



eMagazynier — Sugestia odłożenia

Którą lokalizację dodatek podpowiada przy odkładaniu • stała lokalizacja i **Kolejność strefy** + **cechy** algorytmicznie **StrefaOdłożenia** | **SugestiaOdłożenia**

eMagazynier — Sugestia odłożenia

Adresat: **wdrożeniowiec (konfiguracja w enovie)**

Wersja dokumentacji: **2026-09-01**

SPIS TREŚCI

- Po co to jest
- Jak wybiera się lokalizację
 - 2.0 Lokalizacje przypisane
 - 2.1 Reguła dodatku
 - 2.2 Kolejność to nie trasa
- Ustawienia bez kodu
- Cecha StrefaOdłożenia
- Cecha SugestiaOdłożenia
- Gotowe warianty
- Czego cechy nie obejmują
- Pułapki
- Diagnostyka

1. Po co to jest

Magazynier po przyjęciu ma dostać **adres**, a nie decyzję. Sugestia odłożenia to jeden wiersz na ekranie: lokalizacja, do której ma zawieźć towar.

Bez niej odkładanie jest wiedzą w głowie — gdzie zwykle stoi ten towar, gdzie jest miejsce, gdzie akurat nie zajął kierownik. Nowy pracownik tej wiedzy nie ma, a stały nie zawsze jest w pracy. Podpowiedź zamienia tę wiedzę w regułę, którą da się zmienić w jednym miejscu.

Sterowanie jest **trzystopniowe** i po każdym kolejnym stopniu sięgasz dopiero wtedy, gdy poprzedni nie wystarczy:

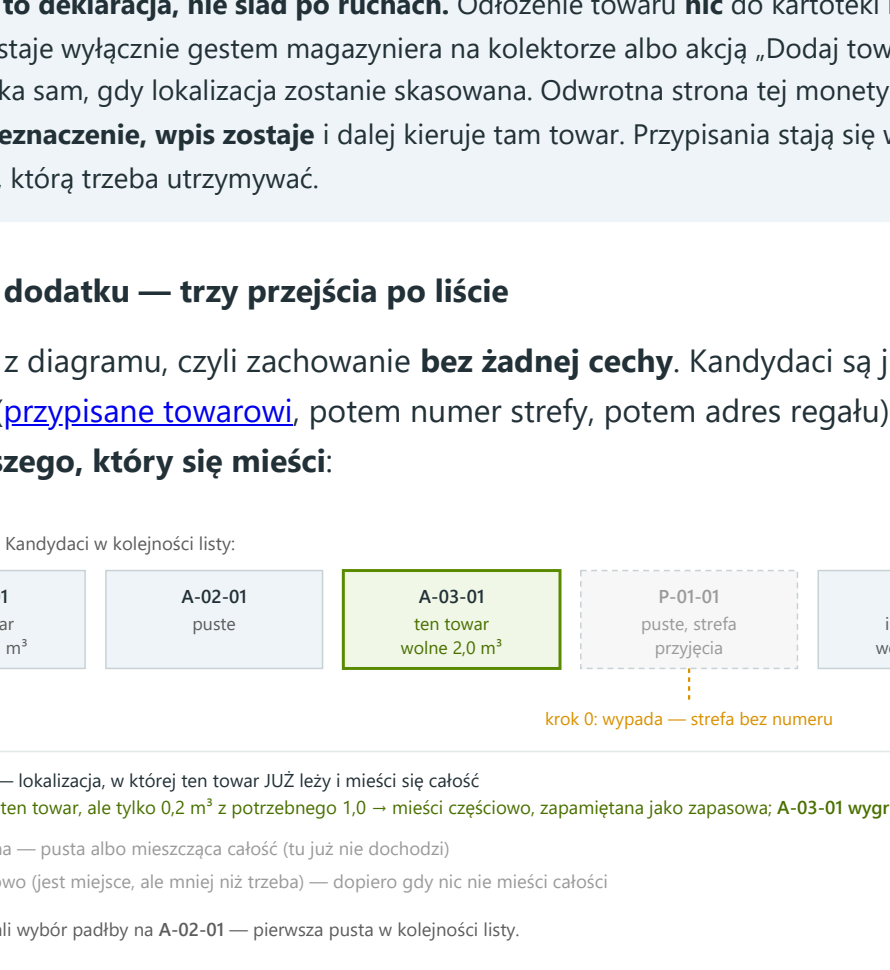
stopień	czym	kiedy sięgasz
1. bez kodu	stała lokalizacja towaru, lokalizację przypisaną , Kolejność strefy , wymiary lokalizacji	zawsze zacznij tutaj — to załatwia większość wdrożeń
2. własny filtr	cecha StrefaOdłożenia — filtr kandydatów po strefie	„odkładany wyłącznie do magazynowania”
3. własny wybór	cecha SugestiaOdłożenia — wskazanie konkretnej lokalizacji	„towary sypkie tylko na dolne półki”

Stopnie 1 i 3 to **cechy algorytmiczne enovy** — kilka linijek kodu, które dodatek woła w trakcie implementacji podpowiedzi. Zakłada je gotowa akcja, opisując je rozdziały [4](#) i [5](#).

Ta strona opisuje **wybór lokalizacji**. Kolejność, w jakiej magazynier obchodzi halę — czyli jak posortowana jest lista odłożenia i kompletacji — to osobny mechanizm: [Trasa przez magazyn](#). Obie rzeczy czytają tę samą **cechę Kolejność strefy** , więc warto znać obie.

2. Jak wybiera się lokalizację

Pięć kroków, zawsze w tej samej kolejności. Pierwszy, który da odpowiedź, kończy sprawę.



Stać lokalizację rozstrzyga PRZED oboma cechami — i to jest świadomy model, nie przeoczenie. Stała jest „ścieżką na skróty” dla asortymentu z przypisanym miejscem, a algorytm rzadzi całą resztą. Chcesz, żeby towar szedł Twoim algorytmem — **nie ustawiaj mu stałej lokalizacji**.

2.0 Lokalizacje przypisane towarowi

Na kartotece towaru przypisujesz lokalizację, a **jedną z nich w każdym magazynie** możesz oznaczyć jako **domyślną**. Te dwie rzeczy działają zupełnie inaczej.

na kartotece	co robi przy odkładaniu
domyślna (stała lokalizacja)	kończy sprawę — towar jedzie tam bez pytania reguły i bez pytania cech
przypisana , bez oznaczenia domyślnej	idzie na początek listy kandydatów — wygrywa wewnątrz każdego kroku reguły, ale reguła dalej decyduje
nieprzypisana	zwykły kandydat — o kolejności decyduje numer strefy, a wewnątrz strefy adres reguły

Przypisanie wyprzedań nawet **nieprzypisanej** przypisana lokalizacja w namiocie (strefa 2) zostanie zaproponowana przed numer strefy w hali (strefa 1). Mówi bowiem o **tym towarze**, a numer strefy tylko o kolejności zapewniania budynku. Numer strefy rozstrzyga dopiero **wewnątrz** każdej z grup — gdy towar ma przypisaną lokalizację i w hali, i w namiocie, **piękną** będzie ta z hali. Tak samo zachowuje się **stała lokalizacja**, która numeru strefy nie ogląda w ogóle.

Filtru stref to nie omija. Przypisana lokalizacja w strefie **bez numeru** wypada tak samo jak każda inna — przypisanie zmienia kolejność, nie skład listy. Ta sama zasada rządzi stałą lokalizacją: strefa bez numeru unieważnia ją bez wyjątku.

Przypisanie NIE przeskaluje konsolidacji — i to jest decyzja, nie przeoczenie. Gdy przypisana lokalizacja jest pusta, a ten sam towar leży już gdzie indziej i tam się mieści, wygrywa **konsolidacja**. Inaczej nowa dostawa rozbiłaby towar na dwie lokalizacje mimo wolnego miejsca w jednej. Czyny dwa przynosi przy kompletacji i dwa miejsca do pomylenia przy inwentaryzacji. Chcesz twardego przypisania — użyj **domyślnej**.

Przypisania to deklaracja, nie ślad po ruchach. Odłożenie towaru **nic** do kartoteki nie dopisuje — wpis powstaje wyłącznie gestem magazyniera na kolektorze albo akcją „Dodaj towary” w enovie, a znika sam, gdy lokalizacja zostanie skasowana. Odrotna strona tej monety: **gdy regał zmienia przeznaczenie, wpis zostaje** i dalej kieruje tam towar. Przypisania stają się więc konfiguracją, którą trzeba utrzymywać.

2.1 Reguła dodatku — trzy przejścia na liście

To jest krok 5 z diagramu, czyli zachowanie **bez żadnej cechy**. Kandydaci są już sortowani ([przypisane towarowi](#)), potem numer strefy, potem adres regału i reguła bierze **pierwszego, który się mieści**:

Potrzeba 10 m³. Kandydaci w kolejności listy:	A-01-01 ten towar wolne 0,2 m³	A-02-01 pusta	A-03-01 ten towar wolne 2,0 m³	P-01-01 pusta, strefa przyjścia	A-04-01 sny towar wolne 3,0 m³

1. konsolidacja — lokalizacja, w której ten towar **JUŻ** leży i **mieści się całość**.
A-01-01 ma ten towar, ale tylko 0,2 m³ z potrzebnego 1,0 — mieści częściowo, zapamiętana jako **zapasowa**; A-03-01 wygrywa

2. pierwsza wolna — pusta albo mieszcząca całość (tu już nie dochodzi)
3. mieści częściowo (jest miejsce, ale mniej niż trzeba) — dopiero gdy nic nie mieści całości

Bez towaru w hali wybór padłby na A-02-01 — pierwsza pusta w kolejności listy.

przejście	bierze
krok 0 — filtr	zostają wyłącznie lokalizacje ze stref składowania (patrz Kolejność strefy)
1. konsolidacja	lokalizacja, w której ten towar już leży i mieści się całość
2. pierwsza wolna	lokalizacja pusta albo taka, w której mieści się całość
3. mieści częściowo	gdy nic nie mieści całości — konsolidacja przed pustą

W każdym z tych przejść **reguła bierze pierwszą** przypisaną lokalizację w kolejności listy — a listę otwierają **lokalizacje przypisane towarowi**. Przypisanie działa więc wewnątrz kroków, nie zamiast nich: przypisana wygra i konsolidację, i pierwszą wolną, ale **pusta przypisana nie wyprzedzi konsolidacji** w innej lokalizacji.

„Mieści częściowo” znaczy: jest tam wolne miejsce, ale mniej, niż wynosi odkładana ilość — towar jedzie tylko w części. Reguła rozróżnia trzy stany i traktuje je zupełnie inaczej:

stan lokalizacji	wolne miejsce	co robi reguła
mieści całość	tyło, ile trzeba, albo więcej — także wtedy, gdy pojemności nikt nie zadeklarował	bierze od razu
mieści częściowo	jakiś jest, ale mniej niż trzeba	zapamiętuje jako zapasową i szuka dalej
pełna	zero albo mniej (przepelniona)	pomija zupełnie — nigdy nie zostanie zaproponowana

Lokalizacja mieszcząca częściowo wraca do gry dopiero w kroku 3, czyli wtedy, gdy w całej hali nic nie mieści całości. Lepiej dać magazynierowi adres i ostrzec o przepelnieniu, niż odpowiedzieć „nie ma miejsca” i zostawić go z paletą na korytarzu.

Sugestia działa ZAWSZE, także gdy hala jest pełna. Krok 3 istnieje po to, żeby magazynier dostał adres komunikatu „nie ma miejsca”. O przepelnieniu ostrzeża dopiero **bramka pojemności** przy zapisie, pytaniem „odłożyć mimo to?”.
Ważne: sugestia działa zawsze, niezależnie od stanu lokalizacji.

Lokalizacja bez zadeklarowanych wymiarów nie ma limitu, a nie „nie ma miejsca”. Pojemność 0 znaczy „nikt nie mierzył”, więc taka lokalizacja przechodzi każdy warunek pojemnościowy. Dopóki nie wprowadzisz wymiarów, cała reguła sprowadza się do „konsolidacja, potem pierwsza pusta”.

2.2 Kolejność kandydatów to nie trasa

„Kolejność listy” znaczy na tej stronie zawsze jedno: **kolejność, w jakiej reguła rozważa kandydatów** — przypisane towarowi, potem numer strefy, potem adres regału. To kolejność **wyboru**, a nie kolejność, w jakiej magazynier obejdzie halę.

co ustala	czym
Dokąd pojedzie towar — jeden adres na pozycję	ta strona
W jakiej kolejności magazynier obejdzie wskazane adresy	Trasa przez magazyn

Cecha TrasaMagazynu nie bierze udziału w wyborze. Klucz trasy liczy się dopiero dla lokalizacji, którą sugestia już wskazała — najpierw wybór, potem kolejność. Choćbyś napisał najbardziej wymyślną trasę, nie zmieni ona ani jednego adresu na liście odłożenia; zmieni tylko to, w jakiej kolejności magazynier je obejdzie.

„Ten towar ma leżeć bliżej rampy” to zadanie dla tej strony, nie dla trasy. Trasa ułoży najkrótszą drogę po adresach, które dostanie — ale ich nie wybiera. Bliżość rampy wyrażasz **numerem strefy** albo cechą **SugestiaOdłożenia** . Numer strefy jest przy tym jedyną daną, którą czytają oba mechanizmy naraz.

3. Ustawienia bez kodu — zacznij stąd

Trzy rzeczy do ustawienia, zanim w ogóle sięgniesz po **cechę** algorytmiczną. U większości klientów to jest całe wdrożenie sugestia.

Stać lokalizację towaru

gdzie	kartoteka towaru — cecha Lokalizacja
zakres	jedna na magazyn — lokalizacja z innego magazynu nie jest sugestią dla tego zasobu
znaczenie	domysły, nie żądanie

Towar ze stałą lokalizacją jedzie tam bez pytania kogokolwiek — ani reguły, ani Twoich cech. Ale gdy się nie mieści, sterowanie wraca do reguły i towar dostaje inną lokalizację; w dzienniku zostaje ślad z nazwą towaru i metrami. Przypisania nikt nie kasuje, więc gdy miejsce się zwolni, stała zadziała znowu sama.

Kolejność strefy

gdzie	element słownika Strefa — cecha Kolejność strefy (liczba całkowita)
kto zakłada	akcja menu Handel → Magazyn → Kolektory → Czynności → WebAPI → Konfiguracja
zakres	per strefa — numerujesz od 1

wartość	co robi z odkładaniem
1, 2, 3...	kolejność zapewniania : strefa 1 pierwsza, do 2 towar trafia dopiero, gdy w 1 nie ma miejsca
pusta / 0	strefa nie jest proponowana — to przystanek procesu (przyjęcie, kontrola, wydanie), nie półka

Numerem strefy ustawia się „zapelnij halę, dopiero potem namiot” — bez pisania choćby linijek. Reguła bierze pierwszego kandydata, który się mieści, w kolejności listy, a listę porządkuje właśnie ten numer. Priorytet zapewniania jest liczbą, nie regułą do napisania.

Dotyczy towaru, który nie ma własnego adresu. Numer strefy porządkuje kandydatów, ale [stała lokalizacja i lokalizacje przypisane](#) stoja przed nim: towar ze stałą w namiocie albo przypisany **wyłącznie** do namiotu pojedzie tam także wtedy, gdy hala jest pusta. Chcesz, żeby dany towar podlegał kolejności zapewniania — nie dawaj mu przypisać w strefie, do której ma trafić dopiero na końcu.

Ten sam numer rządzi też trasą. Kolejność, w jakiej magazynier obchodzi halę przy odkładaniu i kompletacji, liczy się z tej samej cechy — zob. [Trasa przez magazyn](#). Zmieniając numer, zmieniasz obie rzeczy naraz: co się zapienia najpierw i dokąd magazynier idzie najpierw. Zwykle o to właśnie chodzi, ale warto o tym wiedzieć przed zmianą.

W bazie, w której nikt nie numerował strefy, filtr się nie włącza. Cecha liczbowa bez wartości czyta się jako 0 , więc wszystkie strefy miałyby zero i sugestia zniknęłaby w całości — dlatego filtr do pustej listy jest **ignorowany**. Ta sama zasada rządzi cechą **StrefaOdłożenia** .

Wymiary lokalizacji i towarów

Pojemność lokalizacji i objętość towaru decydują o tym, czy „mieści się całość”. Dopóki ich nie ma, reguła nie „mierz” — działa, ale sprowadza się do konsolidacji i pierwszej pustej lokalizacji. To bywa **wystarczające**; wymiarowanie hali zaczyna się opłacać dopiero tam, gdzie realnie brakuje miejsca.

4. Cecha StrefaOdłożenia — własny wybór

Filtruje kandydatów po symbolu strefy. Najczęstsza potrzeba wdrożeniowa brzmi „nie odkładaj do strefy przyjęcia”, a nie „wybierz mi lokalizację”.

gdzie	Narzędzia → Opcje → Systemowe → Definicje cech, tabela CfgNodes , zakładka Algorytm
kto zakłada	akcja menu Magazyn → Kolektory → Czynności → WebAPI → Cechy sugestii odłożenia
typ	Tekst
zakres	globalny — ten sam kod dla każdego magazynu, rozdzieli je po magazyn.Symbol

```
public string Get_Feature_StrefaOdłożenia(  
    Soneta.Towary.Towar towar,  
    Soneta.Magazyny.Magazyn magazyn,  
    System.Collections.Generic.IReadOnlyList<EndIF.Magazyn.Lokalizacje.KandydatGniazda> double potrzeba)
```

zwracasz	znaczy
"M"	wybieraj wyłącznie z lokalizacji strefy M
"M K"	kilka stref — rozdzielasz znakiem
pusty tekst	bez filtra
symbol spoza hali	filtr pominięty — jeden wpis w dzienniku, sugestia liczy się dalej

Zwracając samą strefę, zostawiasz **całą regułę wyboru** dodatkowi — nie musisz jej u siebie otwierać, a odzwrotna rozjechałaby się z produktem przy pierwszej poprawce.

Symbol musi istnieć w TEJ bazie. Strefa, której nie ma wśród kandydatów, jest ignorowana, więc literówka w symbolu wywoła dokładnie tak samo jak działający algorytm — sugestia dalej daje lokalizację, tylko wolny filtru nie ma. Akcja zakładająca cechę wypisuje w komentarzu **strefy odczytane z bazy** wraz z ich numerami; zajrzyj tam, zanim zaczniesz debugować.

5. Cecha SugestiaOdłożenia — własny wybór

Wskazuje konkretną lokalizację i jej **zdanie jest nadrzędne**. Siegasz po nią, gdy potrzebujesz reguły, której nie da się wyrazić strefą.

gdzie	Narzędzia → Opcje → Systemowe → Definicje cech, tabela CfgNodes , zakładka Algorytm
kto zakłada	akcja menu Magazyn → Kolektory → Czynności → WebAPI → Cechy sugestii odłożenia
typ	Liczba całkowita
zakres	globalny — ten sam kod dla każdego magazynu

```
public int Get_Feature_SugestiaOdłożenia(  
    Soneta.Towary.Towar towar,  
    Soneta.Magazyny.Magazyn magazyn,  
    System.Collections.Generic.IReadOnlyList<EndIF.Magazyn.Lokalizacje.KandydatGniazda> double potrzeba)
```

Zwracasz **LokId** jednego z podanych kandydatów. Po założeniu **nic się nie zmienia**: wstawiony algorytm jest kopią reguły dodatku, więc wybór zostaje ten sam. Zmienia się tylko to, kto go liczy — od tej chwili kod z bazy, nie z DLL.

return 0; znaczy „**żadna lokalizacja**”, a **NIE „policz to za mnie”**. Od istnienia tej cechy rządził Twój kod: cokolwiek zwrócił, reguła dodatku nie dolicza się po cichu, bo dwie reguły kładą po kolei to znowu dwie prawdy o tym samym. **Powrót do reguły dodatku = skasowanie definicji cechy**: samo wyczyszczenie ciała metody da brak sugestii dla wszystkich.

Co dostajesz w kandydatach

pole	znaczenie
LokId	to zwracasz
Skrot	adres {rząd}-{pion}-{poziom} , np. F-01-00
Strefa	symbol strefy; pusty, gdy lokalizacja strefy nie ma
KolejnosćStrefy	numer strefy; 0 = nie jest miejscem składowania
Kod	pełny adres {magazyn}-{strefa}-{rząd}-{pion}-{poziom} , np. F-M-A-01-00 ; To po nim idzie kolejność wewnątrz strefy — czyli po regale: rząd, pion, poziom
SaZasoby	czy cokolwiek w gnieździe leży
MaTenTowar	czy leży tu już ten towar — warunek konsolidacji
Przypisana	czy ta lokalizacja jest przypisana temu towarowi na kartotece; takie stoją na początku listy
Objetosć	pojemność w m³; 0 = niezadeklarowana, czyli brak limitu
WołneMiejsce	ile jeszcze wejdzie

Kolejność kandydatów już niesie priorytet — przypisane towarowi, potem numer strefy, potem kod (adres regału) — więc sortowania nie musisz otwierać: wystarczy wziąć **pierwszego pasującego**. Jeżeli cecha **StrefaOdłożenia** też istnieje, kandydaci są tu już po jej filtrze.

Lokalizacja **spoza podanej listy** jest odrzucana: liczy wtedy reguła dodatku, a w dzienniku zostaje ślad. Bez tej bramki dąłoby się wskazać lokalizację, której zapis i tak nie przyjmie — czyli różnicę, co ci magazynier widzi, z tym, co naprawdę się zapisze.

Kolejne wywołanie akcji Konfiguracja nie nadpisze cechy — Twój kod jest nieetykalny także wtedy, gdy ktoś kliknie ją drugi raz. Skutek uboczny: wstawiony algorytm to **kopia, nie podpięcie**. Gdy reguła produktu się poprawi, ta baza zostanie przy wersji, którą ma. Żeby zobaczyć nowe przykłady i nową regułę, skasuj definicję i załóż ponownie.

6. Gotowe warianty

Wszystkie siedzą w komentarzu wstawianym do cechy przy zakładaniu. Poniżej te, po które sięga się najczęściej.

Reklamacja osobno — StrefaOdłożenia

Towar oznaczony cechą idzie do strefy kontroli, reszta do magazynowania:

```
if (Equals(towar.Features["Reklamacja"], true)) return "K";  
return "M";
```

Inny magazyn, inna reguła — StrefaOdłożenia

Cecha jest globalna, więc ten sam kod wykonuje się dla każdego magazynu; rozdzielasz je po symbolu:

```
if (magazyn.Symbol == "SERWIS") return "K";  
return "M";
```

Towary sypkie na dolne poziomy — SugestiaOdłożenia

Worka nie wnosi się na czwartą półkę. Skrot: ma format {rząd}-{pion}-{poziom} , a **poziomy liczą się od 00** — 00 to podłoga:

```
if (Equals(towar.Features["Sypki"], true))  
foreach (var kandydat in kandydaci)  
    if (kandydat.Skrot.EndsWith("-00")) || kandydat.Skrot.EndsWith("-01"))  
        return kandydat.LokId;
```

Wyjątki dopisuj ZARAZ ZA KROKIEM 0, a nie przed nim. Przed filtrem lista kandydaci zawiera jeszcze lokalizacje strefy przyjęcia i wydania, więc Twój foreach mógłby wskazać towarowi przystanek procesu zamiast półki. Gdy żaden wyjątek nie pasuje, pętla nic nie zwraca i sterowanie idzie do reguły niżej — dokładnie tak, jak ma być.

Wyjątek dotyczący samej STREFY róż w StrefaOdłożenia , nie tutaj. Ona filtruje kandydatów wcześniej, więc lokalizacja z difiltrowanej strefy w ogóle nie dojdzie do tego kodu. Sprawdź też, czy zamiast kodu nie wystarczy **numer strefy** — odsunięcie namiotu za halę robi się liczbą.

7. Czego cechy nie obejmują

Dwie dziury w zasięgu, obie świadome. Trzeba je znać, bo inaczej sprawdzasz swój algorytm przyciskiem, którego on nie dotyczy — i nie masz jak się domyślić dlaczego.

Towar ze stałą lokalizacją

Gdy towar ma stałą lokalizację w tym magazynie i się w niej mieści, adres wynika z kartoteki i **żadna z cech nie jest wołana**. Chcesz, żeby szedł algorytmem — zdejmij mu stałą lokalizację.

Odłożenie z wypelnionym parametrem „Rodzaj”

Przyciski „Odłóż pozycję / zasoby / z listy” i „Relokuj” w enovie, a także to samo pole w żądaniu kolektora, przyjmują **rodzaj**: Konsolidacja , Pierwsza dostępna , Stała lokalizacja . Rodzaj jest świadomym żądaniem magazyniera, a cecha algorytmem **podpowiedzi** — więc podanie rodzaju znaczy „tym razem nie podpowiadaj”.

ścieżka	pyta cechy?	filtr stref składowania?	przypisane pierwsze?
kolumna „Sugestia”, „Odłóż wg sugestii”, kolektor	tak	tak	tak
odłożenie z wypelnionym rodzajem	nie	tak	nie
towar ze stałą lokalizacją	nie	tak	nie dotyczy — stała kończy sprawę

Asymetria w kolumnie „filtr stref” jest celowa. Cecha jest **preferencją wdrożenia** („u nas wybiera się inaczej”) i magazynier ma prawo ją pominąć, wskazując rodzaj. Numer strefy jest **faktem o budynku**: strefa przyjęcia i wydania to przystanki procesu, nie półki, i żaden przycisk nie czyni ich półką.

Dlatego strefa bez numeru unieważnia nawet **stałą lokalizację** towaru — a ścieżka rodzaju „Stała lokalizacja” wtedy **odmawia**, z nazwą lokalizacji i strefy, zamiast po cichu podstawia inną. Magazynier nie może widzieć na ekranie A-01-00 , a mieć zapis do B-01-00 ; jedyną dopuszczalną alternatywą jest odmowa, która mówi dlaczego.

8. Pułapki

Zmienilem algorytm i nic się nie stało

Po każdej zmianie ciała algorytmu wymuś rekompilację: Opcje → Systemowe → Kompilator → przycisk „Wymuś rekompilację kodu”.

Skompilowany algorytm siedzi w pamięci **procesu**, więc działające WebAPI liczy dalej starym ciałem, a dziennik **milczy** — nic się przecież nie wydarzyło.

Założenie i skasowanie cechy działa natychmiast. Δ Ale skasowanie i **założenie jej od nowa** z poprawionym kodem — już nie: nazwa ta sama, więc proces bierze ciało ze swojego cache’u.

Liczba argumentów musi zgadzać się co do jednego

FeatureDefinition.CheckArgs porównuje ją z sygnaturą metody, więc **zostaw sygnaturę dokładnie taką, jaką wstawiła akcja**. Skreślenie argumentu, którego nie używasz, tak samo jak podanie miejsc w hali. Odłożenie całej listy **nie zatrzymuje się** na takiej pozycji: reszta idzie na półki, a te pozycje zostają, z komunikatem, który **wymienia towary**.

„Odłożono 3 z 4 pozycji. Reguła odkładania nie wskazała lokalizacji dla: BIKINI. Te pozycje zostają na liście — wskaż im lokalizację ręcznie.”

Komunikat **nie proponuje wyłączenia przepelnienia**, bo ponowienie niczego nie zmieni — lokalizacji nadal nie ma. Pytanie, którego zgoda nie załatwia, uczy klikania na ślepo.

Taka pozycja ładuje na końcu listy odłożenia. Bez adresu nie ma jak umieścić jej w kolejce przejścia hali ([Trasa przez magazyn](#)), więc kolektor stawia ją za wszystkimi, które adres dostały — magazynier wybiera jej lokalizację sam, gdy skończy obchód.

eMagazynier — dokumentacja wdrożeniowa: sugestia odłożenia • powiązana: [Trasa przez magazyn](#) [Odłożenie na kolektorze](#) • [strona produktu](#)