

## Терминология

Нека се различават два основни типа продукти: **реални и виртуални**.

Към **реалните продукти** спадат:

- I. **Детайли (елементи)** – това са най-малките елементи описани в склада, които не могат да се разпадат.
- II. **Продуктите (или сглобки)** се състоят най-малко от два детайла и/или продукти, като няма възможност да съдържат сами себе си.

*Продуктите - сглобки са предварително асемблирани. За всеки продукт – сглобка се съхранява информация дали може да бъде разкомплектован.*

В склада имаме описани само **детайлите и продуктите - сглобки**.

В следствие на възможността продуктите - сглобки да бъдат разкомплектовани до съставните им части, в таблицата Склад **наличното количество** може да се описва със следните видове количества:

- **количество в сглобки**, т.е. колко броя от съответния детайл или продукт се намират окомплектовани (сглобени) в определени продукти.
- **самостоятелно количество**, т.е. отделните бройки от детайла
- **общо количество**

$\text{колич. в сглобки} + \text{самостоятелно количество} = \text{общо количество}$

Разкомплектоването на даден продукт се извършва чрез специална форма и специален приемно-предавателен протокол. След попълването на формата за разкомплектоване и отпечатване на ППП, системата автоматично извършва следните действия:

1. Продуктът се изписва от склада след попълване на приемно – предавателен протокол, като се посочва съответното основание за изписването на продукта.

*При изписването на продукта от склада се намалява **количество в сглобки** за всеки от детайлите и подпродуктите в продукта, със съответния брой, с който участват в продукта.*

2. Неговите съставни части (детайли) се вписват в склада, отново след попълване на приемно – предавателен протокол.

*При последващото вписването на детайлите, съставлящи продукта, се увеличава **единичното количество** за всеки от тях, отново със съответния брой, с който участват в продукта.*

*Движението на стоки в склада може да се осъществява **единствено** при наличие на съответния приемно – предавателен протокол.*

*Информацията, която трябва да се съхранява за всеки реален продукт:*

<b>№</b>	<b>Име на поле</b>	<b>Размер</b>	<b>Описание</b>
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Име на продукт	128 (nvarchar)	Записва се името на продукта
3	Маска по подразбиране	1 (bit)	Дали да бъде използвана маска по подразбиране за каталожния номер на продукта. Маската на кат. номер е виртуална, реално в базата кат. номер е записан без маска и тя само се визуализира.
4	Каталожен номер	32(char)	Каталожният номер на продукта е уникален.
5	Вън от употреба	1 (bit)	Информация дали продукта е вън от употреба. Ако да, не може да участва в комплектите. По подразбиране false.
6	Сглобка	1 (bit)	Съдържа информация дали реалният продукт е сглобка. По подразбиране false.
7	Разкомплектоване	1(bit)	Указва дали продукта – сглобка може да се разкомплектова. Може да бъде true, само ако и Сглобка е true. По подразбиране false.
8	Производител	16 (binary)	Идентификатор на производителя.
9	Единична цена	8 (double)	Указва единичната цена.
10	Количество за цена	4 (int)	Количеството, съответстващо на единичната цена. Цяло, неотрицателно число.
11	Мерни единици	4 (int)	Указват се мерните единици към количеството.
12	Количество в опаковка	4 (int)	Какво е количеството в една опаковка.

№	Име на поле	Размер	Описание
13	Минимално количество	4 (int)	Какво е минималното количество от съответния продукт, което трябва да присъства в склада. Ако наличното количество е под минималното... съответни мерки.
14	Следи за мин. количество	1(bit)	Дали съответния продукт да бъде следен за мин. количество.
15	Бележки	1024 (nvarchar)	

Име на продукт

☐ Без маска

Каталожен номер

Производител

Сглобка

☐ Продукт – Сглобка

☐ Може да се разкомплектова

Създай схема

Количество

В опаковка

Минимално

☐ Следи за мин. кол.

Цена

Ед. цена

За количество

Бележки

Потвърди

Откажи

При поставяне на отметка в полето **Без маска**, каталожният номер на съответния продукт няма да използва маската по подразбиране. В действителност каталожните номера ще бъдат записвани в базата без маската по подразбиране, като в полето **Маска по подразбиране** се отбелязва дали се използва такава. Маската просто се визуализира по подходящ начин в текстовите полета.

Ако описваният продукт е **сглобка**, се поставя отметка в полето **Продукт – сглобка**. Ако продукта може да се разкомплектова и поставим отметка в полето **Може да се разкомплектова**, бутонът **Създай схема** става активен. Чрез него се създава виртуален продукт, чиято схема описва начина на разкомплектоване на

сглобката.

## **Виртуални продукти**

**Виртуалните продукти не са реално съществуващи продукти.**

Виртуален продукт е продукт, който обединява няколко (повече от един) реални или виртуални продукта. Най-простият виртуален продукт се състои само от реални продукти (повече от един).

Всеки виртуален продукт съдържа в себе си схема (алгоритъм), която описва условията, при които ще бъде включен всеки един от поделементите му. Подразбиращата се схема е **безусловен избор на елемент**, според която всички описани поделементи се включват към виртуалния продукт.

Нека  $vp_1, vp_2, \dots, vp_n$  са вече дефинирани виртуални продукти.

Нов виртуален продукт може да се дефинира като се обединят реални и **вече** дефинирани виртуални продукти, или само описани виртуални продукти.

Тази структура може да се описва като дърво с едно ниво на йерархия.

Дървото се строи отдолу нагоре, като в основата са реалните продукти описани в склада.

В зависимост от комбинациите на описаните поделементи (реални или виртуални продукти) като **резултат** от виртуалния продукт се получават различни конфигурации. Комбинирането на описаните поделементи се осъществява на базата на схемата (алгоритъма), която е избрана за съответния виртуален продукт.

**Резултатът от виртуалния продукт съдържа само реални продукти.**

Даден виртуален продукт може да има различни **версии**. Въвеждането на понятието версия се налага поради следната причина:

Нека резултат от даден виртуален продукт е използван в някой документ. В последствие промяната на схемата на виртуалния продукт е невъзможна.

Ако искаме в бъдеще да се използва друга схема, се създава нова версия на вече съществуващия продукт, а не нов виртуален продукт.

За да се предотврати повторното добавяне на елементите, които няма да бъдат променяни може да се добави бутон (съответно клавишна комбинация), който да копира тези елементи. Ако елементите, които трябва да бъдат променени са виртуални продукти, трябва да бъдат създадени нови техни версии.

Каталожния номер на виртуалния продукт ще се формира от две части: **основен каталожен номер и версия**. Дължината на версията е 3 цифри. Т.е. могат да се създадат максимум 999 версии към даден виртуален продукт.

Каталожният номер на всеки резултат на описан виртуален продукт се състои от три части: **основния каталожен номер, версията и поредния номер на**

резултата.

Поредния номер на резултата е с дължина 5 цифри.

*Коя от всички версии на даден виртуален продукт се ползват и е актуална?*

Използва се последната добавена версия, като останалите се поддържат, за да няма нарушение на вече записаните данни.

За всяка версия на виртуалния продукт може да бъде добавена схема или алгоритъм, по който да бъдат комплектовани съставните му елементи.

Схемата(Алгоритъмът) описва условията, при които към дадения виртуален продукт ще бъде включен всеки един от поделементите му.

На всяка версия съответства **точно** една схема за избор на елементите.

За всеки продукт – сглобка, който може да се разкомплектова трябва да бъде описана схема, по която това да бъде осъществено. Към този **реален продукт** може да бъде **асоцииран виртуален**, описващ съставящите го елементи и начина им на асемблиране. Връзката между реалния продукт-сглобка и описаният към него виртуален продукт се описва или в допълнителна таблица или може да се добави допълнително поле в таблицата за виртуалните продукти (например полето **Схема към продукт-сглобка**). На всеки реален продукт може да съответства не повече от един виртуален.

Даден виртуален продукт има достъп до наследниците на своите наследници единствено чрез децата си.

Даден виртуален продукт може да има внуци, посредством собствените си деца, но не по друг начин.

Реалните продукти не могат да имат деца.

За описанието на **Виртуален продукт** могат да се използват следните данни:  
(Сега не се сецам за други...)

№	Име на поле	Размер	Описание
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Име на продукт	128 (nvarchar)	Записва се името на продукта
3	Основен каталож. номер	24 (nvarchar)	Основния каталожен номер е еднакъм за всички версии към даден продукт.

№	Име на поле	Размер	Описание
4	Схема за продукт-сглобка	16 (binary)	Номер на продукта сглобка, за който виртуаления продукт е описание на схемата, по която се разкомплектова даден продукт-сглобка. Ако това поле е null, то виртуалния продукт не описва схемата на разкомплектоване на никой реален продукт – сглобка.
5	Бележка	1024 (nvarchar)	

При създаването на даден виртуален продукт, може да бъде построено дървото (като се пренебрегнат изискванията за едно ниво на йерархия). Създаденият виртуален продукт е валиден, когато е листата на дървото има само реални елементи. В противен случай, ако в поне един лист има виртуален продукт, то следва че имаме невалиден виртуален продукт.

Даден виртуален продукт е валиден, ако се състои само от реални продукти или всички виртуални продукти в състава му са валидни.

В крайна сметка идеята на описаните виртуални продукти е да се използват при създаването на каталозите и техните страници. Офертите, поръчките и заявките се основават на тези каталози.

Описание на версиите към виртуалните продукти:

№	Име на поле	Размер	Описание
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Виртуален продукт ID	16 (binary)	Номера на виртуалния продукт, за който се създава нова версия.
3	Версия	3(char)	Съдържа поредната версия на вирт. продукт. Полето може да се попълва автоматично, като се проверят колко са версиите за основния каталожен номер в базата.
4	Алгоритъм (схема)	16 (binary)	Схемата по подразбиране за даден виртуален продукт е <b>безусловен избор</b> .

№	Име на поле	Размер	Описание
5	Бележка	1024 (nvarchar)	

Броят версии на даден виртуален продукт може да се съхранява в таблицата Допълнителна информация.

При добавяне на нова версия към съществуващ виртуален продукт, от таблицата Допълнителна информация се изтегля текущия брой на версията към съответния виртуален продукт. Като номер на новата версия се записва текущият брой + 1. Промяната се актуализира и в таблицата Доп. информация, като стойността на полето **Брой версии** за съответния виртуален продукт се увеличава с 1.

На всяка версия на даден виртуален продукт съответства една – единствена схема. Елементите на схемата, която е избрана за съответната версия се записват в таблица, съдържаща следната информация:

№	Име на поле	Размер	Описание
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Идентификатор-родител	16 (binary)	Идентификатор на версията на виртуалния продукт, към който трябва да се включи описания елемент.
3	Реален продукт	1 (bit)	Дали включеният елемент е реален или виртуален продукт.
4	Идентификатор на прилежащия елемент	16 (binary)	Номера на елемента, които ще бъде включен в състава на комплект или друг виртуален продукт.
5	Етикет	128 (nvarchar)	Когато е виртуален продукт, съвпада с името му. Ако е реален продукт, то в това поле се дава по-описателна информация за него.
6	Количество	8 (double)	Количество, с което ще участва в комплектската.
7	Бележки	1024 (nvarchar)	

Полетата Идентификатор – родител, Реален елемент? и Идентификатор на прилежащия елемент уникално идентифицират даден ред.

/Дори е възможно без Реален елемент?, защото идентификаторите са глобални./

**Резултатите** от даден виртуален продукт се записват в базата при тяхното генериране, като следствие от изискванията на потребителя.

Стъпките на генериране на резултата от виртуалния продукт могат да се съхраняват във временна таблица или някаква колекция!

Проверява се дали генерираната конфигурация съответства на друга, вече съществуваща и описана.

Ако е така, за получения резултат не се генерира нов каталожен номер, а се връща вече съществуващия.

В противен случай генерираният резултат се записва в таблицата и получава нов каталожен номер.

Каталожният номер на резултата се състои от каталожния номер на виртуалния продукт, плюс поредния номер на резултата.

Броят на записаните резултати към даден виртуален продукт се отбелязва в таблицата Допълнителна информация. При записване на нов резултат към даден виртуален продукт, увеличаваме и съответния брой резултати. Следователно каталожния номер на резултата може да се генерира автоматично, като за пореден номер на резултата се записва броя от таблицата Допълнителна информация + 1.

Информацията, която ще се съхранява в таблицата **Резултати за виртуален продукт** е следната:

№	Име на поле	Размер	Описание
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Виртуален продукт ID	16 (binary)	Записва се номер на виртуалния продукт, от който е получен резултата.
3	Пореден номер	5 (char)	Записва се поредния номер на резултата.

Първичен ключ в таблицата е полето Идентификатор. Полетата Виртуален продукт и пореден номер идентифицират уникално съответния запис.

В резултата от даден виртуален продукт се включват само реални продукти.

За да се опише конфигурацията на даден резултат се използва таблицата **Съдържание на резултат**:

№	Име на поле	Размер	Описание
1	Резултат ID	16 (binary)	Записва се номер на резултата



№	Име на поле	Размер	Описание
2	Елемент от схемата ID	16 (binary)	Отбелязваме към кой елемент от схемата се отнася съответния запис. Елементът от схемата, който реферираме съответства на реален продукт. От него могат да се проследят (чрез полето <i>parent_ID</i> ) съответните стъпки, които е направил потребителя при избора си.
3	Количество	8(double)	Отбелязва се количеството, с което съответния реален продукт участва в състава на виртуалния. Това е общото количество, с което участва съответния продукт.

*Полетата Резултат ID и Елемент от схема ID идентифицират уникално записа.*

*Нека даден виртуален обект е включен към схема с количество  $x$ . Елементите  $e1, \dots, e_n$ , които се включват в него участват съответно с количества  $k1, \dots, k_n$ . Когато бъде оформена крайната комплектка  $e1, \dots, e_n$  ще бъдат включени с количества съответно  $x \cdot e1, \dots, x \cdot e_n$ .*

*За да се определи дали даден виртуален продукт е в наличност, трябва да бъдат проверени наличните количества на всичките му прилежащи елементи. Ако виртуалният продукт е съставен само от реални продукти, той е в наличност, ако в склада наличните количества за прилежащите му реални елементи са по-големи от 0.*

*Ако липсва наличност за даден продукт, при добавянето му в определен документ, той ще бъде оцветен в червено.*

*Когато се избира какъв елемент да бъде добавен, ако той е виртуален първо трябва да бъде записан в таблицата за виртуалните продукти като се опише неговата схема. След като се получи неговият идентификатор, се добавя запис в таблицата за продуктите към даден комплект.*

*Към даден реален продукт в състава на виртуален, не може да бъде добавяна схема.*

*Към виртуалните продукти и по конкретно към алгоритмите, описани за тях могат да се добавят и различни параметри. Например при описване на проверката по височина или ширина се добавят съответно параметър височина или ширина. Таблицата Параметри за Схема съдържа полетата:*

<b>№</b>	<b>Име на поле</b>	<b>Размер</b>	<b>Описание</b>
1	Идентификатор	16 (binary)	Генерира се автоматично уникален идентификатор.
2	Идентификатор-родител	16 (binary)	Номера на виртуалния продукт, към който ще бъде добавен параметър.
3	Име на параметър	128(nvarchar)	Попълваме името на съответния параметър.
4	Описание	128(nvarchar)	Добавя се забележка или описание към параметър.
5	Тип на параметър	16 (binary)	Възможностите са десетична дроб, цяло число и текст.
6	Минимална стойност	32(nvarchar)	Минималната стойност, която може да приеме параметър.
7	Максимална стойност	32(nvarchar)	Максималната стойност, която може да приеме параметър.

При въвеждането на даден поделемент към описвания виртуален продукт, трябва да се посочат горната и долната граница, между които може да попада стойността на параметър.

Формата за преглед на всички виртуални продукти, за добавяне или изтриване на такива може да е следната:

Виртуален продукт

**Детайли**

Име

Кат. номер

Бележки

**Версии на продукт**

Версия	Схема
1	

Добави Промени Изтрий

**Елементи, участващи в схемата**

Етикет	Реален	Каталожен номер	Количество
1	<input type="checkbox"/>		

Нов

Изтрий

Запази

Потвърди Откажи

При създаване на нов виртуален продукт се попълват полетата *Име* и *Кат. номер*. Полето *Бележки* не е задължително за попълване.

Таблицата **Версии** съдържа всички създадени до момента версии на съответния продукт. Информацията, която се визуализира е *Номер на версия*, *Схема* (или алгоритъм) за избор на елементи и *Бележки*.

При избор на версия, в таблицата **Елементи в схемата** се зарежда списъкът от всички елементи – виртуални или реални, които са добавени към схемата.

Ако искаме да добавим нова версия към виртуалния продукт като натискаме бутона *Добави* на таблицата *Версии*. Появява се следната форма:

Намерът на версията се генерира автоматично. Към схемата на версията могат да се добавят различен брой параметри (таблицата **Параметри**).

Информацията посочена в таблицата е *Име на параметъра*, *Тип* и *Бележки*.

В последствие стойностите на въведените параметрите трябва да се определят за всеки елемент, участващ в описанието на схемата.

Версия

Версия на виртуален продукт

Версия

Алгоритъм

Бележки

Параметри

	Параметър	Тип	
1			

Добави Промени Изтрий

Елементи, участващи в схемата

	Етикет	Реален	Каталожен номер	Количество
1		<input type="checkbox"/>		

Добави Промени Изтрий

Потвърди Откажи

В таблицата 'Елементи, участващи в схема' описваме всички елементи, които ще бъдат включени в схемата.

Бутонът **Промени** не е активен, поради невъзможността схемата да бъде променяна директно. За да бъде променено нещо по схемата, трябва да бъде създадена нова версия.

При натискане на бутона **Добави** се появява следния прозорец:

Добави

Добави

☒ Реален продукт

☐ Виртуален продукт

Потвърди Откажи

Избира се дали да бъде добавен реален или виртуален продукт.

Когато се добавя реален продукт, за него не се отбелязва схема, по която да бъде добавен. Подразбира се безусловен избор.

Избира се каталожния номер на реалния продукт, който ще бъде добавен към схемата.

В полето **Етикет** попълваме информацията, която ще бъде визуализирана на потребителя при оформянето на комплектската.

Определя се количеството, с което избрания елемент ще участва в комплектската.

В таблицата '**Стойности за параметри**' се съдържа списъкът от параметри, зададени за родителския запис. Тук трябва да бъдат определени минималната и максималната стойност, между които трябва да се движат стойностите на параметъра.

Бутоните **Добави** и **Изтрий** към таблицата '**Стойности за параметри**' не е активен. Не трябва да има възможност да се добавят или изтриват допълнителни параметри. Активен е единствено бутона **Промени**, така могат да се задават граничните стойности на параметъра.

Когато искаме да бъде добавен виртуален елемент, формата е следната:

**Добавяне на виртуален продукт към схема**

Каталожен номер

Етикет

Количество

**Стойности за параметри**

	Параметър	Мин. ст-т	Макс. ст-т	
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Има добавен единствено бутон '**Създай**'.

Падащото меню Каталожен номер в случая съдържа каталожните номера на всички записани виртуални продукти. Ако липсва описан определен виртуален продукт, той може да бъде създаден посредством бутона '**Създай**'.

При натискане на бутона се появява формата за създаване на нов виртуален продукт:

**Виртуален продукт**

Име

Каталожен номер

Бележки

След потвърждаване на информацията се появява формата за създаване на нова версия и се описва схемата на виртуалния продукт.

Новосъздадения виртуален продукт ще бъде добавен веднага към падащото мен.

Съществуват предварително зададени схеми за асемблиране, наречени **Първични алгоритми**.

Всички **първични алгоритми** наследяват абстрактния клас *AbstractAlgorithm*.

Основните методи в класа са:

- **void loadProductAlgorithms...**

Като параметър на метода се подава идентификатора на комплекта, който ще бъде асемблиран. Попълва се колекция със съответните елементи, участващи в образуването на комплекта.

- **abstract** колекция **processProductAlgorithms ...**

Методът е абстрактен и се пренаписва във всеки наследник. Като входен параметър се подава колекция от елементите, които трябва да бъдат обработени по някакъв начин (това са елементите, участващи в образуването на комплекта).

Видове алгоритми:

1. **Избор от списък** (*UserSelectAlgorithm*) – изборът се прави на базата на списък с генерирани възможни елементи (детайли или продукти) и потребителят трябва да определи, кои от тях ще бъдат включени в окомплектовката.

Алгоритъмът има 3 разновидности:

- **Избор на 0 или повече редове**, т.е. не е задължително да бъде избрана някаква възможност, както не е задължително да бъде само една.
- **Избор на задължително една от възможностите**, т.е. трябва да бъде избрана точно една от изброените възможности (*ObligatoryUserSigleSelectAlgorithm*)
- **Избор на възможности, като техният брой е зададен в някакъв интервал [min, max]**. (*ObligatoryUserMultipleSelectAlgorithm*)  
Например: Изберете от 2 до 4 възможности от... следните...

#### ***UserSelectAlgorithm extends AbstractAlgorithm***

- Метода **processProductAlgorithms** описва конкретния начин на обработване на елементите. Първо потребителят трябва да направи избор из списък от възможности, след което обработката на елементите продължава. От подадената като параметър колекция *productAlgorithms* се изтриват елементите, които не са избрани от потребителя.
- Въвежда се горна граница на броя на елементите, които алгоритъма може да обработи (**getMaxSelectionLength**). Трябва също да се определи и най-малко колко възможности трябва да избере потребителя, преди да може да продължи (**getMinSelectionLength**).

#### ***ObligatoryMultipleUserSelect Algorithm extends UserSelectAlgorithm***

- Определя се, че трябва да бъде избран поне един елемент (пренаписва се метода **getMinSelectionLength()**), като ограничение отгоре не се прави, т.е. могат да бъдат избрани всички възможни елементи.

#### ***ObligatoryUserSingleSelectAlgorithm extends ObligatoryMultipleUserSelect***

- в **getMaxSelectionLength()** се определя, че трябва бъде избран точно един елемент
- в **public ListSelectionModel getSelectionModel()** се определя, че може да се може да бъде избран един единствен ред от списъка

2. **Безусловно включване на елементи** – безусловно включва всички елементи в окомплектовката, без ограничения.

*UnconditionalIncludeAlgorithm extends AbstractAlgorithm*

в метода **processProductAlgorithms** се връща цялата колекция, без да се обработва

3. **Алгоритъм за избор, в зависимост от някаква граница (Избор на число от диапазон).**

*NumericSpanSelectAlgorithm extends AbstractAlgorithm*

Алгоритъмът има следния смисъл:

*Ако бъде намерена една единствена стойност, тя се включва в окомплектовката автоматично, без запитване или предупреждение.*

*Ако не бъде намерена стойност, която да удовлетворява условията, то на потребителят ще да бъде дадена възможност да избере измежду всички възможности, с които е правено сравнението.*

*Ако бъде намерен списък от възможности, които да удовлетворяват условието, то отново потребителя има възможността да избере измежду тях.*

Алгоритъм за избор в зависимост от някаква граница е например проверката по широчина или височина.

Когато искаме да комбинираме едновременно проверка по височина и широчина, е добре проверката да става на два паса.

Първо да се осъществи проверката по височина например. Намира се съответния интервал или група от интервали, в които попада съответната стойност. Следва проверката в широчина.

Ако при първата проверка не са намерени съвпадения, се предоставя пълния списък с възможности за избор. След като вече е направен избор за височината, се преминава към проверка на широчината.

4. **Алгоритъм по съвпадение.**

При този алгоритъм имаме точно фиксирани варианти за избор.

*Ако клиентското изискване не съвпада с никоя от възможностите, то се появява списък с всички възможности и потребителят трябва да избере една от тях.*

*Ако е намерено съвпадение, то се включва автоматично в окомплектовката без да има съобщение.*

Например: Избор на ориентация, ориентацията може да бъде лява или дясна...



Всеки описан комплект може да се възприема като окомплектовъчен алгоритъм, потребителски дефиниран, създаден на базата на предварително зададените **Първични алгоритми**.

**В дадена Оферта могат да бъдат включени както отделни детайли и продукти, така и комплекти. За всеки от комплектите трябва да се съхранява информация за използвания окомплектовъчен алгоритъм и за отговорите, които са генерирани по време на тази комплектовка. Отговорите могат да бъдат както автоматично генерирани от алгоритъма, така и получени в резултат на запитване към потребителя.**

Обработката на алгоритмите трябва да се реализира независимо от графичния интерфейс, чрез генериране на определени събития. Очаква се те да върнат някакъв резултат след обработка.

### Двуплоскостно отваряне тип АК 80 - 130

Съдържа следните детайли (изброени са само детайлите, които използват различни алгоритми в процеса на комплектоване):

#### **1. Комплект JET АК 80**

- Различни възможности за основен цвят, описани в таблицата с екстрите – Бял, Кафяв, Сребрист.

#### **2. Beutel Ferschluss**

- Описани са три възможни спецификации – 10/14, 12/15, 14/18.

#### **3. Karton Schere JET АК 80**

- Използва се проверка по височина, за да бъдат описани четирите възможности; съответно интервалите са [355 – 600], [601 - 1000], [1001 - 1400], [1401 – 1700]

За описанието на същия комплект стъпките могат да бъдат следните:

Създава се виртуален продукт с име **Двуплоскостно отваряне тип АК 80 – 130**.

(Ако този виртуален продукт се създава при описанието на страница на каталог, то това ще бъде описано в полето Основен виртуален продукт на таблицата Виртуални продукти.)

Задава се каталожен номер на продукта. Версията му се записва автоматично, след записването на продукта в базата, като се взема в предвид броя на версиите създадени към съответния продукт. Ако това е първата версия, в полето ще бъде записано 001. За схема на съответния продукт се установява **безусловен избор** или не се избира схема.

Към списъка от продукти, участващи в конфигурирането му трябва да се добавят:

#### **1. Комплект JET АК 80**

Съществуват три възможности за избор на Комплект JET АК 80, в зависимост от основния цвят. За да бъдат описани тези възможности се създава **нов виртуален продукт** с име JET АК 80.

Добавя се каталожен номер и за съответната версия се отбелязва, че ще бъде използвана схемата **Избор на точно 1 (Потребителски избор на точно един елемент)**.

Към този виртуален продукт JET АК 80 се добавят три реални продукта, съответстващи на трите възможности за основен цвят, като за етикет на всеки от продуктите може да се посочи съответния цвят.

#### **2. Beutel Ferschluss**

По аналогичен начин на избор на цвят.

Съществуват три възможности, в зависимост от спецификацията. Към корена се добавя нов виртуален продукт (нека го наречем **Beutel Ferschluss**) и се определя, че използваната схема за избор на елемент ще бъде **Избор на точно 1 (Потребителски избор на точно един елемент)**.

Нека този виртуален продукт обединява три реални продукта, описващи съответно трите възможни спецификации. Добавяме тези три реални продукта към виртуалния, като на всеки от тях се поставя етикет със спецификацията.

**Обобщение:**

Изборът на цвят, спецификация и ориентация се реализира с помощта на Потребителски избор на точно един (избор на един).

**3. Karton Schere JET AK 80**

При избора на подходящия продукт, ръководна роля има височината. Различават се 4 реални продукта, в зависимост от височината. Тези възможности могат да се обединят в един виртуален продукт (Karton Schere JET AK 80). За схема на съответната версия на продукта се използва **Избор на число от диапазон (Диапазон число)**.

Към алгоритъма могат да бъдат зададени един или няколко параметъра, по които да се прави сравнението. В случая за параметър на сравнение се задава височина.

Към така избраната схема могат да бъдат добавени като реални, така и виртуални продукти. За всеки от елементите обаче, трябва да бъдат въведени съответно минимална и максимална стойност на параметрите.

В случая могат да се добавят 3 реални продукта. За всеки от тях се определят границите на параметрите.

<b>Елемент</b>	<b>Минимална ст-т</b>	<b>Максимална ст-т</b>
E1	355	600
E2	601	1000
E3	1001	1400
E4	1401	1700

Тези стойности ще бъдат сравнявани с въведените от потребителя за съответния параметър / параметри.

Дървото изглежда така:

**Дуплоскостно отваряне тип АК 80 – 130**

**Виртуален продукт JET AK 80 (Избор на точно един)**

- Етикет БЯЛ  
Реален продукт, чиито каталожен номер съответства на продукта с основен цвят бял.
- Етикет КАФЯВ  
Реален продукт, чиито каталожен номер съответства на продукта с основен цвят кафяв.

- Етикет СРЕБРИСТ

Реален продукт, чиито каталожен номер съответства на продукта с основен цвят сребрист.

**Beutel Ferschluss (Избор на точно един)**

- Етикет 10/14

Добавен е реален продукт, който съответства на указаната спецификация

- Етикет 12/15

- Етикет 14/18

**Karton Schere JET AK 80 (Избор на число от диапазон)**

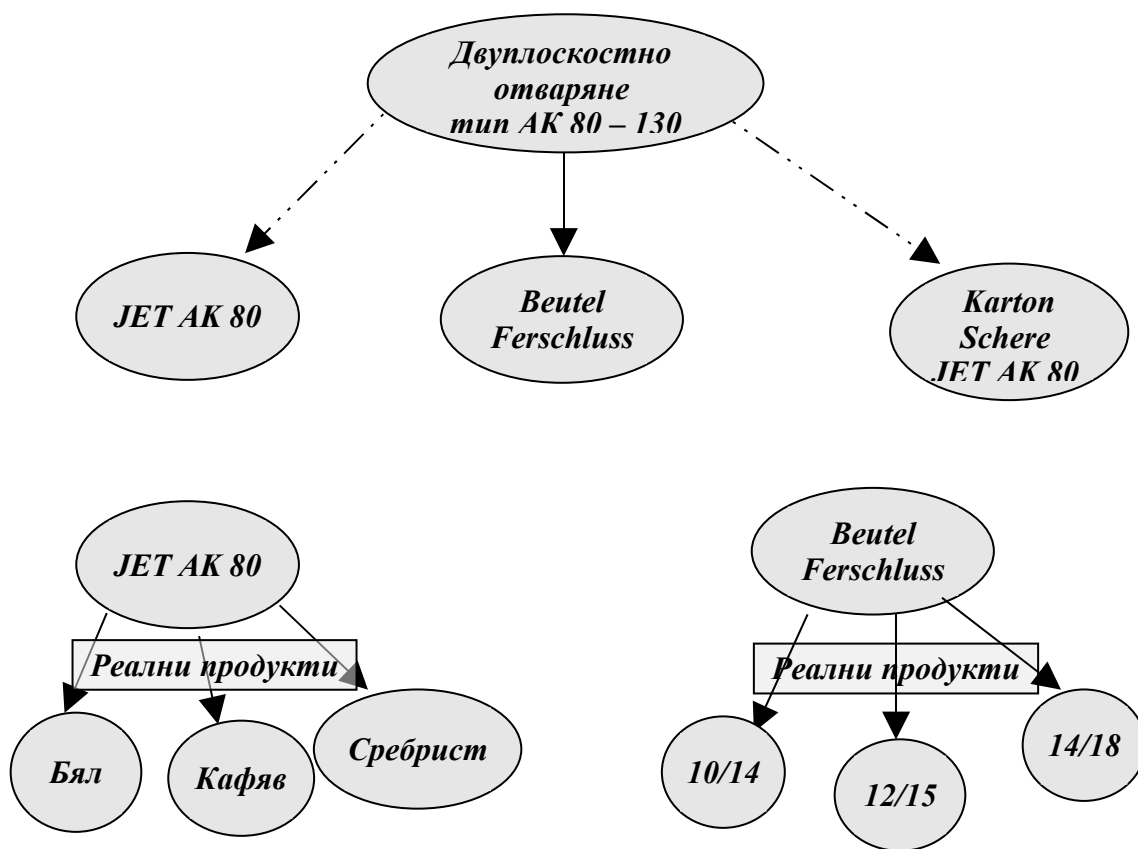
- Етикет 355 – 600

Описан е реалния продукт, който съответства на посочените параметри

- Етикет 601 - 1000

- Етикет 1001– 1400

- Етикет 1401– 1700



Ако искаме да създадем алгоритъм, който да обединява избор на спецификация и проверка по височина, начинът е следния:

Създава се виртуален продукт ВП.

Използваната към него схема е **Избор на точно един**.

Като поделементи на този виртуален продукт могат да бъдат въведени

различните спецификации. Тъй като на всяка спецификация трябва да съответства и проверка по височина, спецификациите ще бъдат създадени като виртуални обекти, към които могат да се опишат съответните проверки за височината.

Създаваме **C1**, първата спецификация, като виртуален обект и алгоритъма, който ѝ съответства е **Избор на число от диапазон**. Като параметър задаваме височина.

Към **C1** могат да бъдат добавени произволен брой реални или виртуални обекти. В случая можем да добавим два реални обекта **E11** и **E12**, за които определяме границите на височината.

Създаваме втората спецификация **C2**, отново като виртуален обект. Следва описание на проверките за височина по аналогичен начин. Като не е необходимо задачите интервали за височината към спецификация **C1** и **C2** да съвпадат.

Как се осъществява комплектоването на такъв продукт?

Първо е зададен избор на спецификация, следователно на потребителя се дава възможност да избере измежду всички описани спецификации. След избор на спецификация, трябва да бъде въведена стойност за параметъра височина. Прави се сравнение в кои от описаните към схемата интервали за височина попада зададената от потребителя стойност.

Ако съпадението е единствено, то съответния елемент  $E_{ij}$  се добавя към комплектовката.

Ако има повече от едно съпадение, потребителят трябва да избере измежду тях.

Ако няма интервал, в който да попада зададената от потребителя стойност, то той може да избере измежду всички описани към дадената схема възможности.

