

События модуля

- События связанные с вебхуками
- События связанные с отгрузками
- События связанные с товарами и модификациями
- События связанные с заказами

События связанные с вебхуками

1. OnPostWebhookProcessing

Это событие вызывается после обработки сущности вебхука. Оно срабатывает как при отложенной обработке вебхука в очереди, так и при обработке в реальном времени. Если в теле запроса вебхука присутствует несколько сущностей, событие будет вызываться для каждой из них отдельно.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnPostWebhookProcessing',
    'OnPostWebhookProcessingHandler'
);

/**
 * @param string $href ссылка на сущность
 * @param string $action действие на которое срабатывает вебхук (CREATE, UPDATE и т.д.)
 * @param string $type код сущности вебхука (product, customerorder и т.д.)
 * @return void
 */
function OnPostWebhookProcessingHandler(string $href, string $action, string $type)
{
    // код обработчика
}
```

События связанные с отгрузками

1. OnBeforeAprixSendDemandToMS

Это событие вызывается перед отправкой запроса в МойСклад на создание/изменение отгрузки. В обработчике события можно модифицировать тело запроса перед отправкой его в МойСклад.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnBeforeAprixSendDemandToMS',
    'OnBeforeAprixSendDemandToMSHandler'
);

/**
 * @param string $action действие на которое срабатывает событие. Может быть CREATE или UPDATE
 * @param array $msData массив тела запроса (передается по ссылке и может быть модифицирован)
 * @return void
 */
function OnBeforeAprixSendDemandToMSHandler(string $action, array &$msData)
{
    // код обработчика
}
```

2. OnBeforeAprixUpdateOrderWithModifiedShipment

Это событие вызывается в цикле синхронизации отгрузок в направлении из МойСклад на сайт. Оно срабатывает перед сохранением заказа после изменения или добавления отгрузки на сайте. Это событие можно использовать для модификации заказа перед его сохранением после обновления отгрузки в заказе.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnBeforeAprixUpdateOrderWithModifiedShipment',
    'OnBeforeAprixUpdateOrderWithModifiedShipmentHandler'
);

/**
 * @param \Bitrix\Sale\Order $order объект заказа сайта
 * @param array $msDemand массив отгрузки из МойСклад
 * @return void
 */
function OnBeforeAprixUpdateOrderWithModifiedShipmentHandler(\Bitrix\Sale\Order $order, array $msDemand)
{
    // код обработчика
}
```

События связанные с товарами и модификациями

1. OnBeforeAprixCreateElementinSite

Это событие вызывается перед созданием товара или торгового предложения на сайте при синхронизации данных из МойСклад в сторону сайта. Для торговых предложений событие срабатывает только при обработке вебхука, во время создания или обновления родительского товара, и не вызывается в цикле синхронизации модификаций.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnBeforeAprixCreateElementinSite',
    'OnBeforeAprixCreateElementinSiteHandler'
);

/**
 * @param string $action действие на которое срабатывает событие. Может быть CREATE или UPDATE
 * @param array $arFields массив полей товара или торгового предложения, по которым они создаются на
 * сайте (передаётся по ссылке и может быть модифицирован).
 * @return void
 */
function OnBeforeAprixCreateElementinSiteHandler(string $action, array &$arFields)
{
    // код обработчика
}
```

2. OnAprixBeforeUpdateProduct

Событие, которое вызывается перед добавлением товара на сайт:

Пример обработчика события (init.php):

```

\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAprixBeforeUpdateProduct',
    'OnAprixBeforeUpdateProductHandler'
);

/**
 * @param int $idProduct идентификатор товара на сайте
 * @param array &$arProduct массив, который содержит поля товара на сайте и поля соответствующего
 товара в МойСклад. Передается по ссылке и может быть модифицирован
 * @return void
 */
function OnAprixBeforeUpdateProductHandler(int $idProduct, array &$arProduct)
{
    // код обработчика
}

```

3. OnAprixAfterUpdateProduct

Событие, которое вызывается после обновления товара на сайте.

Пример обработчика события (init.php):

```

\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAprixAfterUpdateProduct',
    'OnAprixAfterUpdateProductHandler'
);

/**
 * @param int $idProduct идентификатор товара на сайте
 * @param array $arProduct массив, который содержит поля товара на сайте и поля этого же товара в
 МойСклад.
 * @return void
 */
function OnAprixAfterUpdateProductHandler(int $idProduct, array $arProduct)
{
    // код обработчика
}

```

4. OnAprixBeforeAddProduct

Событие, которое вызывается перед созданием товара на сайте.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAprixBeforeAddProduct',
    'OnAprixBeforeAddProductHandler'
);

/**
 * @param array &$arProduct массив, который содержит поля товара на сайте и поля этого же товара в
 * МойСклад. Передается по ссылке, поэтому может быть модифицирован.
 * @return void
 */
function OnAprixBeforeAddProductHandler(array &$arProduct)
{
    // код обработчика
}
```

5. OnAprixAfterAddProduct

Событие вызывается после добавления товара на сайт.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAprixAfterAddProduct',
    'OnAprixAfterAddProductHandler'
);

/**
 * @param int $idProduct идентификатор нового товара, который был добавлен на сайт
 * @param array $arProduct массив, который содержит поля товара на сайте и поля соответствующего
 * товара в МойСклад.
 * @return void
 */
function OnAprixAfterAddProductHandler(int $idProduct, array $arProduct)
{
    // код обработчика
}
```

```
}
```

6. OnAprixAfterAddOffer

Событие, которое вызывается после добавления торгового предложения на сайт.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.synccms',
    'OnAprixAfterAddOffer',
    'OnAprixAfterAddOfferHandler'
);

/**
 * @param int $idSKU идентификатор торгового предложения на сайте
 * @param array $arSKU массив, который содержит поля торгового предложения на сайте и поля
 соответствующей модификации в МойСклад.
 * @return void
 */
function OnAprixAfterAddOfferHandler(int $idSKU, array $arSKU)
{
    // код обработчика
}
```

7. OnAprixAfterUpdateOffer

Событие, которое вызывается после обновления торгового предложения на сайте.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.synccms',
    'OnAprixAfterUpdateOffer',
    'OnAprixAfterUpdateOfferHandler'
);

/**
 * @param int $idSKU идентификатор торгового предложения на сайте
 * @param array $arSKU массив, который содержит поля торгового предложения на сайте и поля
 соответствующей модификации в МойСклад.
 * @return void
 */
```



```
*/  
function OnAprixAfterUpdateOfferHandler(int $idSKU, array $arSKU)  
{  
    // код обработчика  
}
```

8. OnAprixAfterCreateOffer

Событие, которое вызывается после создания торгового предложения на сайте.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(  
    'design2u.syncms',  
    'OnAprixAfterCreateOffer',  
    'OnAprixAfterCreateOfferHandler'  
);  
  
/**  
 * @param int $idSKU идентификатор торгового предложения на сайте  
 * @param array $arSKU массив, который содержит поля торгового предложения на сайте и поля  
 соответствующей модификации в МойСклад.  
 * @return void  
 */  
function OnAprixAfterCreateOfferHandler(int $idSKU, array $arSKU)  
{  
    // код обработчика  
}
```

События связанные с заказами

1. OnBeforeAprixSendOrderToMS

Это событие вызывается перед отправкой запроса в МойСклад на создание заказа. В обработчике события можно модифицировать тело запроса перед отправкой его в МойСклад.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnBeforeAprixSendOrderToMS',
    'OnBeforeAprixSendOrderToMSHandler'
);

/**
 * @param string $action действие на которое срабатывает событие. На данный момент только "CREATE"
 * @param array &$arOrderJSON массив тела запроса (передается по ссылке и может быть
модифицирован)
 * @return void
 */
function OnBeforeAprixSendOrderToMSHandler(string $action, array &$arOrderJSON)
{
    // код обработчика
}
```

2. OnAprixAddBasketItems

Это событие вызывается перед добавлением товаров в корзину при выгрузке заказа из МойСклад на сайт. Оно предназначено для предотвращения добавления в корзину той или иной позиции.

Пример обработчика события (init.php):

```

\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAprixAddBasketItems',
    'OnAprixAddBasketItemsHandler'
);

/**
 * @param int $idOrder идентификатор заказа на сайте
 * @param array &$excludedProductIds пустой массив, который передаётся по ссылке. Добавленные в него
 идентификаторы товаров будут исключены из позиций заказа на сайте.
 * @return void
 */
function OnAprixAddBasketItemsHandler(int $idOrder, array &$excludedProductIds )
{
    // код обработчика
}

```

3. OnBeforeAprixSendUserToMS

Событие которое вызывается перед созданием или обновлением контрагента в МойСклад.

Пример обработчика события (init.php):

```

\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnBeforeAprixSendUserToMS',
    'OnBeforeAprixSendUserToMSHandler'
);

/**
 * @param string $action действие на которое срабатывает событие "CREATE" или "UPDATE"
 * @param array &$arCounterpartyFields массив для создания или обновления контрагента в МойСклад.
 Передаётся по ссылке, поэтому может быть модифицирован в теле обработчика.
 * @param int $idUser Идентификатор пользователя на сайте
 * @return void
 */
function OnBeforeAprixSendUserToMSHandler(string $action, array &$arCounterpartyFields, int $idUser)
{
    // код обработчика
}

```

4. OnAfterAprixSendUserToMS

Событие, которое вызывается после создания контрагента в МойСклад.

Пример обработчика события (init.php):

```
\Bitrix\Main\EventManager::getInstance()->addEventHandler(
    'design2u.syncms',
    'OnAfterAprixSendUserToMS',
    'OnAfterAprixSendUserToMSHandler'
);

/**
 * @param array $arUserData массив данных пользователя на сайте
 * @param string $idCounterparty идентификатор контрагента в МойСклад
 * @return void
 */
function OnAfterAprixSendUserToMSHandler(array &$arUserData, string $idCounterparty)
{
    // код обработчика
}
```