal	Les cotes offertes par le bookmaker pour la sélection du joueur.	
18	Description	Chaîne de caractères	Toute information complémentaire.	
19	Status	Énumération: [Pending] [Won] [Lost] [Void]	Le statut de la sélection du joueur pour le marché de cet élément.	Voir tableau 3.13.

**Tableau 3.12:** Champs de données (sous-éléments) sous l'élément *BetSliptemCarrier*.

Status	Description
Pending	Le résultat du marché n'a pas encore été décidé.
Won	Le résultat du marché est le même que la sélection du joueur.
Lost	Le résultat du marché est différent de la sélection du joueur.
Void	Le résultat du marché ne peut pas être déterminé (par exemple, un événement annulé).

**Tableau 3.13:** *BetSliptemCarrier* — valeurs du champ **Status**.

BetCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	BetReferenceNumber	Chaîne de caractères	Numéro de référence de ce pari. Attribué par le système du bookmaker.	Identifiant unique dans ce bulletin de pari uniquement.
2	BetSliptemReferenceNumbers	Chaîne de caractères	Chaîne de caractères séparée par des virgules du numéro de référence (tableau 3.12, champ 1) de chaque élément du bulletin de pari inclus dans ce pari.	
3	Description	Chaîne de caractères	Toute information complémentaire.	
4	NumberOfCombinations	Nombre entier	Le nombre total de combinaisons dérivées de ce pari.	

5	MinOdds	Nombre décimal	Cotes minimales offertes pour ce pari.	
6	MaxOdds	Nombre décimal	Cotes maximales offertes pour ce pari.	
7	InitialStake	Nombre décimal	La mise initiale totale sur ce pari. Inclut le bonus.	
8	Payout	Nombre décimal	Le paiement final total de ce pari. Inclut le bonus.	

**Tableau 3.14:** Champs de données (sous-éléments) sous l'élément BetCarrier.

BetStatusCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Status	Énumération [Pending] [Won] [Lost] [Void]	Le statut actuel de ce pari.	Voir tableau 3.16.
2	CreatedOnDate	Date et heure	Date et heure UTC à laquelle ce statut a été atteint dans le système du bookmaker.	

**Tableau 3.15:** Champs de données (sous-éléments) sous l'élément BetStatusCarrier.

Status	Description
Pending	Le pari est toujours en attente.
Won	Au moins une des combinaisons sous-jacentes du pari a été réalisée.
Lost	Aucune des combinaisons sous-jacentes du pari n'a été réalisée.
Void	Le pari n'est plus considéré comme valide.

**Tableau 3.16:** BetStatusCarrier — valeurs du champ Status.

BetSlipStatusCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Status	Énumération [Submitted] [Accepted] [Cancelled] [Rejected] [Won] [Lost] [Refund] [PaidWon] [PaidRefund]	Le statut actuel de ce bulletin de pari.	Voir la figure 2.2 et le tableau 3.18 pour le cycle de vie du bulletin de pari.
2	CurrentPayout	Nombre décimal	Paiement total déjà sécurisé par ce bulletin de pari. Inclut le bonus.	
3	SettledStake	Nombre décimal	Tout montant de mise qui a été réglé. Inclut le bonus.	
4	CreatedOnDate	Date et heure	La date et l'heure UTC lorsque ce changement dans Status ou CurrentPayout ou SettledStake a été enregistré dans le système du bookmaker.	

**Tableau 3.17:** Champs de données (sous-éléments) sous l'élément BetSlipStatusCarrier.

Status	Description
Submitted	Le bulletin de pari a été créé dans le système du bookmaker.
Accepted	Le bookmaker a accepté le bulletin de pari.
Cancelled	Le joueur a annulé le bulletin de pari.
Rejected	Le bookmaker a rejeté le bulletin de pari.
Won	Un ou plusieurs paris sur le bulletin de pari sont gagnants. Le montant n'a pas encore été versé au joueur.
Lost	Aucun des paris sur le bulletin de pari n'est gagnant.
Refund	La mise sera remboursée au joueur.
PaidWon	Les gains ont été versés au joueur.
PaidRefund	Le remboursement a été versé au joueur.

**Tableau 3.18:** BetSlipStatusCarrier — valeurs du champ Status.

**Remarque:** Chaque bulletin de pari créé en transmettant des données à la méthode *CreateBetSlips* peut avoir plusieurs éléments *BetSlipStatusCarrier* et/ou *BetStatusCarrier* qui y sont attachés (figure 3.7). Cette fonctionnalité permet d'inclure plus d'un changement dans les paramètres sous-jacents de ces nœuds, dans le même appel. Par exemple, si un bulletin de pari nouvellement créé a été envoyé et accepté dans la même fenêtre de 90 secondes, l'expéditeur peut choisir de transmettre les deux changements dans un seul appel à cette méthode en incluant deux éléments distincts et convenablement horodatés *BetSlipStatusCarrier* à sa structure XML.

Account				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Username	Chaîne de caractères	Identifiant (UID) du compte qui a placé ce pari.	Pour la classe A, il s'agira d'une caisse.
2	IsVerified	Boolean	Indiquer si le compte qui a placé ce bulletin de pari est un compte vérifié ou non.	

**Tableau 3.19:** Champs de données (sous-éléments) sous l'élément Compte.

Voici un exemple de structure de requête SOAP/XML pour cette méthode:

```

Requête d'échantillon de CreateBetSlips:

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ
est">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:CreateBetSlips>
      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>
      <tem:betSlips>
        <!--One or more repetitions:-->
        <int:CreateBetSlipRequest>

          <int:BetSlipStatus>
            <!--One or more repetitions:-->
            <BetSlipStatusCarrier>
              <int:CreatedOnDate>2019-09-27T19:19:13.6183845+03:00</int:CreatedOnDate>
              <int:CurrentPayout>412</int:CurrentPayout>
              <int:SettledStake>26</int:SettledStake>
              <int:Status>Submitted</int:Status>
            </BetSlipStatusCarrier>
          </int:BetSlipStatus>
        </tem:betSlips>
      </tem:CreateBetSlips>
    </soapenv:Body>
  </soapenv:Envelope>

```

```

<int:CreatedOnDate>2019-09-27T19:20:13.6183845+03:00</int:CreatedOnDate>
<int:CurrentPayout>412</int:CurrentPayout>
<int:SettledStake>26</int:SettledStake>
<int:Status>Accepted</int:Status>
</BetSlipStatusCarrier>
</int:BetSlipStatus>

<int:Commision>1</int:Commision>
<int:Description>Unit Test APSOHAUYUI</int:Description>
<int:Payout>55</int:Payout>
<int:PayoutBonus>55</int:PayoutBonus>
<int:ReferenceNumber>UT-38115dcca4ce4-1277-4c56-b48b-7ac46018123</int:ReferenceNumber>
<int:IssuerLicenseNumber>SHOP-UT-0002</int:IssuerLicenseNumber>

<int:Account>
  <int:IsVerified>true</int:IsVerified>
  <int:Username>UT-User332014</int:Username>
</int:Account>

<int:BetSlipItems>
  <!-- One or more repetitions:-->
  <int:CreateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.6193833+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>1</int:ItemReferenceNumber>
    <int:Status>Pending</int:Status>
    <int:BetType>Outright</int:BetType>
    <int:Competition>UEFA Champions League</int:Competition>
    <int:CompetitorA>Team 01</int:CompetitorA>
    <int:CompetitorB>Team 02</int:CompetitorB>
    <int:Description>Unit Test PTEQMSOCPU</int:Description>
    <int:EventKey>1</int:EventKey>
    <int:EventName>Team 01 Vs Team 15</int:EventName>
    <int:MarketType>Correct Score</int:MarketType>
    <int:MarketTypeKey>Correct Score</int:MarketTypeKey>
    <int:MarketTypeKeyBMRS>Correct Score</int:MarketTypeKeyBMRS>
    <int:Odds>8</int:Odds>
    <int:Region>Europe</int:Region>
    <int:SelectionKeyBMRS>1</int:SelectionKeyBMRS>
    <int:SelectionName>1</int:SelectionName>
    <int:Sport>Soccer</int:Sport>
  </int:CreateBetSlipRequest.BetSlipItemCarrier>

  <int:CreateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.621383+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>2</int:ItemReferenceNumber>
    <int:Status>Pending</int:Status>
    <int:BetType>Outright</int:BetType>
    <int:Competition>UEFA Champions League</int:Competition>
    <int:CompetitorA>Team 01</int:CompetitorA>
    <int:CompetitorB>Team 02</int:CompetitorB>
    <int:Description>Unit Test YGPIJAEKNL</int:Description>
    <int:EventKey>1</int:EventKey>
    <int:EventName>Team 01 Vs Team 16</int:EventName>
    <int:MarketType>Correct Score</int:MarketType>
    <int:MarketTypeKey>Correct Score</int:MarketTypeKey>
    <int:MarketTypeKeyBMRS>Correct Score</int:MarketTypeKeyBMRS>
    <int:Odds>6</int:Odds>
    <int:Region>Europe</int:Region>
    <int:SelectionKeyBMRS>1</int:SelectionKeyBMRS>
    <int:SelectionName>1</int:SelectionName>
    <int:Sport>Soccer</int:Sport>
  </int:CreateBetSlipRequest.BetSlipItemCarrier>

</int:BetSlipItems>

<int:Bets>
  <!-- Zero or more repetitions:-->
  <int:CreateBetSlipRequest.BetCarrier>

```

```

<int:BetReferenceNumber>1</int:BetReferenceNumber>
<int:BetStatus>
  <!-- One or more repetitions:-->
  <int:BetStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:13.6223832+03:00</int:CreatedOnDate>
    <int:Status>Pending</int:Status>
  </int:BetStatusCarrier>
</int:BetStatus>
<int:Payout>100</int:Payout>
<int:BetSlipItemReferenceNumbers>1</int:BetSlipItemReferenceNumbers>
<int:Description>Unit Test GGVRFPBDS</int:Description>
<int:InitialStake>22</int:InitialStake>
<int:MaxOdds>2</int:MaxOdds>
<int:MinOdds>5</int:MinOdds>
<int:NumberOfCombinations>86</int:NumberOfCombinations>
</int:CreateBetSlipRequest.BetCarrier>

<int:CreateBetSlipRequest.BetCarrier>
  <int:BetReferenceNumber>2</int:BetReferenceNumber>
  <int:BetStatus>
    <!-- One or more repetitions:-->
    <int:BetStatusCarrier>
      <int:CreatedOnDate>2019-09-27T19:20:13.6233824+03:00</int:CreatedOnDate>
      <int:Status>Pending</int:Status>
    </int:BetStatusCarrier>
  </int:BetStatus>
  <int:Payout>100</int:Payout>
  <int:BetSlipItemReferenceNumbers>2</int:BetSlipItemReferenceNumbers>
  <int:Description>Unit Test ARMUBSNYNQ</int:Description>
  <int:InitialStake>67</int:InitialStake>
  <int:MaxOdds>1</int:MaxOdds>
  <int:MinOdds>9</int:MinOdds>
  <int:NumberOfCombinations>28</int:NumberOfCombinations>
</int:CreateBetSlipRequest.BetCarrier>

</int:Bets>

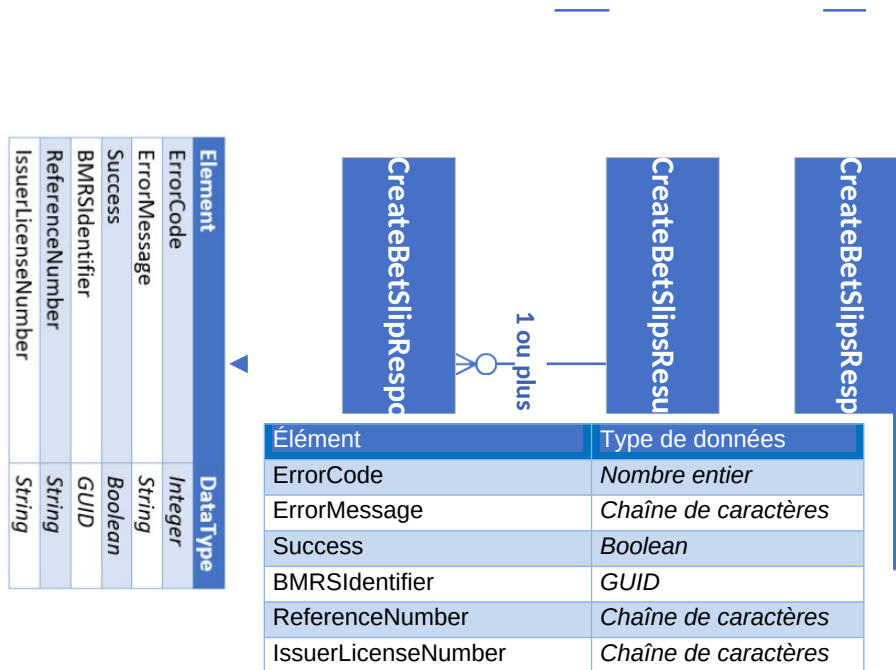
<int:CreatedOnDate>2019-09-27T19:20:13.6166518+03:00</int:CreatedOnDate>
<int:InitialStake>16</int:InitialStake>
<int:InitialStakeBonus>20</int:InitialStakeBonus>
<int:MaxPayout>100</int:MaxPayout>
<int:MinPayout>20</int:MinPayout>
<int:TerminalId>FLWVE</int:TerminalId>
<int:TotalNumberOfCombinations>2</int:TotalNumberOfCombinations>
</int:CreateBetSlipRequest>
</tem:betSlips>
</tem:CreateBetSlips>
</soapenv:Body>
</soapenv:Envelope>

```

### 3.4.2.3 Méthode Response

Quand la méthode *CreateBetSlips* est appelée, l'API répond avec la structure XML illustrée à la figure 3.8. La réponse fournit un retour d'information sur chaque bulletin de pari reçu, au moyen de son champ *ReferenceNumber*. Les types d'erreurs qui pourraient être renvoyés sont décrits dans le tableau 3.20.

Figure 3.8: La structure XML de *CreateBetSlipsResponse*.



ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber:'{0}' LicenseIdentifier:'{1}' DataEntryKey:'{2}'»	Les informations sur le titulaire transmises dans l'en-tête ne correspondent à aucun licencié dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.).
1501	«Représentant du titulaire de licence '{0}' introuvable»	Licence du représentant inconnue
1503	«Le titulaire de licence {0} ne représente pas le titulaire de licence {1}»	La licence du représentant ne représente pas le bookmaker.
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, un code de pays non valide, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues).
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, Betslips, BetSlipItems, Bets, Accounts, AccountTransactions, Restrictions).
1514	«Le titulaire de licence {0} n'accepte pas les paris. Type de titulaire de licence: {1}»	Les bulletins de paris sont autorisés uniquement pour la classe B ou les magasins. La classe A et les représentants doivent associer leurs bulletins de pari à un magasin.

**Tableau 3.20:** Types d'erreurs dans CreateBetSlipsResponse.

Voici un exemple de cette méthode de réponse SOAP/XML:

Exemple de réponse CreateBetSlips
<pre> &lt;s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;s:Body&gt;     &lt;CreateBetSlipsResponse xmlns="http://tempuri.org/"&gt;       &lt;CreateBetSlipsResult         xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo         nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;         &lt;a:CreateBetSlipResponse&gt;           &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt;           &lt;a:ErrorMessage i:nil="true"/&gt;           &lt;a:Success&gt;true&lt;/a:Success&gt;           &lt;a:BMRSIdentifier&gt;f7971e7c-09e5-47a6-a980-38822fe35597&lt;/a:BMRSIdentifier&gt;           &lt;a:ReferenceNumber&gt;UT-38115dcca4ce4-1277-4c56-b48b-7ac46018123&lt;/a:ReferenceNumber&gt;           &lt;a:IssuerLicenseNumber&gt;SHOP-UT-0002&lt;/a:IssuerLicenseNumber&gt;         &lt;/a:CreateBetSlipResponse&gt;       &lt;/CreateBetSlipsResult&gt;     &lt;/CreateBetSlipsResponse&gt;   &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

### 3.4.3 Méthode UpdateBetSlips

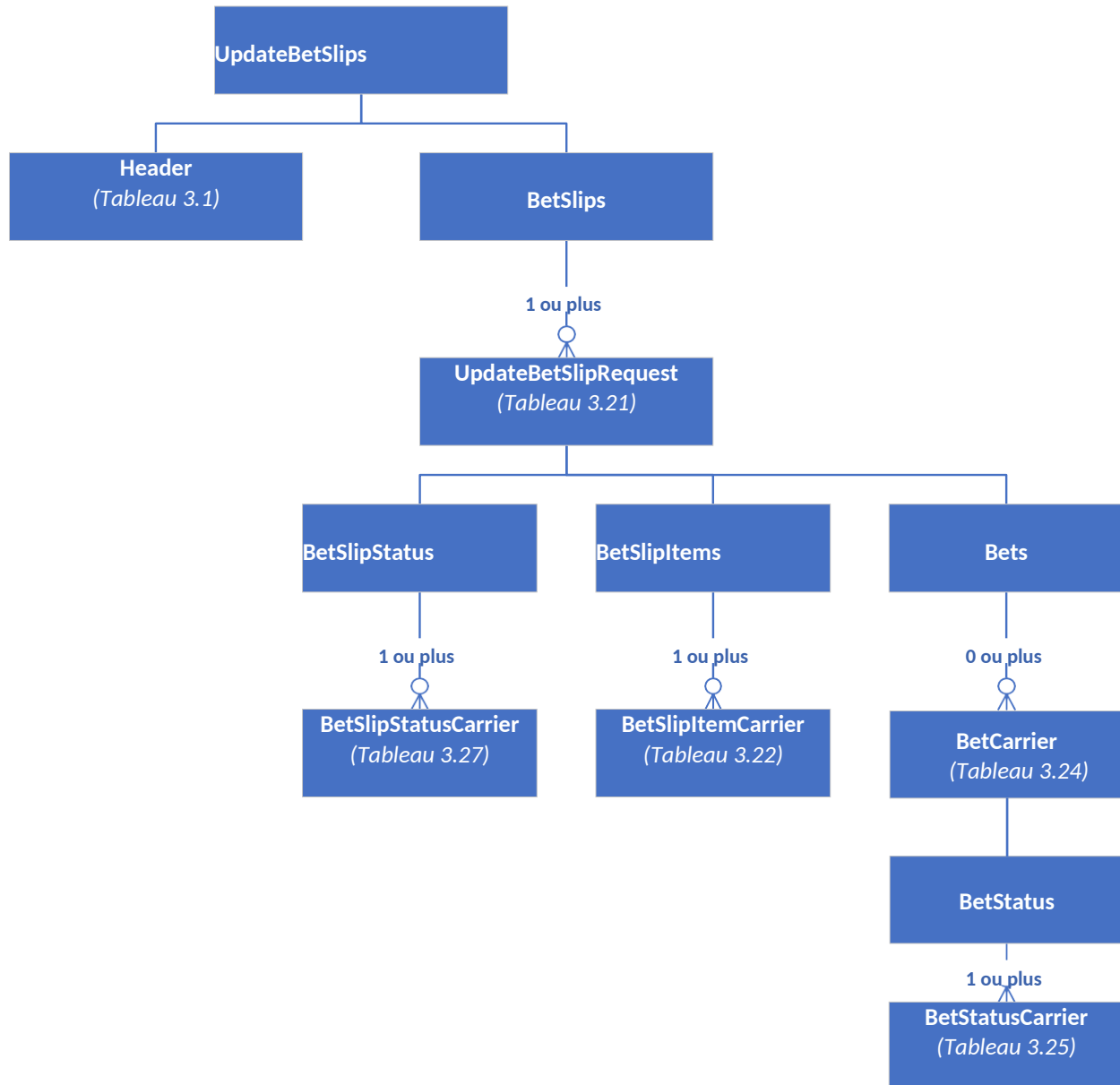
#### 3.4.3.1 Description

Une fois créés, les bulletins de pari suivent leur cycle de vie jusqu'au règlement, à l'issue duquel le joueur qui a créé un bulletin de pari a perdu ou reçu ses gains (ou un remboursement au cas où le bulletin deviendrait nul). Un changement dans le statut du bulletin de pari (tableau 3.18), la conclusion et le résultat d'un élément du bulletin de pari, un retrait partiel par un joueur ou le changement de statut d'un pari du bulletin, sont tous des exemples où le statut du bulletin de pari a été modifié.

L'API BMRS fournit la méthode *UpdateBetSlips* pour refléter de tels changements du statut d'un bulletin de pari. La méthode accepte les données utilisées pour référencer un bulletin de pari, ainsi que les données pour les champs qui nécessitent une mise à jour. Comme pour les fonctions du BMRS précédemment examinées, *UpdateBetSlips* peut recevoir ses données par lots, c'est-à-dire que plusieurs bulletins peuvent être mis à jour par l'intermédiaire d'un seul appel. Chaque lot ne peut contenir qu'une seule fois le *ReferenceNumber* d'un bulletin de pari.

### 3.4.3.2 Méthode Request

La structure XML de cette méthode est illustrée à la figure 3.9. Des éléments de données uniques sont décrits dans les tableaux 3.21 à 3.28.



**Figure 3.9:** La structure XML de la méthode `UpdateBetSlips`.

UpdateBetSlipRequest				
N°.	Nom du champ	Type de données	Description	Commentaires
1	ReferenceNumber	Chaîne de caractères	Identifiant unique attribué par le système du bookmaker.	Utilisé pour faire référence au bulletin qui fait l'objet de cette mise à jour.
2	IssuerLicenseNumber	Chaîne de caractères	Le numéro de licence de l'émetteur du bulletin de pari.	
3	Payout	Nombre décimal	Le paiement final total de ce bulletin de pari après le règlement de tous les paris.	Inclut le bonus.
4	PayoutBonus	Nombre décimal	Tout montant de bonus dans le paiement final total de ce bulletin de pari.	
5	Description	Chaîne de caractères	Toute information complémentaire.	
6	Commission	Nombre décimal	Montant absolu de la commission versée au représentant autorisé.	

**Tableau 3.21:** Champs de données (sous-éléments) sous l'élément UpdateBetSlipRequest.

BetSlipItemCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	ItemReferenceNumber	Chaîne de caractères	Numéro de référence attribué à cet élément du bulletin de pari par le système du bookmaker.	Identifiant unique dans ce bulletin de pari uniquement. Utilisé ici pour faire référence à l'élément soumis à cette action de mise à jour.
2	EventStartDate	Date et heure	La date et l'heure UTC à laquelle l'événement sous-jacent commence.	
3	EventResult	Chaîne de caractères	Le résultat final de l'événement sous-jacent.	
4	Status	Énumération: [Pending] [Won] [Lost] [Void]	Le statut de la sélection du joueur pour le marché de cet élément.	Voir tableau 3.23.

**Tableau 3.22:** Champs de données (sous-éléments) sous l'élément BetSlipItemCarrier.

Status	Description
Pending	Le résultat du marché n'a pas encore été décidé.
Won	Le résultat du marché est le même que la sélection du joueur.
Lost	Le résultat du marché est différent de la sélection du joueur.
Void	Le résultat du marché ne peut pas être déterminé (par exemple, un événement annulé).

**Tableau 3.23:** *BetSliptemCarrier* — valeurs du champ **Status**.

BetCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	BetReferenceNumber	Chaîne de caractères	Numéro de référence de ce pari. Attribué par le système du bookmaker.	Identifiant unique dans ce bulletin de pari uniquement. Utilisé ici pour faire référence au pari.
2	Payout	Nombre décimal	Le paiement final total de ce pari. Inclut le bonus.	

**Tableau 3.24:** *Champs de données (sous-éléments) sous l'élément BetCarrier.*

BetStatusCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Status	Énumération [Pending] [Won] [Lost] [Void]	Le statut actuel de ce pari.	Voir tableau 3.26.
2	CreatedOnDate	Date et heure	Date et heure UTC à laquelle ce statut a été atteint dans le système du bookmaker.	

**Tableau 3.25:** *Champs de données (sous-éléments) sous l'élément BetStatusCarrier.*

Status	Description
Pending	Le pari est toujours en attente.
Won	Au moins une des combinaisons sous-jacentes du pari a été réalisée.
Lost	Aucune des combinaisons sous-jacentes du pari n'a été réalisée.
Void	Le pari n'est plus considéré comme valide.

**Tableau 3.26:** *BetStatusCarrier* — valeurs du champ **Status**.

BetSlipStatusCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	Status	Énumération [Submitted] [Accepted] [Cancelled] [Rejected] [Won] [Lost] [Refund] [PaidWon] [PaidRefund]	Le statut actuel de ce bulletin de pari.	Voir la figure 2.2 et le tableau 3.28 pour le cycle de vie du bulletin de pari.
2	CurrentPayout	Nombre décimal	Tout paiement sécurisé par ce bulletin de pari. Inclut le bonus.	
3	SettledStake	Nombre décimal	Tout montant de mise sur ce bulletin de pari qui a été réglé. Inclut le bonus.	
4	CreatedOnDate	Date et heure	La date et l'heure UTC lorsque ce changement dans Status ou CurrentPayout ou SettledStake a été enregistré dans le système du bookmaker.	

**Tableau 3.27:** *Champs de données (sous-éléments) sous l'élément BetSlipStatusCarrier.*

Status	Description
Submitted	Le bulletin de pari a été créé dans le système du bookmaker.
Accepted	Le bookmaker a accepté le bulletin de pari.
Cancelled	Le joueur a annulé le bulletin de pari.
Rejected	Le bookmaker a rejeté le bulletin de pari.
Won	Un ou plusieurs paris sur le bulletin de pari sont gagnants. Le montant n'a pas encore été versé au joueur.
Lost	Aucun des paris sur le bulletin de pari n'est gagnant.
Refund	La mise sera remboursée au joueur.
PaidWon	Les gains ont été versés au joueur.
PaidRefund	Le remboursement a été versé au joueur.

**Tableau 3.28:** *BetSlipStatusCarrier* — valeurs du champ **Status**.

**Remarque:** Chaque bulletin de pari mis à jour en transmettant les données à la méthode *UpdateBetSlips* peut avoir plusieurs éléments *BetSlipStatusCarrier* et/ou *BetStatusCarrier* qui y sont attachés (figure 3.9). Cette fonctionnalité permet d'inclure plus d'un changement dans les paramètres sous-jacents de ces nœuds, dans le même appel. Par exemple, si un bulletin de pari a été Won et PaidWon dans la même fenêtre de 90 secondes, l'expéditeur peut choisir de transmettre les deux changements dans un seul appel à cette méthode en incluant deux éléments distincts et convenablement horodatés *BetSlipStatusCarrier* à sa structure XML.

Voici un exemple de requête SOAP/XML pour cette méthode:

**Requête d'échantillon UpdateBetSlips:**

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/"
xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ
est">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:UpdateBetSlips>

      <tem:header>
        <int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
        <int:LicenseNumber>A-UT-0001</int:LicenseNumber>
        <int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
      </tem:header>

      <tem:betSlips>
        <!--One or more repetitions-->
        <int:UpdateBetSlipRequest>
```

```

<int:BetSlipStatus>
  <!--One or more repetitions:-->
  <BetSlipStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:12.6183845+03:00</int:CreatedOnDate>
    <int:CurrentPayout>412</int:CurrentPayout>
    <int:SettledStake>26</int:SettledStake>
    <int:Status>Won</int:Status>
  </BetSlipStatusCarrier>
  <BetSlipStatusCarrier>
    <int:CreatedOnDate>2019-09-27T19:20:13.6183845+03:00</int:CreatedOnDate>
    <int:CurrentPayout>412</int:CurrentPayout>
    <int:SettledStake>26</int:SettledStake>
    <int:Status>PaidWon</int:Status>
  </BetSlipStatusCarrier>
</int:BetSlipStatus>

<int:Commision>1</int:Commision>
<int:Description>description</int:Description>
<int:Payout>100</int:Payout>
<int:PayoutBonus>10</int:PayoutBonus>
<int:ReferenceNumber>UT-38115dcacfce4-1277-4c56-b48b-7ac46018123</int:ReferenceNumber>
<int:IssuerLicenseNumber>SHOP-UT-0002</int:IssuerLicenseNumber >

<int:BetSlipItems>
  <!--One or more repetitions:-->
  <int:UpdateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>1</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.6193833+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>1</int:ItemReferenceNumber>
    <int:Status>Won</int:Status>
  </int:UpdateBetSlipRequest.BetSlipItemCarrier>

  <int:UpdateBetSlipRequest.BetSlipItemCarrier>
    <int:EventResult>2</int:EventResult>
    <int:EventStartDate>2019-10-07T19:20:13.621383+03:00</int:EventStartDate>
    <int:ItemReferenceNumber>2</int:ItemReferenceNumber>
    <int:Status>Won</int:Status>
  </int:UpdateBetSlipRequest.BetSlipItemCarrier>

</int:BetSlipItems>

<int:Bets>
  <!--Zero or more repetitions:-->
  <int:UpdateBetSlipRequest.BetCarrier>
    <int:BetReferenceNumber>1</int:BetReferenceNumber>
    <int:BetStatus>
      <!-- One or more repetitions:-->
      <int:BetStatusCarrier>
        <int:CreatedOnDate>2019-09-27T20:20:13.6233824+03:00</int:CreatedOnDate>
        <int:Status>Won</int:Status>
      </int:BetStatusCarrier>
    </int:BetStatus>
    <int:Payout>120</int:Payout>
  </int:UpdateBetSlipRequest.BetCarrier>

  <int:UpdateBetSlipRequest.BetCarrier>
    <int:BetReferenceNumber>2</int:BetReferenceNumber>

    <int:BetStatus>
      <!-- One or more repetitions:-->
      <int:BetStatusCarrier>
        <int:CreatedOnDate>2019-09-27T20:20:13.6233824+03:00</int:CreatedOnDate>
        <int:Status>PaidWon</int:Status>
      </int:BetStatusCarrier>
    </int:BetStatus>

    <int:Payout>100</int:Payout>
  </int:UpdateBetSlipRequest.BetCarrier>

```

```

        </int:Bets>
    </int:UpdateBetSlipRequest>
</tem:betSlips>
</tem:UpdateBetSlips>
</soapenv:Body>
</soapenv:Envelope>
    
```

### 3.4.4.3 Méthode Response

Quand la méthode *UpdateBetSlips* est appelée, l'API répond avec la structure XML illustrée à la figure 3.10. La réponse fournit un retour d'information sur chaque bulletin de pari reçu, au moyen de son champ *ReferenceNumber*. Les types d'erreurs susceptibles d'être renvoyés sont décrits dans le tableau 3.29.

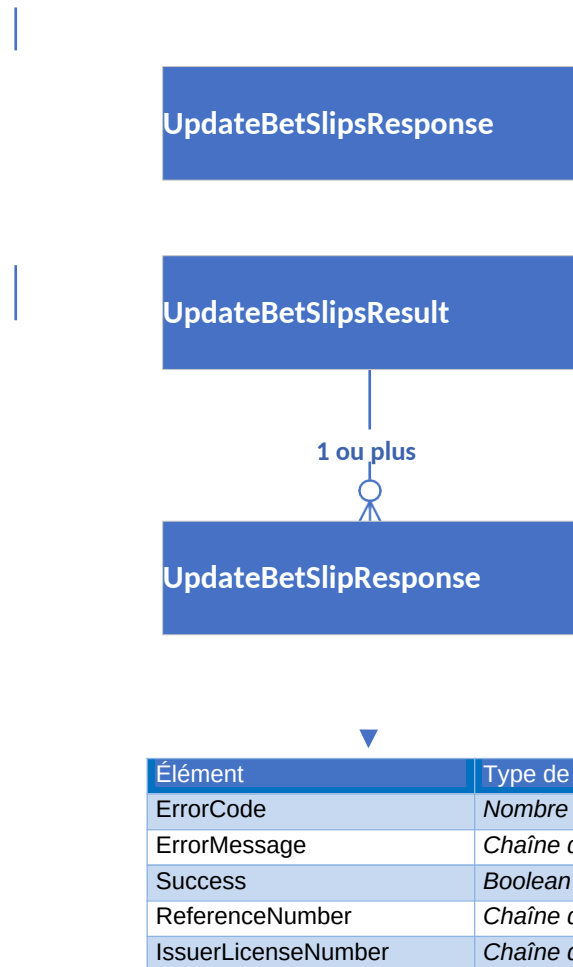


Figure 3.10: La structure XML de *UpdateBetSlipsResponse*.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber: '{0}' LicenseIdentifier: '{1}' DataEntryKey: '{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1501	«Représentant du titulaire de licence '{0}' introuvable»	Licence du représentant inconnue
1503	«Le titulaire de licence {0} ne représente pas le titulaire de licence {1}»	La licence du représentant ne représente pas le bookmaker
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, des codes de pays non valides, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues)
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, BetSlips, BetSlipItems, BetLines, Accounts, AccountTransactions, Restrictions)
1514	«Le titulaire de licence {0} n'accepte pas les paris. Type de titulaire de licence: {1}»	Les bulletins de paris sont autorisés uniquement pour la classe B ou les magasins. La classe A et les représentants doivent associer leurs bulletins de pari à un magasin.

**Tableau 3.29:** Types d'erreurs dans UpdateBetSlipsResponse.

Voici un exemple de réponse SOAP/XML:

Exemple de réponse UpdateBetSlips:
<pre> &lt;s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;s:Body&gt;     &lt;UpdateBetSlipsResponse xmlns="http://tempuri.org/"&gt;       &lt;UpdateBetSlipsResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;         &lt;a:UpdateBetSlipResponse&gt;           &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt;           &lt;a:ErrorMessage i:nil="true"/&gt;           &lt;a:Success&gt;true&lt;/a:Success&gt;           &lt;a:ReferenceNumber&gt;UT-38115dcca4ce4-1277-4c56-b48b-7ac46018123&lt;/a:ReferenceNumber&gt;           &lt;a:RepresentativeLicenseNumber&gt;SHOP-UT-0002&lt;/a:RepresentativeLicenseNumber&gt;         &lt;/a:UpdateBetSlipResponse&gt;       &lt;/UpdateBetSlipsResult&gt;     &lt;/UpdateBetSlipsResponse&gt;   &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

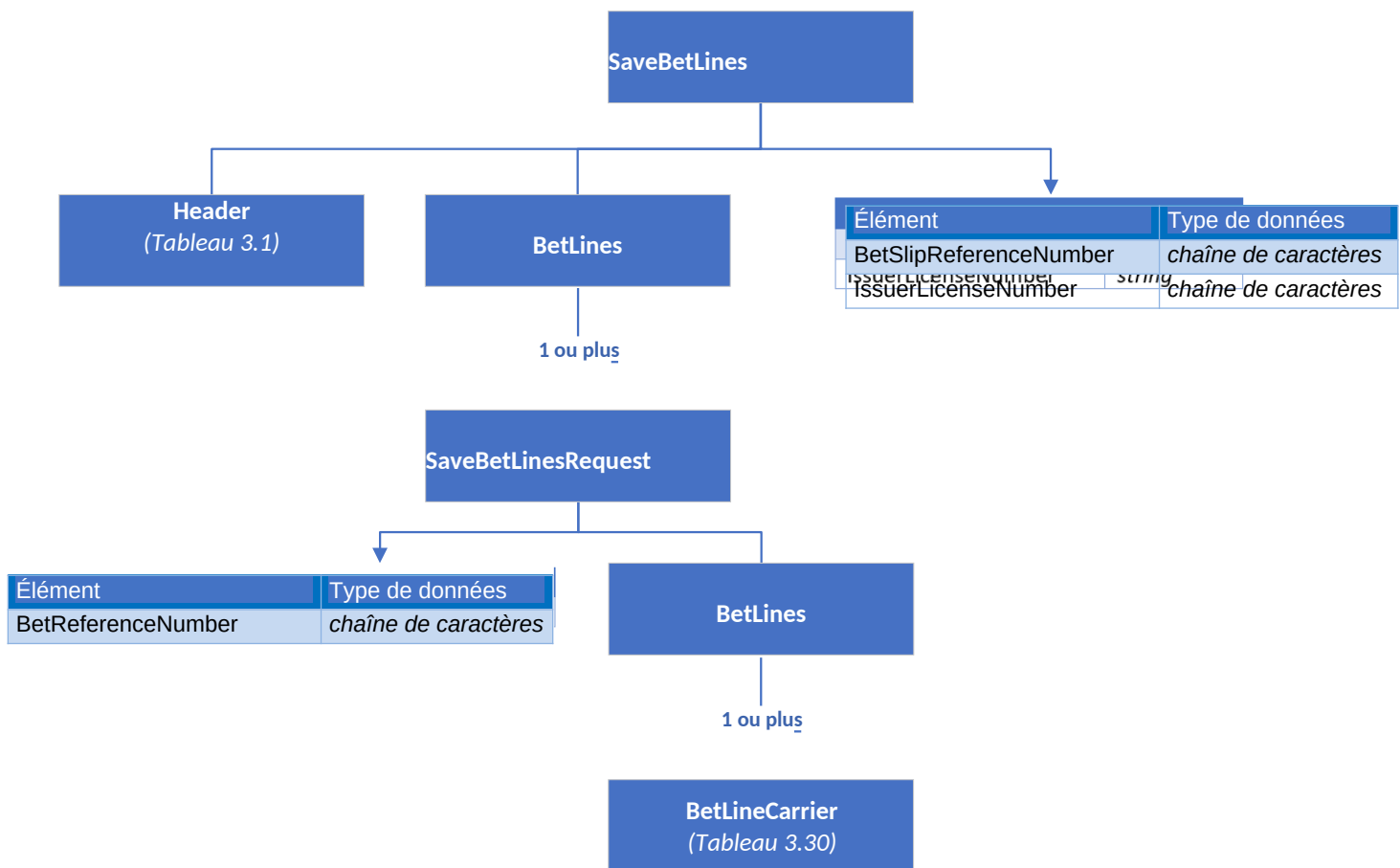
### 3.4.5 Méthode SaveBetLines

#### 3.4.5.1 Description

**Remarque:** La NBA n'exige pas les lignes de n'importe quel pari dans la fenêtre régulière de transmission de données de 90 secondes. Cette méthode doit seulement être appelée à la suite d'une demande de l'Autorité concernant un ou plusieurs bulletins de pari.

Les combinaisons ou lignes dérivées de paris peuvent être exigées par la NBA dans des circonstances exceptionnelles où un bulletin de pari doit être examiné dans ses moindres détails. Pour répondre à de tels cas, l'API du BMRS fournit la fonction *SaveBetLines*, dont la structure est illustrée à la figure 3.11. Les champs de données sous l'élément *BetLineCarrier* sont décrits dans le tableau 3.30.

#### 3.4.5.2 Méthode Request



**Figure 3.11:** La structure XML de la méthode *SaveBetLines*.

BetLineCarrier				
N°.	Nom du champ	Type de données	Description	Commentaires
1	LineReferenceNumber	Nombre entier	Un identifiant unique parmi d'autres lignes de ce pari, attribué par le système du bookmaker.	De préférence un nombre entier séquentiel.
2	BetSlipItemReferenceNumbers	Chaîne de caractères	Une chaîne de caractères séparée par des virgules faisant référence aux éléments du bulletin de pari de cette ligne.	
3	Odds	Nombre décimal	Les cotes combinées de cette ligne.	
4	Stake	Nombre décimal	Le montant de la mise sur cette ligne. Inclut le bonus.	

**Tableau 3.30:** Champs de données (sous-éléments) sous l'élément BetLineCarrier.

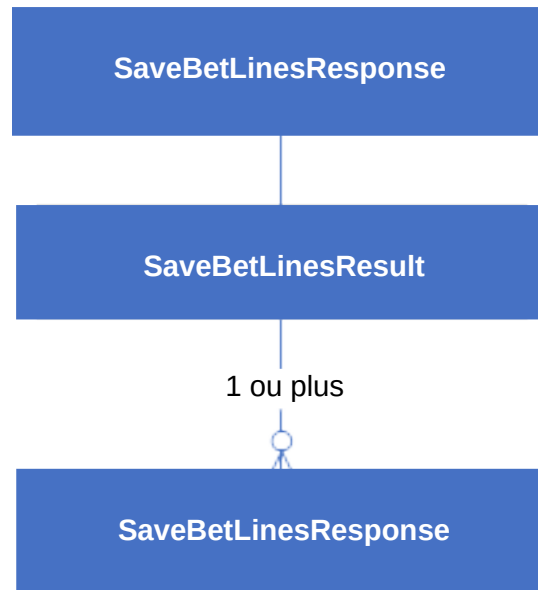
Voici un exemple de requête SOAP/XML de cette méthode:

Requête d'échantillon SaveBetLines
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tem="http://tempuri.org/" xmlns:int="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Requ est"&gt;   &lt;soapenv:Header/&gt;   &lt;soapenv:Body&gt;     &lt;tem:SaveBetLines&gt;        &lt;tem:header&gt;         &lt;int:DataEntryKey&gt;C2EDF452-9B0E-4B75-85BA-D8C95F242371&lt;/int:DataEntryKey&gt;         &lt;int:LicenseNumber&gt;B-UT-0001&lt;/int:LicenseNumber&gt;         &lt;int:LicenseeIdentifier&gt;6254122F-CA27-4034-A629-B855DD0D25EC&lt;/int:LicenseeIdentifier&gt;       &lt;/tem:header&gt;        &lt;tem:betSlipReferenceNumber&gt;UT-38115dcca4e4-1277-4c56-b48b-7ac460188ff4&lt;/tem:betSlipReferenceNumber&gt;       &lt;tem:IssuerLicenseNumber&gt;B-UT-0001&lt;/tem:IssuerLicenseNumber&gt;        &lt;tem:betLines&gt;         &lt;!--One or more repetitions:--&gt;         &lt;int:SaveBetLinesRequest&gt;           &lt;int:BetLines&gt;             &lt;!--One or more repetitions:--&gt;             &lt;int:SaveBetLinesRequest.BetLineCarrier&gt;               &lt;int:BetSlipItemReferenceNumbers&gt;1&lt;/int:BetSlipItemReferenceNumbers&gt;               &lt;int:LineReferenceNumber&gt;1&lt;/int:LineReferenceNumber&gt;               &lt;int:Odds&gt;12&lt;/int:Odds&gt;               &lt;int:Stake&gt;100&lt;/int:Stake&gt;             &lt;/int:SaveBetLinesRequest.BetLineCarrier&gt;           &lt;/int:BetLines&gt;           &lt;int:BetReferenceNumber&gt;?&lt;/int:BetReferenceNumber&gt;         &lt;/int:SaveBetLinesRequest&gt;       &lt;/tem:betLines&gt;     &lt;/tem:SaveBetLines&gt;    &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>



### 3.4.5.3 Méthode Response

À la suite d'un appel à la méthode *SaveBetLines*, la structure de réponse de l'API est illustrée à la figure 3.12. Les types d'erreurs qui pourraient être renvoyés sont décrits dans le tableau 3.31.



Élément	Type de données
ErrorCode	Nombre entier
ErrorMessage	Chaîne de caractères
Success	Boolean
BetReferenceNumber	Chaîne de caractères

**Figure 3.12:** La structure XML de *SaveBetLinesResponse*.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber: '{0}' LicenseIdentifier: '{1}' DataEntryKey: '{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1501	«Représentant du titulaire de licence '{0}' introuvable»	Licence du représentant inconnue
1503	«Le titulaire de licence {0} ne représente pas le titulaire de licence {1}»	La licence du représentant ne représente pas le bookmaker
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)
1505	«Champ '{0}': la valeur '{1}' est non valide»	La valeur transmise n'est pas valide (par exemple, des codes de pays non valides, des énumérations non définies, des valeurs négatives non attendues ou des valeurs nulles non attendues)
1506	«Champ '{0}': la valeur '{1}' est dupliquée»	Les éléments dupliqués existent dans la même requête (par exemple, BetSlips, BetSlipItems, BetLines, Accounts, AccountTransactions, Restrictions)
1514	«Le titulaire de licence {0} n'accepte pas les paris. Type de titulaire de licence: {1}»	Les bulletins de paris sont autorisés uniquement pour la classe B ou les magasins. La classe A et les représentants doivent associer leurs bulletins de pari à un magasin.

**Tableau 3.31:** Types d'erreurs dans SaveBetLinesResponse.

Voici un exemple de réponse SOAP/XML de cette méthode:

Exemple de réponse SaveBetLines:
<pre> &lt;s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;s:Body&gt;     &lt;SaveBetLinesResponse xmlns="http://tempuri.org/"&gt;       &lt;SaveBetLinesResult xmlns:a="http://schemas.datacontract.org/2004/07/InteliScape.NBA.BMRS.BusinessLogic.IntegrationManagement.Carrier.Respo nse" xmlns:i="http://www.w3.org/2001/XMLSchema-instance"&gt;         &lt;a:SaveBetLinesResponse&gt;           &lt;a:ErrorCode&gt;0&lt;/a:ErrorCode&gt;           &lt;a:ErrorMessage i:nil="true"/&gt;           &lt;a:Success&gt;true&lt;/a:Success&gt;           &lt;a:BetReferenceNumber?&lt;/a:BetReferenceNumber&gt;         &lt;/a:SaveBetLinesResponse&gt;       &lt;/SaveBetLinesResult&gt;     &lt;/SaveBetLinesResponse&gt;   &lt;/s:Body&gt; &lt;/s:Envelope&gt; </pre>

## 3.5 Appels système

Cette dernière classe de méthodes du BMRS agit en dehors du champ d'application de toute activité commerciale de paris et est utilisée pour fournir un retour d'information aux deux extrémités du canal de communication d'échange de données, à savoir l'API et le système du bookmaker.

### 3.5.1 Méthode Heartbeat

#### 3.5.1.1 Description

Un appel à la méthode CreateHeartbeat est un signal horodaté du système du bookmaker vers le BMRS et confirme que la connexion entre les deux points d'extrémité est active, ce qui devient pertinent pendant les périodes de faible trafic, lorsque d'autres méthodes ne sont plus appelées parce qu'il n'y a pas de données à transmettre.

**Remarque:** Un appel périodique à cette fonction est obligatoire, c'est-à-dire qu'il n'est pas dépendant des données car il ne transporte aucune donnée autre qu'un horodatage.

#### 3.5.1.2 Méthode Request

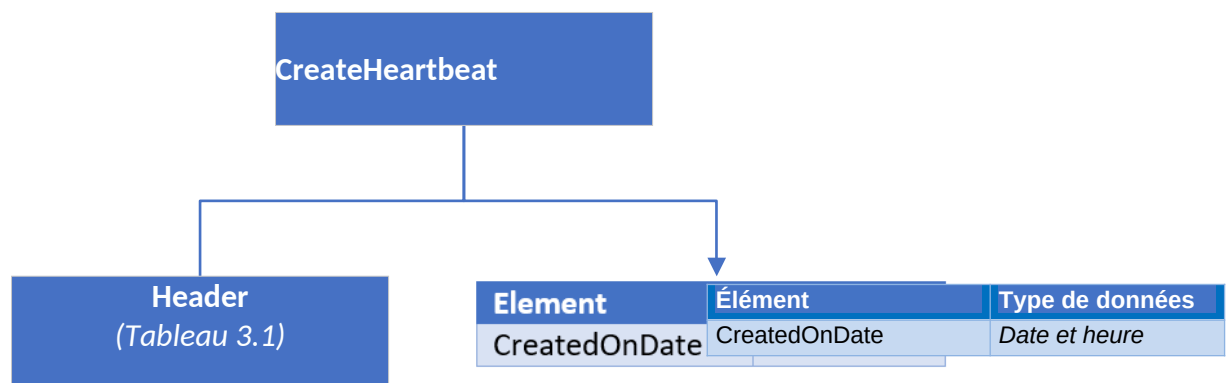


Figure 3.13: La structure XML de la méthode CreateHeartbeat.

Voici un exemple de requête SOAP/XML de cette méthode:

Requête d'échantillon Heartbeat:
<pre> &lt;soapenv:Envelope xmlns:soapenv="http" xmlns:tem="" xmlns:inttt est"&gt; &lt;soapenv:Header/&gt; &lt;soapenv:Body&gt; &lt;tem:CreateHeartbeat&gt; &lt;tem:header&gt; &lt;int:DataEntryKey&gt;8F063140-4D72-3F01-1E8C-09C90F86722D&lt;/int:DataEntryKey&gt; &lt;int:LicenseNumber&gt;A-90181&lt;/int:LicenseNumber&gt; &lt;int:LicenseIdentifier&gt;B3411DE4-5E4B-4199-FC08-4C4E0B49C2CF&lt;/int:LicenseIdentifier&gt; &lt;/tem:header&gt; &lt;tem:createdOnDate&gt;2019-09-27T15:35:52.939Z&lt;/tem:createdOnDate&gt; &lt;/tem:CreateHeartbeat&gt; &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

## 3.5.2 Méthode GetErrorLogs

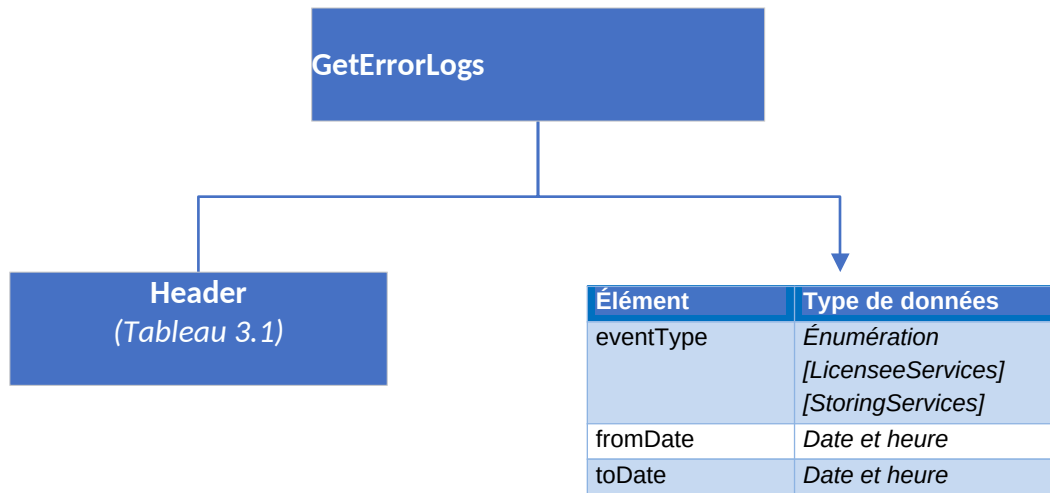
### 3.5.2.1 Description

Comme indiqué précédemment, le BMRS renvoie une réponse à chaque fois que l'API est appelée, et fournit, à l'intérieur de cette réponse, des commentaires pour toutes les erreurs possibles qui auraient pu être rencontrées dans la tentative de traiter le paquet de données envoyé. Ces erreurs, cependant, sont d'une portée limitée car elles sont le résultat d'une vérification préliminaire des données qui a lieu pendant le processus de mise en file d'attente de données (services aux titulaires de licence) et avant toute opération au niveau de la base de données (services de stockage). Par conséquent, le système émetteur (bookmaker) n'est pas informé des incidents qui auraient pu se produire plus en aval dans le processus de stockage.

La méthode *GetErrorLogs* vise à atténuer la limitation décrite ci-dessus et à fournir un retour d'information à un niveau plus approfondi, à la demande de l'expéditeur. La structure XML de l'appel à cette méthode est illustrée à la figure 3.14 et la réponse équivalente à la figure 3.15. Le tableau 3.34 répertorie les types d'erreurs qui pourraient être renvoyés par ce dernier.

**Remarque:** Contrairement aux autres méthodes de l'API, *GetErrorLogs* ne suit pas un cycle régulier et n'est appelé que lorsque l'expéditeur souhaite examiner les erreurs.

### 3.5.2.2 Méthode Request



**Figure 3.14:** La structure XML de la méthode GetErrorLogs.

Voici un exemple de requête SOAP/XML pour cette méthode:

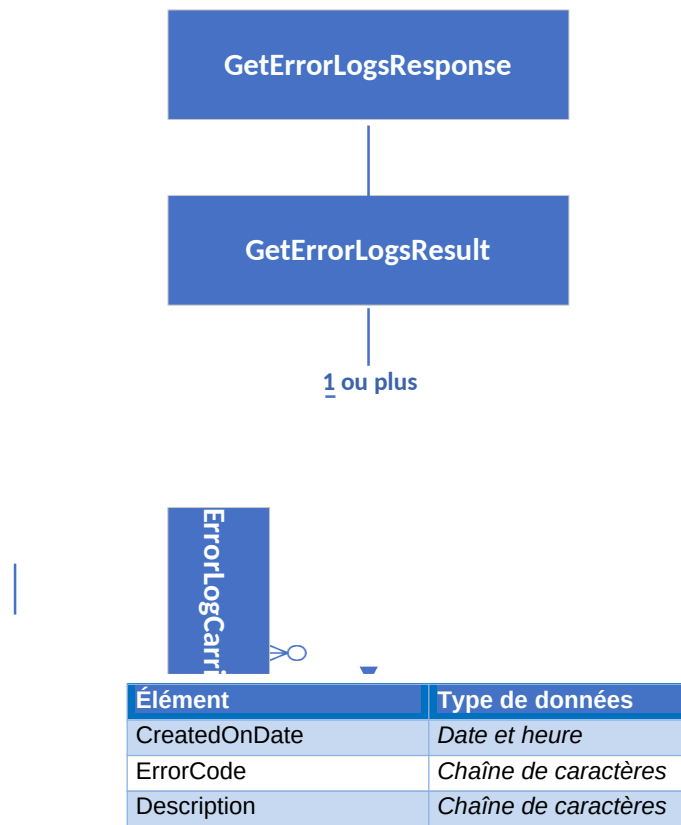
**Requête d'échantillon GetErrorLogs:**

```
<soapenv:Envelope xmlns:soapenv="http" xmlns:tem="" xmlns:inttt est">
<soapenv:Header/>
<soapenv:Body>
<tem:GetErrorLogs>
<tem:header>
<int:DataEntryKey>F2D64583-DFC1-4DF1-A95D-02BCD1CBA489</int:DataEntryKey>
<int:LicenseNumber>A-UT-0001</int:LicenseNumber>
<int:LicenseeIdentifier>19666639-C7F8-4942-876D-CA1088B75EFE</int:LicenseeIdentifier>
</tem:header>
<tem:eventType>StoringService</tem:eventType>
<tem:fromDate>2019-09-20T15:00:00.000Z</tem:fromDate>
<tem:toDate>2019-09-27T15:00:00.000Z</tem:toDate>
</tem:GetErrorLogs>
</soapenv:Body>
</soapenv:Envelope>
```

EventType	Description
LicenseeService	Erreurs détectées lors de la soumission à l'API.
StoringService	Erreurs détectées lors du traitement des messages pour le stockage.

**Tableau 3.33:** Types d'événements dans GetErrorLogs.

### 3.5.2.3 Méthode Response



**Figure 3.15:** La structure XML de GetErrorLogsResponse.

ErrorCode	ErrorMessage	Description
1500	«Aucun titulaire de licence trouvé pour LicenseNumber: '{0}' LicenseeIdentifiant: '{1}' DataEntryKey: '{2}'»	Les informations sur le titulaire de licence transmises dans l'en-tête ne correspondent à aucun titulaire de licence dans le BMRS (par exemple, LicenceNo non valide, DataEntryKey non valide, etc.)
1504	«Champ '{0}': la valeur est nulle ou vide»	Une valeur ou un élément obligatoire est manquant (par exemple, l'en-tête de la requête)

**Tableau 3.34:** Types d'erreurs LicenseeService dans GetErrorLogsResponse.

Exemple de réponse GetErrorLogs

```

<s:Envelope xmlns:s="http">
  <s:Body>
    <GetErrorLogsResponse xmlns="http">
      <GetErrorLogsResult xmlns:a=" nse" xmlns:i="htt">
        <a:ErrorLogCarrier>
          <a:CreatedOnDate>2019-09-25T15:40:38+03:00</a:CreatedOnDate>
          <a:Description>Storing transaction failed. Transaction with Reference Number: 'RefNo1' for Username: 'testxp1' already exist</a:Description>
          <a:ErrorCode>1507</a:ErrorCode>
        </a:ErrorLogCarrier>
        <a:ErrorLogCarrier>
          <a:CreatedOnDate>2019-09-25T15:40:51+03:00</a:CreatedOnDate>
          <a:Description>Storing transaction failed. Transaction with Reference Number: 'RefNo1' for Username: 'testxp1' already exist</a:Description>
          <a:ErrorCode>1507</a:ErrorCode>
        </a:ErrorLogCarrier>
      </GetErrorLogsResult>
    </GetErrorLogsResponse>
  </s:Body>
</s:Envelope>

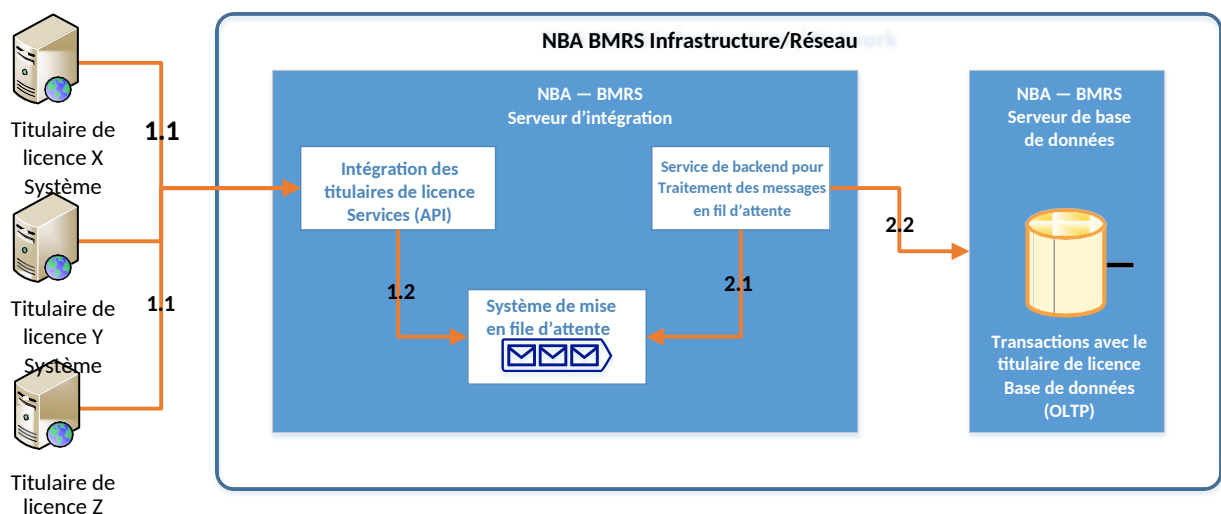
```

## 4.0 Protocoles de communication BMRS

La communication entre la plateforme d'un bookmaker et le BMRS de la NBA reposera sur un protocole primaire et sur un protocole secondaire.

### 4.1 Protocole de communication primaire — L'API BMRS

Le protocole de communication primaire prend la forme d'une interface de programmation d'application (API) disponible sur Internet, à travers laquelle les entités sous licence envoient les données requises à la NBA dans une fenêtre de temps de 90 secondes. Au niveau de la réception, un mécanisme de mise en file d'attente est utilisé par le BMRS pour assurer un flux de données ininterrompu et contrer les goulets d'étranglement potentiels dans la transmission pendant les heures de pointe, comme illustré dans le diagramme ci-dessous.



**Figure 4.1:** Aperçu de l'infrastructure du BMRS.

La liste suivante énumère le flux élevé de réception et de stockage des données:

- **Étape 1.1:** Le système du titulaire de licence envoie des données à l'API
- **Étape 1.2:** L'API valide les données et les stocke dans le système de file d'attente. Avant de quitter, l'API renvoie une réponse qui peut contenir une liste d'erreurs.
- **Étape 2.1:** Un fil d'exécution séparé s'exécute constamment et lit les messages stockés dans la file d'attente.
- **Étape 2.2:** Les messages dans la file d'attente sont traités et stockés dans la base de données des transactions du titulaire de licence.

### 4.1.1 Spécifications techniques

Le protocole d'interface API utilisé par le BMRS est les services web **XML/SOAP 1.1**, complétés par des métadonnées communicatives sur des fichiers WSDL pour permettre la découverte et l'interprétation efficaces des méthodes disponibles. Ce protocole est maintenu par le «World Wide Web Consortium (W3C)» et repose sur des protocoles de couche d'application qui sont indépendants du système d'exploitation et donc largement adoptables.

#### **À propos de SOAP**

*SOAP est une spécification de protocole de messagerie pour l'échange d'informations structurées dans la mise en œuvre de services web dans les réseaux informatiques. Son but est d'induire l'extensibilité, la neutralité et l'indépendance. Il utilise l'ensemble d'information XML pour son format de message, et s'appuie sur les protocoles de couche d'application, le plus souvent Hypertext Transfer Protocol (HTTP) ou Simple Mail Transfer Protocol (SMTP), pour la négociation et la transmission de messages. SOAP permet aux processus s'exécutant sur des systèmes d'exploitation différents (tels que Windows et Linux) de communiquer en utilisant le langage de balisage extensible (XML). Puisque des protocoles web comme HTTP sont installés et exécutés sur tous les systèmes d'exploitation, SOAP permet aux clients d'invoquer des services web et de recevoir des réponses indépendamment du langage et des plateformes.*

#### **À propos de WSDL**

*Le langage de description des services web (WSDL) est un langage de définition d'interface basé sur XML qui est utilisé pour décrire les fonctionnalités offertes par un service web. L'acronyme est également utilisé pour toute description WSDL spécifique d'un service web (également appelé fichier WSDL), qui fournit une description lisible par machine de la façon dont le service peut être appelé, des paramètres auxquels il s'attend et des structures de données qu'il renvoie. Par conséquent, son but est à peu près similaire à celui d'une signature de type dans un langage de programmation.*

Il est recommandé aux développeurs d'utiliser le fichier WSDL fourni des API BMRS pour générer les classes proxy dans leurs systèmes en utilisant l'emplacement réseau du fichier WSDL ou en important le fichier dans leurs projets. Cela permettra la génération automatique correcte de toutes les méthodes, classes et champs connexes, éliminant ainsi les erreurs éventuelles résultant de la création manuelle de toutes les commandes de requête et de réponse XML.

#### 4.1.2 Authentification/autorisation

L'authentification des systèmes connectés au BMRS est réalisée au moyen de secrets partagés (une clé API — des UUID aléatoires), qui doivent être intégrés dans l'en-tête de chaque requête de lots adressée aux services web. Il est à noter que les services web seront apatrides, donc les appels seront authentifiés sur toutes les requêtes.

Tant l'authentification que l'autorisation ont lieu au **niveau de la licence**, ce qui signifie que chaque entreprise aura un compte système par licence (par exemple, un pour la classe A et un pour la classe B dans le cas où ils ont les deux). Les titulaires de licence seront autorisés à envoyer des données concernant les activités de paris qui sont liées aux magasins représentatifs qui leur sont associés.

#### 4.1.3 Sécurité (TLS)

La sécurité de la couche de transport (TLS), autrement connue sous le nom de couche de sockets sécurisée (SSL), est utilisée pour protéger le canal de communication entre les systèmes des titulaires de licence et les services web exposés par le BMRS. La TLS protégera les données pendant qu'elles sont en transit en les cryptant.

#### 4.1.4 Règles d'accès au réseau

Les services web du BMRS suivront les règles d'accès au réseau au niveau du pare-feu pour limiter leur exposition à une liste blanche prédéfinie d'adresses IP. À ce titre, les entités réglementées devront communiquer à l'Autorité les adresses IP de leurs systèmes afin de permettre un accès approprié au réseau.

## 4.2 Protocole de communication secondaire — Fichiers XML

Le protocole d'intégration secondaire prendra la forme de fichiers XML et doit seulement être employé avec le consentement de la NBA.

Un schéma XML prédéfini (XSD) est mis à la disposition des titulaires de licence afin de s'assurer que les données sont conformes à une norme unifiée. Les entités réglementées devront préparer les données conformément au schéma XSD et les envoyer aux agents

informatiques de la NBA pour import. Les entités réglementées devront préparer les données conformément au schéma XSD à intervalles réguliers, dont la période doit être convenue avec l'Autorité. Ces fichiers doivent alors être hébergés dans un endroit en ligne sécurisé auquel la NBA a accès pour la récupération de données suivant un protocole «pull» ou envoyés à l'Autorité par d'autres canaux de communication.

Les mêmes bases de données décrites dans l'article précédent (Protocole primaire — Services web) seront utilisées pour les modèles des fichiers XML.