**Описание интеграционного модуля Инфоскан-1С**

Устройство "Инфоскан" может работать в двух режимах:

- Пассивный (формат JSON). Устройство хранит в себе измерения и ждет когда 1С подключится и прочитает их. Для работы в этом режиме предусмотрена "Форма опроса устройства". Можно считать данные с устройства как в режиме реального времени, так и прочитать историю измерений.

- Активный (формат SOAP). В этом режиме устройство при сканировании измерений отправляет данные на HTTP-сервис. Для работы в этом режиме необходимо развернуть web- сервер и опубликовать на нем web-сервисы из данного расширения. В этом режиме в зависимости от настроек устройства можно работу выстроить в несколько этапов. Для этого в конфигурации есть 2 web-сервиса:

CheckBarcode – при первом сканировании товара в 1с запрашивается наименование товара по штрихкоду. Если 1С по пришедшему штрихкоду найдет товар в базе, то отправит его наименование и оно отобразится на устройстве «Инфоскан»

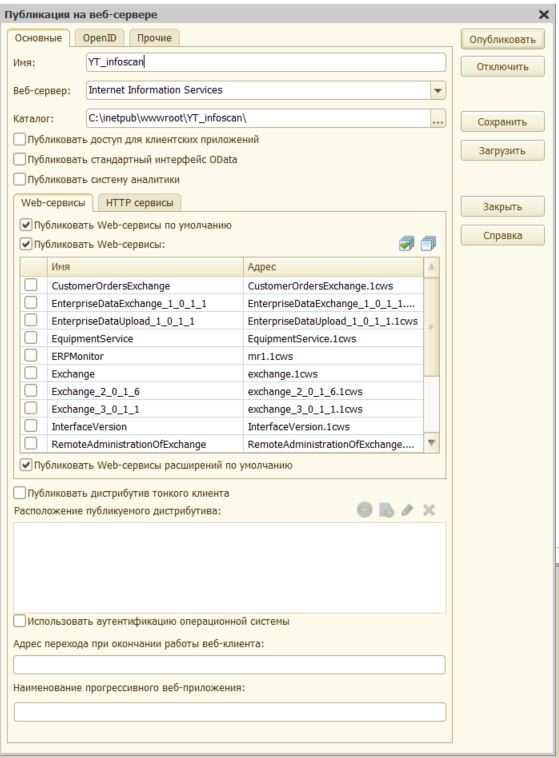
SaveParametrs – при сканировании товара происходят измерения его ВГХ и отправляются в 1С, после чего они сохраняются в бд, в реквизиты указанные в настройках хранения. Этот этап может выполняться как последующий этапу проверки наличия товара в базе, так и самостоятельно без предварительной проверки.

Если публиковать web-сервисы из конфигуратора средствами 1С, то путь, который надо прописать в настройках устройства, будет примерно таким: <http://192.198.10.10/YT_infoscan/ws/infoscan.1cws?wsdl>

где

192.198.10.10 – адрес web-сервера, на котором опубликованы сервисы

YT\_infoscan - имя публикации (обычно совпадает с именем базы)



Для удобства можно включить логирование в настройках, чтобы можно было отследить данные пришедшие с устройства.

Для хранения ВГХ в 1С используют различные сущности в зависимости от сложившегося способа ведения учета. Данное расширение поддерживает "из коробки" несколько способов хранения данных.

От устройства по умолчанию прилетают 4 характеристики:

- Length

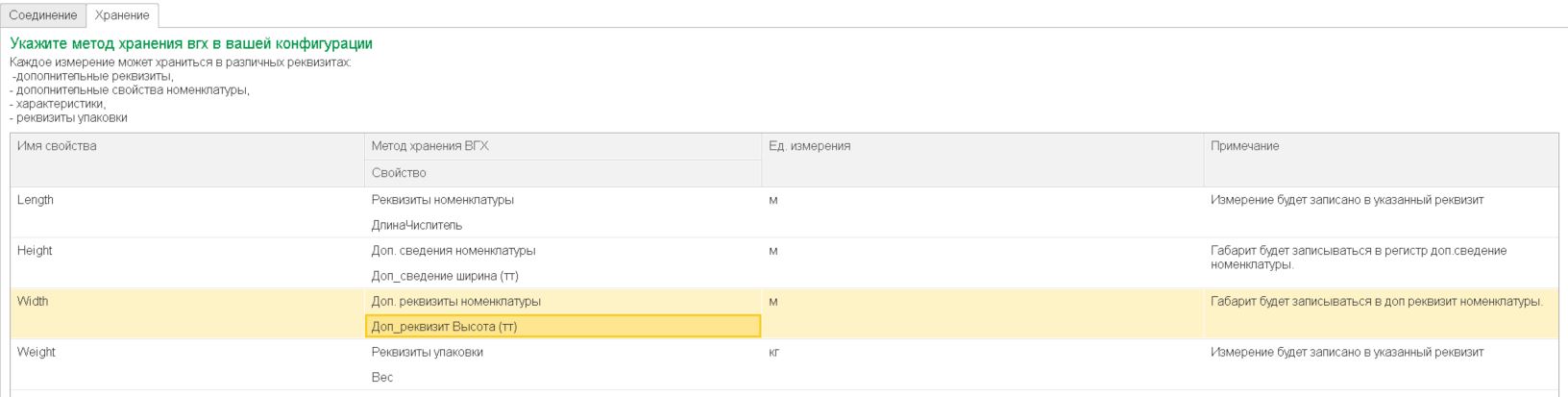
- Height

- Width

- Weight

Каждая характеристика независимо от остальных может храниться в различных типах метаданных.

Ниже приведен пример



Length - значение данной характеристики будет записываться в реквизит ДлинаЧислитель элемента справочника Номенклатура, так метод хранения выбран "Реквизиты номенклатуры"

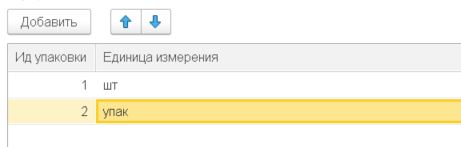
Height - значение данной характеристики будет записываться в Дополнительных сведениях(Регистр сведений) Доп\_сведения ширина

Width - значение данной характеристики будет записываться в Дополнительные реквизиты(табличная часть справочника Номенклатура) Доп\_реквизит Высота

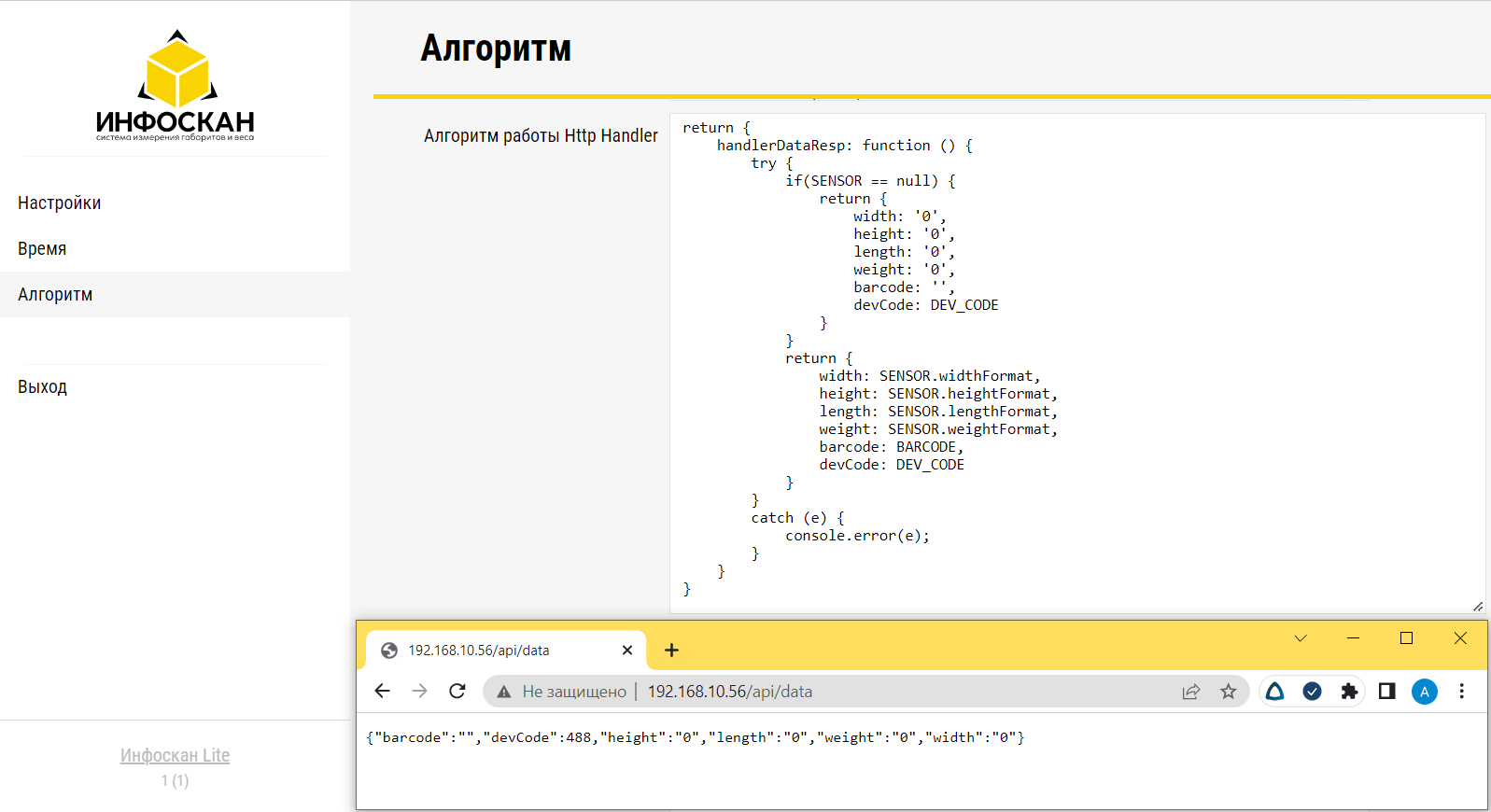
Weight - значение данной характеристики будет записываться в реквизит Вес элемента справочника Упаковки и единицы измерения номенклатуры

Аналогично можно выбрать методы хранения ВГХ Реквизиты Характеристик, Доп сведения Характеристик, Доп реквизиты характеристик, если в учете используются характеристики номенклатуры.

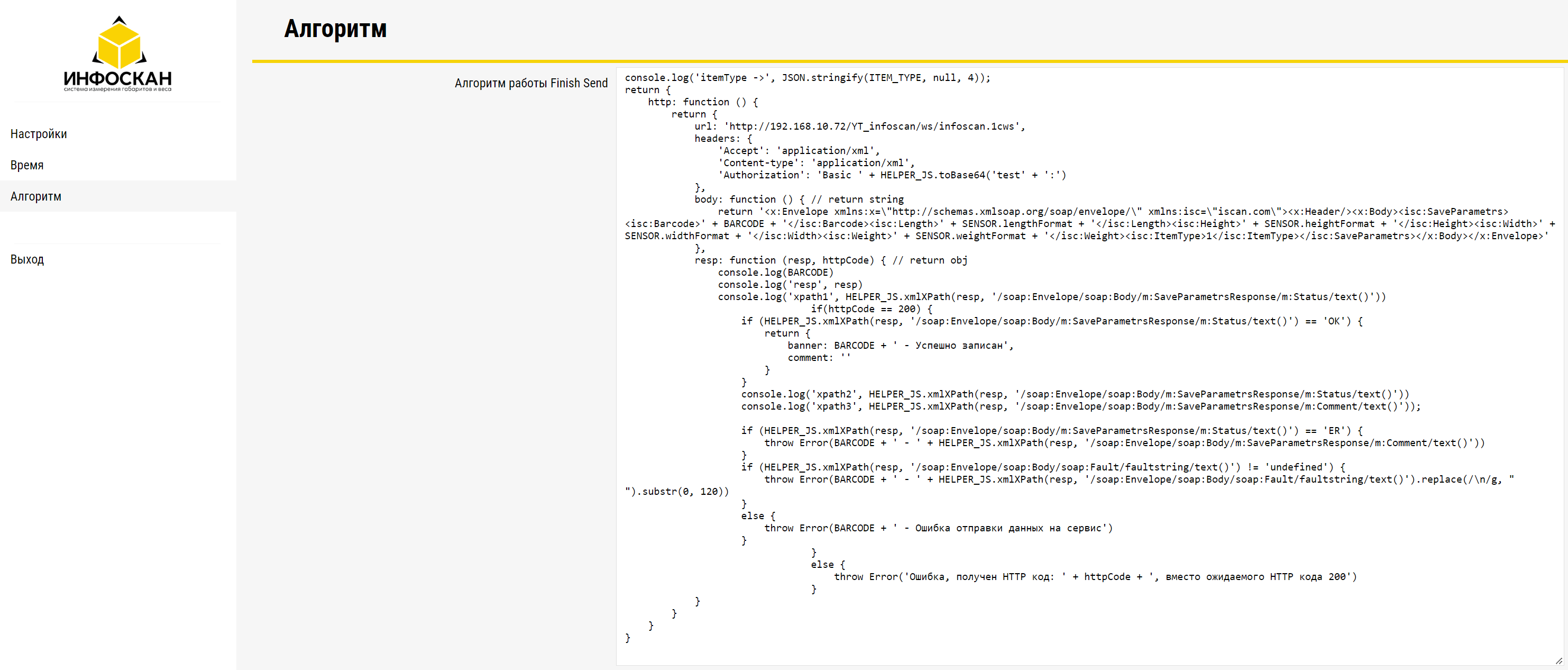
Если в компании учет ведется таким образом что ВГХ привязаны к упаковкам, но штрихкоды не привязаны к упаковкам, т.е. у коробки и штуки один и тот же штрихкод, то необходимо настроить соответствие символов обозначающих упаковки которые прилетают от устройства Инфоскан упаковкам в 1С. Ид упаковки нужно посмотреть в настройках устройства его веб-интерфейсе.



Если после включения устройство оказалось не настроено, то необходимо заполнить алгоритмы работы



|  |
| --- |
| Алгоритм работы Http Handler |
| return {  handlerDataResp: function () {  try {  if(SENSOR == null) {  return {  width: '0',  height: '0',  length: '0',  weight: '0',  barcode: '',  devCode: DEV\_CODE  }  }  return {  width: SENSOR.widthFormat,  height: SENSOR.heightFormat,  length: SENSOR.lengthFormat,  weight: SENSOR.weightFormat,  barcode: BARCODE,  devCode: DEV\_CODE  }  }  catch (e) {  console.error(e);  }  }  } |



|  |
| --- |
| Алгоритм работы Finish Send |
| console.log('itemType ->', JSON.stringify(ITEM\_TYPE, null, 4));  return {  http: function () {  return {  url: 'http://192.168.10.72/YT\_infoscan/ws/infoscan.1cws',  headers: {  'Accept': 'application/xml',  'Content-type': 'application/xml',  'Authorization': 'Basic ' + HELPER\_JS.toBase64('test' + ':')  },  body: function () { // return string  return '<x:Envelope xmlns:x=\"http://schemas.xmlsoap.org/soap/envelope/\" xmlns:isc=\"iscan.com\"><x:Header/><x:Body><isc:SaveParametrs><isc:Barcode>' + BARCODE + '</isc:Barcode><isc:Length>' + SENSOR.lengthFormat + '</isc:Length><isc:Height>' + SENSOR.heightFormat + '</isc:Height><isc:Width>' + SENSOR.widthFormat + '</isc:Width><isc:Weight>' + SENSOR.weightFormat + '</isc:Weight><isc:ItemType>1</isc:ItemType></isc:SaveParametrs></x:Body></x:Envelope>'  },  resp: function (resp, httpCode) { // return obj  console.log(BARCODE)  console.log('resp', resp)  console.log('xpath1', HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Status/text()'))  if(httpCode == 200) {  if (HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Status/text()') == 'OK') {  return {  banner: BARCODE + ' - Успешно записан',  comment: ''  }  }  console.log('xpath2', HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Status/text()'))  console.log('xpath3', HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Comment/text()'));    if (HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Status/text()') == 'ER') {  throw Error(BARCODE + ' - ' + HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/m:SaveParametrsResponse/m:Comment/text()'))  }  if (HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/soap:Fault/faultstring/text()') != 'undefined') {  throw Error(BARCODE + ' - ' + HELPER\_JS.xmlXPath(resp, '/soap:Envelope/soap:Body/soap:Fault/faultstring/text()').replace(/\n/g, " ").substr(0, 120))  }  else {  throw Error(BARCODE + ' - Ошибка отправки данных на сервис')  }  }  else {  throw Error('Ошибка, получен HTTP код: ' + httpCode + ', вместо ожидаемого HTTP кода 200')  }  }  }  }  } |

Успехов Вам и вашему делу!