

Карты сопоставления номенклатуры: настройка (для администратора)

Страница для администратора базы данных. Инструкция для пользователя — «[Карты сопоставления номенклатуры](#)».

Здесь: какие таблицы и скрипты нужны, где хранятся настройки модуля, все параметры с умолчаниями, подключение ИИ, требования к интерфейсам (гридам) окон. Предполагается, что вы знакомы с «[Конфигуратор: Настройка интерфейсов SQL представлений](#)» и форматом ini-описаний «НЭП».

1. Установка: таблицы и скрипты

Скрипты применяются по порядку (Firebird, диалект базы — 1):

Скрипт	Что делает
GOODS_MAP.sql	Таблицы карт (GOODS_MAP_SOURCES) и строк карт (GOODS_MAPS), генератор, словари в CLASSES_ITEMS: тип карты (CLASS_ID = 433), статус (434), способ сопоставления (435).
GOODS_MAP_AUTO.sql	Поля автоподбора (AUTO_*) — что подобрала программа, чтобы «Отменить выбор» мог к этому вернуться.
GOODS_MAP_LOADS.sql	Сессии загрузки: GOODS_MAP_LOADS, GOODS_MAP_LOAD_ROWS.
GOODS_MAP_LOAD_ORDER.sql	GOODS_MAP_LOAD_ROWS.ORDER_ID — номер строки в порядке загрузки (колонка «№»). Без него загрузка и импорт по карте не работают.
GOODS_MAP_STATUS_NOT_OURS.sql	Пункт 5 «Нет у нас» в словаре статусов — только подпись для колонки «Статус» (в базе «Нет у нас» — это «Подтверждено» без позиции).
GOODS_MAP_COMMENTS.sql	Описания таблиц и полей (COMMENT ON).

Синхронизации с удалёнными базами у таблиц карт нет.

Права. Отдельного права на окно пока нет: окно доступно всем, у кого есть доступ к меню «Операции».

2. Где хранятся настройки

Настройки модуля — текст в формате ini (комментарий — строка, начинающаяся с #; список — Ключ=^ ... ^). Действуют по порядку, каждый следующий уровень перекрывает предыдущий:

1. **Умолчания программы** (указаны у каждого параметра ниже).
2. **Скрипт GOODS_MAP_SETTINGS** (скрипт с этим именем в таблице скриптов SCRIPTS) — общие настройки для всех карт.
3. **Настройки карты** — поле «Настройки загрузки» в карточке карты

(GOODS_MAP_SOURCES.MAP_SOURCE_PROPS). Те же секции и ключи; одинаковые ключи карты заменяют общие, остальные общие остаются.

Окно сопоставления показывает **действующие** значения (общие + карта) и записывает в карту только то, что пользователь изменил в окне:

Что в окне	Куда пишется (в настройках карты)
Разметка колонок, «Строка данных с»	[Columns] (Code=, Articul=, Name=, Ed=, Okei=, StrokeCode=, Production=, Price= — буквы колонок; FirstRow=)
«Артикул из наименования»	[Load] ArticulFromName
«Кандидат от, %», «Предлагать от, %»	[Match] MinScore, [Match] ProposeScore
«Искать и в картах»	[Rules] OtherMapIDs
Диалог «Подобрать через ИИ», галка «Запомнить для этой карты»	[AI] — изменённые ключи

Запись не трогает остальной текст настроек: комментарии, порядок и списки сохраняются.

3. Параметры: [Load] — загрузка

Ключ	Умолчение	Что задаёт
LoadNameTemplate	<:SOURCE_NAME> <:DATE> <:TIME>, строк: <:ROWS>	Имя сессии при «Загрузить и сопоставить» (предлагается, пользователь может поправить).
ImportLoadNameTemplate	Импорт: <:DOC_NAME>	Имя сессии при импорте документа по карте.
ArticulFromName	пусто	Правило извлечения артикула из наименования (см. ниже). Обычно задаётся у карты.
NameSimilarity	0.7	Порог схожести наименований (0..1). Строка с тем же кодом / артикулом, что у подтверждённой, но с наименованием, отличающимся сильнее, получает «Проверить».

Теги имени сессии (регистр не важен): <:SOURCE_NAME> — карта, <:DATE> — дд.мм.гггг, <:TIME> — чч:мм, <:USER> — пользователь, <:ROWS> — число строк, <:DOC_NAME> — документ (только импорт).

Правила ArticulFromName:

Значение	Что берётся	Пример
brackets	Текст в последних скобках; несколько слов — последнее	(арт. MVA20-1-016-C) → MVA20-1-016-C
first / last	Первое / последнее слово наименования	
prefix:ТЕКСТ	Первое слово после ТЕКСТ (регистр не важен)	prefix:арт.: «Кабель ВВГ арт. 12345» → 12345
regex:ВЫРАЖЕНИЕ	Регулярное выражение без учёта регистра; артикул — первая группа (нет групп — всё совпадение)	

Значение	Что берётся	Пример
пусто	Не извлекать	

4. Параметры: [Match] — пороги подбора

Ключ	Умолчение	Что задаёт
MinScore	30	«Кандидат от, %»: в кандидаты попадают позиции с оценкой не ниже.
MaxCandidates	10	Сколько кандидатов хранить у строки (1..50).
ProposeScore	70	«Предлагать от, %»: лучший кандидат с такой оценкой ставится строке автоматически. Он же — порог «предложенных» при импорте, если в настройках импорта порог не задан.
ProposeGap	10	...и лучший выше второго не меньше чем на столько, иначе неоднозначно — строка остаётся «Новой».

5. Параметры: [Search] — поля нашей номенклатуры

Списки ^ ... ^, по строке ТАБЛИЦА.ПОЛЕ. Таблицы — GOODS и E_GOODS (пользовательские поля позиции, связь по GOOD_ID); без таблицы — GOODS. Позиция находится по любому из своих наименований / артикулов; значение, которое есть у нескольких позиций, считается неоднозначным и автоматически не предлагается. Ошибка в имени поля — ошибка при загрузке с текстом запроса.

Ключ	Умолчение	Что задаёт
SearchGoodsNameFields	GOODS.GOODS_NAME	Наименования (группа Name).
SearchArticulFields	GOODS.ARTICUL	Артикулы (группа Articul).
SearchStrokeCodeFields	GOODS.STROKE_CODE1, GOODS.STROKE_CODE2	Штрихкоды (группа StrokeCode).
SearchKeyFields	нет	Свои группы полей для правил: строки ИмяГруппы=ТАБЛИЦА.ПОЛЕ[, ТАБЛИЦА.ПОЛЕ ...]. Имена Name, Articul, StrokeCode, MapName, MapArticul заняты.
GoodsWhereSQL	g.CHECKED = 0	Какие позиции могут быть найдены: условие к GOODS (алиас g; E_GOODS — e). Группы (MIX_ID = 0) исключаются всегда.

Что реально попало в поиск, видно во второй строке итога загрузки («Наша номенклатура для поиска: ...»).

Пример:

```
[Search]
SearchGoodsNameFields=^
GOODS.GOODS_NAME
GOODS.GOODS_NAME1
GOODS.GOODS_NAME2
^
SearchKeyFields=^
```

```
SupplierCode=E_GOODS.GOOD_ADV_PROPI
^
```

6. Параметры: [Rules] — правила точной сверки

MatchRules — список правил, по строке:

```
Имя=<:SOURCE( ... )><:TARGET( ... )><:SCORE( ... )>[параметры]
```

Часть	Значения
<:SOURCE(...)> — что взять из строки поставщика	Code — код; Articul — артикул; ArticulFromName — артикул из наименования по [Load] ArticulFromName; NameArticul — каждое «артикулоподобное» слово наименования (есть цифры, от 4 знаков); Name — наименование целиком; StrokeCode — штрихкод
<:TARGET(...)> — с чем сравнивать	Name, Articul, StrokeCode (группы [Search]), своя группа из SearchKeyFields; MapName / MapArticul — наименования / артикулы подтверждённых строк других карт
<:SCORE(...)>	Оценка совпадения 0..100
<:MINLEN(N)>	Значение поставщика не короче N знаков, иначе правило не применяется
<:STATUS(Check)>	Совпадение по правилу всегда ставит «Проверить», а не «Предложено»

Нет списка — встроенные правила:

```
MatchRules=^
CodeToArticul=<:SOURCE(Code)><:TARGET(Articul)><:SCORE(90)>
Articul=<:SOURCE(Articul)><:TARGET(Articul)><:SCORE(95)>
ArticulFromName=<:SOURCE(ArticulFromName)><:TARGET(Articul)><:SCORE(95)>
ArticulInName=<:SOURCE(NameArticul)><:TARGET(Articul)><:SCORE(90)><:MINLEN(6)>
StrokeCode=<:SOURCE(StrokeCode)><:TARGET(StrokeCode)><:SCORE(95)>
Name=<:SOURCE(Name)><:TARGET(Name)><:SCORE(90)>
^
```

Проверяются все правила. Несколько правил указали на одну позицию — к лучшей оценке прибавляется AgreeBonus. Правила указали на разные позиции с близкими оценками — строка «Проверить».

Ключ	Умолчание	Что задаёт
AgreeBonus	5	Прибавка, если несколько правил указали на одну позицию (не выше 100).
ConflictGap	10	Разные позиции с оценками ближе этого — «Проверить».
OtherMapIDs	нет ключа	В каких картах искать (MAP_SOURCE_ID через запятую; пусто — не искать). Пишет окно (список «Искать и в картах»). Если ключ задан — главное двух следующих.
UseOtherMaps	0	Старый способ: 1 — искать во всех включённых картах.

Ключ	Умолчение	Что задаёт
OtherMapsTypes	пусто	Старый способ: только карты этих типов (0 — прочее, 1 — поставщик, 2 — покупатель, 3 — внешняя система).

Когда поиск по другим картам включён и в MatchRules нет правил с целями MapName / MapArticul, добавляются встроенные: по наименованию и артикулу с оценкой 85, по артикулу в наименовании — 80. Другие карты учитываются и в нечётком подборе: кандидат, найденный по наименованию в другой карте, помечается («Найдено по»).

7. Параметры: [Fuzzy] — нечёткий подбор кандидатов

Работает, когда точная сверка ничего не дала. Оценка кандидата — лучшая похожесть его наименований (и наименований в отмеченных картах) с наименованием поставщика ×100, плюс прибавка за артикул, не выше потолка.

Ключ	Умолчение	Что задаёт
MaxScore	89	Потолок оценки нечёткого подбора — ниже точного совпадения.
ArticulBonus	25	Прибавка за артикул в наименовании поставщика.
ArticulBonusMode	Own	Own — только если в наименовании есть слово, равное артикулу этого кандидата; Any — любое общее слово с цифрами от 4 знаков.
TopCount	30	Сколько позиций с наибольшим весом слов оценивать.
CommonWordShare	10	Слово, которое есть больше чем у 1/N всех позиций (и больше чем у 30), считается частым («для», «белый») и не учитывается.

8. Параметры: [AI] — подбор через ИИ

Значения подставляются в диалог кнопки «Подобрать через ИИ»; галка «Запомнить для этой карты» пишет изменённые в карту.

Ключ	Умолчение	Что задаёт
ReturnCount	3	Сколько лучших кандидатов строки ИИ возвращает с процентом (колонок «ИИ, %» в окне кандидатов). Больше — длиннее ответ.
CandidatesCount	20	Сколько кандидатов передавать на строку.
CandidatesMinScore	15	Кандидат не ниже этой оценки нечёткого подбора, %.
BatchSize	15	Строк в одном запросе. Больше — дешевле и быстрее, но длиннее ответ (риск обрезки).
CheckBelow	80	Уверенность ИИ ниже — статус «Проверить».
Model	из AI_SETTINGS	Модель для этой задачи. DeepSeek: deepseek-chat — быстрая и дешёвая, deepseek-reasoner — дольше и дороже, точнее на трудных строках.
Temperature	из AI_SETTINGS	0 — самые предсказуемые ответы (для выбора из списка — 0).

Ключ	Умолчание	Что задаёт
Instructions	нет	Свои пояснения для ИИ — особенности вашей номенклатуры (многострочный список, см. пример ниже), добавляются к инструкции.

Пример:

```
[AI]
ReturnCount=3
Instructions=^
В нашей номенклатуре размеры пишутся через "х" латиницей: 4.2х25.
"Оц." - оцинкованный, "Н/Ж" - нержавеющей.
^
```

9. Подключение к ИИ — скрипт AI_SETTINGS

Общий для всех задач «НЭП», которые обращаются к ИИ. Скрипт AI_SETTINGS (в таблице скриптов SCRIPTS), секция [AI]. Протокол — OpenAI-совместимый (DeepSeek, OpenAI, Mistral, OpenRouter, локальные Ollama / LM Studio): смена поставщика — смена адреса, модели и ключа.

Ключ	Умолчание	Что задаёт
BaseURL	https://api.deepseek.com	Адрес API; /chat/completions дописывается сам.
Model	deepseek-chat	Модель.
ApiKey	—	Ключ API. Обязателен. Нет ключа — при нажатии «Подобрать через ИИ» программа покажет образец скрипта.
Timeout	180	Ожидание ответа, секунд.
Temperature	0	0 — самые предсказуемые ответы.
MaxTokens	0	Потолок длины ответа; 0 — умолчание сервиса. Если ответ обрезается — уменьшите BatchSize или поднимите MaxTokens.
Retries	2	Повторов при перегрузке сервиса (HTTP 429 / 5xx) и обрыве связи.

```
[AI]
BaseURL=https://api.deepseek.com
Model=deepseek-chat
ApiKey=<ваш ключ>
```

Ключ лежит в базе: его видят все, у кого есть доступ к скриптам. В ИИ отправляются наименования, артикулы, коды, единицы и производители строк поставщика и кандидатов из нашей номенклатуры, а для кандидата, найденного по другой карте, — имя этой карты и наименование в ней. Цены и реквизиты клиентов не отправляются.

Сообщения об ошибках: неверный ключ — HTTP 401, нет средств на счёте — HTTP 402; прогон останавливается с понятным текстом.

10. Интерфейсы окон (Конфигуратор)

Стандартные описания приходят с обновлением базы. При правке учитывайте:

GOODS_MAPS — таблица окна сопоставления

- Ключ — GOODS_MAP_ID, алиас таблицы — m.
- **Обязательные скрытые поля** (<:MAIN><:NOTVISIBLE>): MATCH_STATUS, MATCH_METHOD, GOOD_ID, ED_ID — по ним окно красит строки и работает с текущей строкой.
- Колонка «Статус» — поле MATCH_STATUS_NAME со списком словаря 434:
 <:ITEMS(434@ITEM_NAME)>case when m.MATCH_STATUS = 2 and m.GOOD_ID = 0 then 5 else m.MATCH_STATUS end (так показывается «Нет у нас»).
- Параметры окна в запросе — только через \${Form....} (не через имя формы — оно может измениться): \${Form.MapSourceID} — выбранная карта; \${Form.StatusFilter} — фильтр статуса (−1 — все); \${Form.LoadsFilterSQL} — условие по выбранным сессиям (уже с and, пусто — все); \${Form.OrderLoadID} — сессия для колонки «№».
- Колонка «№» — left join GOODS_MAP_LOAD_ROWS lo on (lo.GOODS_MAP_ID = m.GOODS_MAP_ID and lo.MAP_LOAD_ID = \${Form.OrderLoadID}) в запросах выборки и обновления строки, поле lo.ORDER_ID. В SQL самих полей шаблоны \${...} не раскрываются — поэтому через join.
- Колонка «Сессии» — список имён сессий строки: (select RESULT_BLOB from GET_SQL_LIST_BY_BLOB('select Trim(l2.MAP_LOAD_NAME) from GOODS_MAP_LOADS l2 JOIN GOODS_MAP_LOAD_ROWS lr2 ON (lr2.MAP_LOAD_ID = l2.MAP_LOAD_ID) where lr2.GOODS_MAP_ID = :ID1 order by l2.MAP_LOAD_ID', <:Alias>.GOODS_MAP_ID)).
- Окно само открывает датасет (действие AfterUpdateSQL); правка в гриде — только BOX.

GOODS_MAPS_BY_GOOD — окно «Варианты позиции в картах»

- Параметр запроса GOOD_ID — позиция текущей строки окна, из которого открыто (склад, карточка позиции): where m.GOOD_ID = :GOOD_ID.

Набор колонок по умолчанию. Пользовательский набор колонок хранится в реестре под именем окна. Задайте у обоих интерфейсов набор колонок по умолчанию — иначе у нового пользователя (или после переименования окна) таблица может открыться без колонок.

11. Диагностика

- Запуск программы с ключом -log: в журнал пишутся запрос поиска по номенклатуре, раскрытие шаблонов \${...} (строки TPL:) и запросы гридов.
- Итог загрузки во второй строке показывает, по каким полям и скольким позициям строился поиск.
- «Пояснение» строки в окне называет сработавшие правила и причину статуса.

From:

<http://docs.comsystems.ru/> - **Руководство пользователя НЭП**

Permanent link:

http://docs.comsystems.ru/docs/settings/goods_maps_settings

Last update: **2026-10-10 22:24**

