





Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Issue 19 (2016-02-04)

This issue is the nineteenth official release, and includes the following changes:
Update the description of **channelID**.

Issue 18 (2016-01-08)

This is the eighteenth official release
Add **messageID** and **syncFlag** in 2.2.3 Request.

Issue 17 (2015-12-25)

This issue is the seventeenth official release, and includes the following changes:
Add resultCode **7702 : Unsubscription thriggered because user chose 'not renew' or didn't reply for renewal subscription** for **updateDesc** in 2.2.3 Request.
Add the description for the all fields when **updateType** is 5(block) or 6(unblock) in 2.2.3 Request.
Add **0, 16, 17, 116, 117, 142** in **ChannelId** definition in 2.2.3 Request.

Issue 16 (2015-06-18)

This issue is the sixteenth official release, and includes the following changes:
Add reason **6: Automatical Unsubscription due to User being put in blacklist** for **updateReason** in 2.2.3 Request.

Issue 15 (2015-03-31)

This issue is the fifteenth official release, and includes the following changes:
Optimize the description of **isFreePeriod**.

Issue 14 (2015-03-02)

This issue is the fourteenth official release, and includes the following changes:
Optimize the description of **fee**.

Issue 13 (2015-01-12)

This issue is the thirteenth official release, and includes the following changes:
Add the extended parameter **customFields** for **SyncOrderRelationRequest**.

Issue 12 (2014-05-04)

This issue is the twelfth official release, and includes the following changes for Phase3.2:
Add **block**, and **unblock** fields in the **SyncOrderRelation** interface.
Change the description of **serviced**.

Issue 11 (2014-04-01)

This issue is the eleventh official release, and includes the following changes for Phase3.1:
Add **dedicatedAccounOperationType**, **DAExpireDate**, and **DAUnit** fields in the **syncOrderRelationship** interface.



Change History

Issue 10 (2014-03-29)

This issue is the tenth official release, and includes the following changes:

Add the **Level of Requirement** description for the extended parameters transmitted in extensionInfo.

Issue 09 (2013-12-23)

This issue is the ninth official release, and includes the following changes:

Add the fake ID function in Phase2.3 version.

To use the fake ID function, SPs must modify their systems to change all mobile numbers involved in old and new services to fake IDs so that numbers sent by the SP systems to the SDP are all fake IDs. The SDP converts the received fake IDs to mobile numbers for service processing.

SPs must obtain the mapping between mobile numbers involved in old services and fake IDs from the MTN carrier.

If SPs still use mobile numbers when the fake ID function is enabled, service processing will fail.

Issue 08 (2013-11-21)

This issue is the eighth official release, and includes the following changes:

Change the description of the **isFreePeriod**, **effectiveTime**, **accessCode**, **payType**, **transactionID**, **keyword**, **serviceAailability**, **try**, and **updateDesc** field and add the **operCode** field in syncOrderRelation request.

This issue associates with SDP Solution Phase2.2&Phase2.3.

Issue 07 (2013-11-20)

This issue is the seventh official release, and includes the following changes:

Change the length of the **chargeMode** field and the description of the **isFreePeriod**, **MDSPSUBEXPMODE**, **objectType**, **servicePayType**, **fee**, **cycleEndTime**, and **channelID** field in syncOrderRelation request.

This issue associates with SDP Solution Phase2.2&Phase2.3.

Issue 06 (2013-11-08)

This issue is the sixth official release, and includes the following changes:

Change the description of the **serviceld** field and add the **Starttime** field in syncOrderRelation request.

This issue associates with SDP Solution Phase2.3.

Issue 05 (2013-09-22)

This issue is the fifth official release, and includes the following changes:

Change the format from the Huawei style to MTN style.

Issue 04 (2013-09-05)

This issue is the fourth official release, and includes the following changes:

Updated 2.2.3 Request chapter for adding **updateReason**, **operatorID**, **servicePayType**, **fee** field and delete **operCode**, **parentName**, **shortMessage**, **shortcode** and **Status** field.

Issue 03 (2013-08-09)

This issue is the third official release, and includes the following changes:

Added API Functions, Level of Requirement for Parameters, Request Format, Response Format, Namespaces and SOAPAction in the Overview.



Change History

Issue 02 (2013-08-06)

This issue is the second official release, and includes the following changes:
Updated Chapter 2.1 SyncOrderRelation.

Issue 01 (2013-06-22)

This issue is the first official release.



Contents

1 Overview	1
1.1 API Functions	1
1.2 Level of Requirement for Parameters	1
1.3 Request Format	2
1.4 Response Format	2
1.5 Namespace	4
1.6 SOAPAction	4
2 Interface for Subscription Relationship Synchronization	5
2.1 Process	5
2.2 SyncOrderRelation	5
2.2.1 API Function	5
2.2.2 Request URI	6
2.2.3 Request	6
2.2.4 Response	21
2.2.5 Error Codes	22

01

Overview

1.1 API Functions

The SDP provides data synchronization capability application programming interfaces (APIs) for third-party applications (App for short) to connect to it and use its data synchronization capability to synchronize subscription relationships. App is generally developed by various partners of the SDP.



NOTE

Partners are the enterprises and individuals who sign a contract and cooperate with carriers in utilizing the SDP. Partners include service Partners, Developers, and API Partners. In this document, partners are mainly the service Partners, Developers, and API Partners who use APIs for secondary development.

Table 1-1 describes functions of DataSync capability APIs provided by the SDP.

Table 1-1 Functions of DataSync capability APIs

Function	Description	API
Subscription relationship synchronization	<p>The SDP (functioning as the client) invokes an API to synchronize subscription relationships to the App (functioning as the server).</p> <p>Partners must code the App based on the API field requirements so that the App can correctly parse and respond to requests received from the SDP. The App must send a response to the SDP within 30 seconds.</p>	2.2 SyncOrderRelation

1.2 Level of Requirement for Parameters

The App must develop APIs based on the level of requirement for each parameter.

Table 1-2 Level of requirement for parameters

Type	Description
Mandatory	<p>A parameter is always mandatory in a request.</p> <p>Parameters with the Mandatory requirement are used for access authentication or service processing. If a parameter with the Mandatory requirement is left empty in a request, access authentication or service processing fails and the request fails.</p>



1 Overview

Type	Description
Conditional	A parameter is mandatory or optional in specified conditions. Parameters with the Conditional requirement are used for access authentication or service processing in specified conditions. If the specified conditions is met but a parameter with the Conditional requirement is left empty in a request, access authentication or service processing fails and the request fails.
Optional	A parameter is always optional. Parameters with the Optional requirement are not used for service processing.

1.3 Request Format

The SDP provides the SOAP request in the following format:

```
<soapenv:Envelope>
  <soapenv:Header>
    <parameter>...</parameter>
  ...
</soapenv:Header>
  <soapenv:Body>
    <parameter>...</parameter>
  ...
</soapenv:Body>
</soapenv:Envelope>
```

Table 1-3 Request format

Element	Description
<soapenv:Envelope>	Root element in a request, which specifies the namespace.
<soapenv:Header>	Request header. Parameters in this element are defined by the SDP and are mainly information to be processed by the SDP services, including access authentication parameters.
<soapenv:Body>	Request body. Parameters in this element comply with the SOAP protocol.

1.4 Response Format

Success Response Format

The SDP provides the SOAP API success responses in the following format:

```
<soapenv:Envelope>
  <soapenv:Body>
```



1 Overview

```
<parameter>...</parameter>
...
</soapenv:Body>
</soapenv:Envelope>
```

Table 1-4 Success response format

Element	Description
<soapenv:Envelope>	Root element in a success response, which specifies the namespace.
<soapenv:Body>	Success response body. Parameters in this element comply with the SOAP protocol.

Error Response Format

The SDP provides the SOAP API error responses in the following format:

```
<soapenv:Envelope>
  <soapenv:Body>
    <soapenv:Fault>
      <parameter>...</parameter>
    ...

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

Table 1-5 Error response format

Element	Description
<soapenv:Envelope>	Root element in an error response, which specifies the namespace.
<soapenv:Body>	Error response body, contains the <soapenv:Fault> and <detail> elements. This element specifies the error code and error details.
<soapenv:Fault>	Error code and description. For details about error responses, see API Error Responses.
<detail>	Error details, which are the same as the <soapenv:Fault> element information.



1 Overview

1.5 Namespace

Partners must follow the specified namespaces of data types when developing DataSync capability APIs.

- Table 1-6 describes the namespaces of DataSync capability APIs.
- The namespace of data types used by the DataSync capability APIs is **http://www.csapi.org/schema/parlayx/data/v1_0**.

Table 1-6 Namespaces of DataSync capability APIs

Namespace	API
http://www.csapi.org/wsdl/parlayx/data/sync/v1_0	2.2 SyncOrderRelation

1.6 SOAPAction

Leave the **SOAPAction** parameter empty.

The following is an example of the **SOAPAction** parameter setting in an HTTP header:

```
SOAPAction: ""
```

02

Interface for Subscription Relationship Synchronization

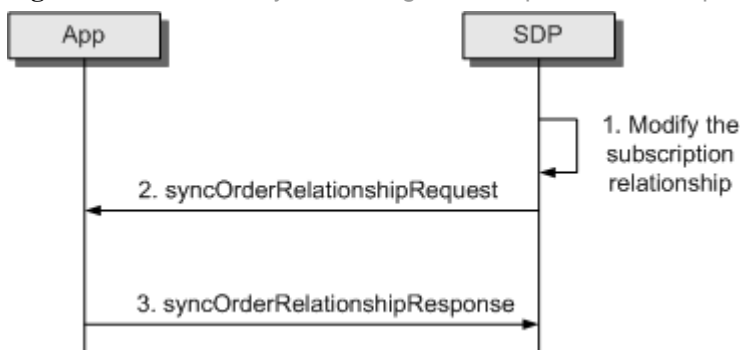
2.1 Process

When a user subscription relationship is generated or modified on the SDP, the SDP uses the SyncOrderRelation interface to synchronize the subscription relationship to App.
The SDP uses the SyncOrderRelation interface to synchronize subscription relationships when:

- A user subscribes to a product on the portal or using an SMS message.
- The carrier subscribes to or unsubscribes from a product for a user.
- Automatic renewal of subscription to a product fails.
- A user unsubscribes from a product on the portal or using an SMS message.

Figure 2-1 shows the process of synchronizing a subscription relationship.

Figure 2-1 Process of synchronizing a subscription relationship to a third party



2.2 SyncOrderRelation

2.2.1 API Function

The SDP (functioning as the client) invokes an API to synchronize subscription relationships to the App (functioning as the server).

Partners must code the App based on the API field requirements so that the App can correctly parse and respond to requests received from the SDP. The App must send a response to the SDP within 30 seconds.



2 Interface for Subscription Relationship Synchronization

2.2.2 Request URI

The request URI is the destination URI of SyncOrderRelation messages sent by the SDP to the App. The URI is defined by the App.

2.2.3 Request

The SDP functions as the client and sends a **SyncOrderRelationRequest** message to the App.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <ns1:syncOrderRelation
      xmlns:ns1="http://www.csapi.org/schema/parlayx/data/sync/v1_0/local">
      <ns1:userID>
        <ID>8619800000001</ID>
        <type>0</type>
      </ns1:userID>
      <ns1:spID>001100</ns1:spID>
      <ns1:productID>1000000423</ns1:productID>
      <ns1:serviceID>0011002000001100</ns1:serviceID>
      <ns1:serviceList>0011002000001100</ns1:serviceList>
      <ns1:updateType>1</ns1:updateType>
      <ns1:updateTime>20130723082551</ns1:updateTime>
      <ns1:updateDesc>Addition</ns1:updateDesc>
      <ns1:effectiveTime>20130723082551</ns1:effectiveTime>
      <ns1:expiryTime>20361231160000</ns1:expiryTime>
      <ns1:extensionInfo>
        <item>
          <key>accessCode</key>
          <value>20086</value>
        </item>
        <item>
          <key>operCode</key>
          <value>zh_CN </value>
        </item>
        <item>
          <key>chargeMode</key>
          <value>0</value>
        </item>
        <item>
          <key>MDSPSUBEXPMODE</key>
          <value>1</value>
        </item>
        <item>
```



2 Interface for Subscription Relationship Synchronization

```
<key>objectType</key>
<value>1</value>
</item>
<item>
  <key>Starttime</key>
  <value>20130723082551</value>
</item>
<item>
  <key>isFreePeriod</key>
  <value>false</value>
</item>
<item>
  <key>operatorID</key>
  <value>26001</value>
</item>
<item>
  <key>payType</key>
  <value>0</value>
</item>
<item>
  <key>transactionID</key>
  <value>504016000001307231624304170004</value>
</item>
<item>
  <key>orderKey</key>
  <value>999000000000000194</value>
</item>
<item>
  <key>keyword</key>
  <value>sub</value>
</item>
<item>
  <key>cycleEndTime</key>
  <value>20130822160000</value>
</item>
<item>
  <key>durationOfGracePeriod</key>
  <value>1</value>
</item>
<item>
  <key>serviceAvailability</key>
  <value>0</value>
</item>
<item>
  <key>durationOfSuspendPeriod</key>
  <value>0</value>
</item>
```



2 Interface for Subscription Relationship Synchronization

```
<item>
  <key>fee</key>
  <value>0</value>
</item>
<item>
  <key>servicePayType</key>
  <value>0</value>
</item>
<item>
  <key>cycleEndTime</key>
  <value>20130813021650</value>
</item>
<item>
  <key>channelID</key>
  <value>1</value>
</item>
<item>
  <key>TraceUniqueID</key>
  <value>504016000001307231624304170005</value>
</item>
<item>
  <key>rentSuccess</key>
  <value>true</value>
</item>
<item>
  <key>try</key>
  <value>false</value>
</item>
</ns1:extensionInfo>
</ns1:syncOrderRelation>
</soapenv:Body>
</soapenv:Envelope>
```

Message Header Parameters

None

Message Body Parameters

Table 2-1 describes parameters in a **syncOrderRelationRequest** message body.



2 Interface for Subscription Relationship Synchronization

Table 2-1 Parameters in a SyncOrderRelationRequest message body

Parameter	Type	Mandatory or Optional	Description
userID	UserID	Mandatory	User ID and user type. For details, see Table 2-2.
spID	String	Mandatory	ID of the partner to which the product belongs. The ID is automatically allocated by the SDP to the partner after successful registration. To obtain the ID, use the registered account to log in to the SDP management portal and query the account information, or obtain it from the notification email after successful registration. [Example] 000201
productID	String	Mandatory	Product ID. The ID is automatically allocated by the SDP when a partner or the carrier releases a subscribable product. [Example] 1000000201
serviceID	String	Mandatory	ID of the service to which the product belongs. The ID is automatically allocated by the SDP when a partner releases a service. The serviceID must be contained during invocation of a service interface developed by service partners and other partners, and must not be contained during invocation of a capability interface developed by API partners, other partners, and developers. [Example] 0003042000001075
serviceList	String	Optional	This parameter is not used in MTN, the partner should ignore this parameter.



2 Interface for Subscription Relationship Synchronization

Parameter	Type	Mandatory or Optional	Description
updateType	int	Mandatory	<p>Operation type.</p> <ul style="list-style-type: none"> • 1: Add • 2: Delete • 3: Update • 5: block • 6: unblock <p>If block/unblock,SP should change the subscription status,but can't change the period of validity of subscription and other information.</p> <p>[Example]</p> <p>1</p>
updateTime	String	Mandatory	<p>Time when the subscription relationship is updated (UTC time).</p> <p>[Format]</p> <p>yyyyMMddHHmmss</p> <p>[Example]</p> <p>20130227000000</p>



2 Interface for Subscription Relationship Synchronization

Parameter	Type	Mandatory or Optional	Description
updateDesc	String	Optional	<p>Description about the update of the subscription relationship.</p> <p>When the extensionInfo field contains the UpdateReason field, the UpdateReason field specifies the subscription relationship update reason.</p> <ul style="list-style-type: none"> UpdateReason =2 <ul style="list-style-type: none"> 7698: The product is automatically unsubscribed from at the end of the grace period. 7699: The product is automatically unsubscribed from at the end of the suspension period. 7702 : Unsubscription thriggered because user chose 'not renew' or didn't reply for renewal subscription. UpdateReason =3 <ul style="list-style-type: none"> 7330: The user account balance is insufficient. 7333: An internal charging error occurs on the SDP. 7271: The SDP does not have sufficient charging information to perform rating. 7999: Other errors occur. <p>[Example] Add a new subscription.</p>
effectiveTime	String	Optional	<p>Time when the subscription relationship takes effect (UTC time).</p> <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Format] yyyyMMddHHmmss</p> <p>[Example] 20130227000000</p>



2 Interface for Subscription Relationship Synchronization

Parameter	Type	Mandatory or Optional	Description
expiryTime	String	Optional	Expires time of the bill cycle (UTC time). no value if updateType=5 (block) or 6(unblock) [Format] yyyyMMddHHmmss [Example] 20130327000000
extensionInfo	NamedParameterList	Optional	Extended information. For the NamedParameterList information, see Table 2-3. For the extended fields transmitted by NamedParameterList , see Table 2-5.

Table 2-2 UserID parameters

Parameter	Type	Length	Level of Requirement	Description
ID	String	36	Mandatory	User ID. If the user is a mobile phone user, this value is a mobile number or the fake ID containing the country code. [Example] <ul style="list-style-type: none"> Mobile number: 8612312345678 Fake ID: f-245-11900000007639
type	int	2	Mandatory	User type. <ul style="list-style-type: none"> 0: mobile phone user. 10: user fake ID [Example] 0

Table 2-3 NamedParameterList parameters

Parameter	Type	Length	Level of Requirement	Description
item	NamedParameter[0...unbounded]	-	Optional	Data type corresponding to the extended field, which consists of key and value . For the NamedParameters information, see Table



2 Interface for Subscription Relationship Synchronization

Parameter	Type	Length	Level of Requirement	Description
				2-3.

Table 2-4 NamedParameter parameters

Parameter	Type	Length	Level of Requirement	Description
key	String	-	Mandatory	ID of the extended field. The value is specified by the SDP. [Example] try
value	String	-	Mandatory	Value of the key. [Example] true

Table 2-5 Extended parameters transmitted in extensionInfo

Key				Value
Parameter	Type	Length	Level of Requirement	Description
accessCode	String	64	Conditional	The SMS/USSD accesscode that the terminal user sends to subscribe product. no value if updateType=5 (block) or 6(unblock) [Example] 1
operCode	String	40	Conditional	Operation code. For example, this parameter must be set for the subscription of the weather forecast service in monthly fee mode. Users in Shenzhen can enter 0755 to subscribe to the weather forecast of Shenzhen. The language ID can also be included. Subscription language. See the language definition in ISO-639. The format is zh_CN. NOTE Currently, only the subscription language is used. Other information is not used.



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
chargeMode	int	4	Conditional	<p>Charging period of the product.</p> <p>The value is set when a partner releases a subscribable product on the SDP management portal.</p> <ul style="list-style-type: none"> • 0: month • 1: by times • 10: multiday • 13: multimonth • 18: day • 19: week • 20: multiweek • 22: multihour <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Example]</p> <p>0</p>
MDSPSUBEXPMODE	String	1	Conditional	<p>Expiration mode of a subscription relationship upon unsubscription.</p> <ul style="list-style-type: none"> • 1: expires immediately • 2: expires next cycle <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Example]</p> <p>1</p>
objectType	String	1	Conditional	<p>Type of the pricing object.</p> <ul style="list-style-type: none"> • 1: Service • 4: Service in package <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Example]</p> <p>1</p>
Starttime	String	-	Conditional	<p>Start time of the bill cycle (UTC time).</p> <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Format]</p> <p>yyyyMMddHHmmss</p>



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
isFreePeriod	boolean	-	Conditional	Indicates whether the subscription is in the period free of charge. <ul style="list-style-type: none"> • true: yes • false: no The value is none when updateType is set to 5 (block) or 6 (unblock).
operatorID	String	-	Conditional	ID of the carrier operator which the terminal user belong to. no value if updateType=5 (block) or 6(unblock)
payType	int4	-	Conditional	This parameter is not used in MTN, the partner ignores this parameter. Payment type. <ul style="list-style-type: none"> • 0: postpaid user • 1: prepaid user no value if updateType=5 (block) or 6(unblock) [Example] 0
transactionID	String	-	Conditional	This parameter is not used in MTN, the partner ignores this parameter. Transaction ID. The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The partner ignores this parameter. [Example] 100001200101110623021721000011
orderKey	String	20	Conditional	Subscription relationship key that uniquely identifies a subscription record. For services that do not support automatic renewal, each subscription generates a subscription relationship record and generates a new key. [Example] 9990000000000000194



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
keyword	String	100	Conditional	<p>This parameter is not used in MTN, the partner ignores this parameter.</p> <p>Subscription and unsubscription command words.</p> <p>When a partner releases a subscribable product on the SDP management portal, the partner sets the subscription and unsubscription command words if the product can be subscribed to by SMS message.</p> <p>If multiple command words are set, the SDP writes any of them to the request during subscription relationship synchronization.</p> <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Example]</p> <p>sub</p>
durationOfGracePeriod	String	4	Conditional	<p>Grace period of the product.</p> <p>NOTE</p> <p>This parameter is displayed only when updateType is set to 1 and updateType is set to 3.</p> <p>no value if updateType=5 (block) or 6(unblock)</p> <p>[Unit]</p> <p>Day</p> <p>[Example]</p> <p>5</p>
serviceAvailability	String	10	Conditional	<p>Indicates whether users can use the service in the grace period.</p> <ul style="list-style-type: none"> • 1: yes • 0: no <p>NOTE</p> <p>This parameter is displayed only when updateType is set to 1 and updateType is set to 3.</p> <p>no value if updateType=5 (block) or 6 (unblock)</p> <p>[Example]</p> <p>0</p>



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
durationOfSuspendPeriod	String	4	Conditional	Suspend period of the product. no value if updateType=5 (block) or 6 (unblock) [Unit] Day [Example] 5
servicePayType	String	-	Conditional	Payment type. 0: Cash no value if updateType=5 (block) or 6 (unblock)
fee	Int	-	Conditional	The amount info. It is the smallest unit of currency if the default currency unit is not specified. When the value of servicePayType is set to following value, the fee means: 0/untoken: amount of the cash. In the following situation, the value of fee is none: <ul style="list-style-type: none"> • Synchronize the subscription relationship before charging • Unsubscribe the product • Charge failed • Free trial days no value if updateType=5 (block) or 6 (unblock)
cycleEndTime	String	14	Conditional	This parameter is not used in MTN, the Partner ignores this parameter. End time of the bill cycle (UTC time). no value if updateType=5 (block) or 6 (unblock) [Format] yyyyMMddHHmmss [Example] 20130227000000
channelID	Int	4	Conditional	Operation channel. <ul style="list-style-type: none"> • no value if updateType=5 (block) or 6 (unblock) • 0: WAP (WAP1x) • 1: web



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
				<ul style="list-style-type: none"> • 2: SMS • 3: USSD • 4: IVR • 5: CC (CRM) • 6: PDA • 7: WAP • 10: Promotion (This value is used for requests sent by the Promotion.) • 15: BetaZone WEB • 16: BetaZone wap • 17: BetaZone ODP • 20: Widget • 21: ODP • 99: Other • 101: UE(User Equipment) • 102: MDSP CC Portal • 103: channel of unsubscription caused by renewal failed • 105: PCCClient • 110: IVR Call Out • 111: SDP Web Tool (CS) • 112: SDP Web Tool (Admin) • 113: SDP Web Tool (CP) • 115: MMS • 116: WAP AOC • 117: USSD VIP • 120: channel of unsubscription caused by user deregistration • 124: channel of unsubscription caused by the blacklist • 125: SP website (added channel type in the MTN project, which is used for subscription on the SP website.) • 142: Get free period because of community capabilities <p>[Example]</p> <p>1</p>



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
TraceUniqueID	String	30	Conditional	<p>Transaction ID.</p> <p>The ID is automatically generated by the SDP and is used only to trace messages during the SDP commissioning. The partner ignores this parameter.</p> <p>[Example]</p> <p>100001200101110623021721000011</p>
rentSuccessful	boolean	N/A	Conditional	<p>Indicates whether the monthly rental is successfully deducted.</p> <ul style="list-style-type: none"> • true: yes • false: no <p>This parameter is valid only when updateType is set to 3.</p> <p>[Example]</p> <p>true</p>
try	boolean	N/A	Conditional	<p>This parameter is not used in MTN, the Partner ignores this parameter.</p> <p>Indicates whether the product is in the trial user period.</p> <ul style="list-style-type: none"> • true: yes • false: no <p>A trial use period can be set when an partner releases a subscribable product supporting automatic subscription renewal.</p> <p>[Example]</p> <p>true</p>



2 Interface for Subscription Relationship Synchronization

Key				Value
Parameter	Type	Length	Level of Requirement	Description
updateReason	int	4	Conditional	<p>Update reason.</p> <p>When updateType is set to 2, the possible reasons are as follows:</p> <ul style="list-style-type: none"> • 1: A user unsubscribes from a product. • 2: Rental fails to be collected upon subscription renewal. • 3: Unsubscription is triggered when the charging fails upon the first subscription. • 4: Unsubscription is triggered when the service status is abnormal. • 5: Unsubscription is triggered when the user status is abnormal. • 6: Automatical Unsubscription due to User being put in blacklist. <p>[Example]</p> <p>1</p>
dedicatedAccountOperationType	String	-	Conditional	<p>Dedicated account operation type.</p> <ul style="list-style-type: none"> • New • Topup.
DAExpirationDate	String	-	Conditional	<p>Expiration time of the DA (Dedicated Account).</p> <p>[Format]</p> <p>yyyyMMddHHmmss</p> <p>[Example]</p> <p>20130227000000</p>
DAUnit	String	-	Conditional	Dedicated account unit, such as MB/Hr/Min.
customFields	String	1024	Conditional	<p>Customized fields.</p> <p>Using “ ” to separate each field.</p> <p>[Example]</p> <p>key1=value1 key2=value2 key3=value3</p> <p>Note</p> <p>Whether the SyncOrderRelation interface needs to send this parameter to the SP depends on whether the SubscribeProduct and userPromotion interface transmit this parameter.</p>



2 Interface for Subscription Relationship Synchronization

Key		Value		
Parameter	Type	Length	Level of Requirement	Description
messageID	String	40	Conditional	This parameter is not used in MTN, the partner should ignore this parameter.
syncFlag	String	1	Conditional	This parameter is not used in MTN, the partner should ignore this parameter.

2.2.4 Response

The App functions as the server, processes **SyncOrderRelation** messages received from the SDP, and sends **SyncOrderRelationResponse** messages to the SDP.

This topic provides a success response example and describes parameters in the response. If a request fails, the App sends an error response that contains an error code.

Example

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:loc="http://www.csapi.org/schema/parlayx/data/sync/v1_0/local">
  <soapenv:Header/>
  <soapenv:Body>
    <loc:syncOrderRelationResponse>
      <loc:result>0</loc:result>
      <loc:resultDescription>OK</loc:resultDescription>
    </loc:syncOrderRelationResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Message Body Parameters

Table 2-6 describes parameters in a **SyncOrderRelationResponse** message body.

Table 2-6 Parameters in a SyncOrderRelationResponse message body

Parameter	Type	Level of Requirement	Description
result	String	Mandatory	<ul style="list-style-type: none"> 0: The request succeeds. Other values: The request is abnormal. For the error codes, see 2.2.5 Error Codes.
resultDescription	String	Mandatory	Description of the subscription relationship synchronization result.
extensionInfo	NamedParameterList	Optional	Extended parameter.



2 Interface for Subscription Relationship Synchronization

Parameter	Type	Level of Requirement	Description
			No extended information.

2.2.5 Error Codes

Table 2-7 describes **SyncOrderRelation** error codes that the App may return upon an exception.

Table 2-7 SyncOrderRelation error codes

Error Code	Description	Cause
1211	The field format is incorrect or the value is invalid.	The Format value is incorrect or in an incorrect format in the body of the subscription relationship synchronization request.
2030	The subscription relationship already exists.	The subscription relationship already exists in the SP system.
2031	The subscription relationship does not exist.	The subscription relationship does not exist in the SP system.
2032	The service does not exist.	The service to which the product belongs does not exist.
2033	The service is unavailable.	The service to which the product belongs is in an abnormal status.
2034	The service is unavailable.	The service to which the product belongs cannot be subscribed to.
2500	An internal system error occurred.	An internal error occurs in the SP system.

