

Том VI
KLT/RBD: программный слой, база данных, сайт и
правовой контур
Volume VI · Russian Edition · v8.1

Иван Борисович Курпишев / Ivan Borisovich Kurpishev
Independent Researcher · Kaliningrad · me@kurpishev.ru
2026

KLT-DOCTRINE-FINAL-MONOGRAPH-VOLUME-VI-KLT-RBD-SOFTWARE-LEGAL-RU-v8.1

Аннотация

Том VI собирает программно-регистрационный контур Доктрины Курпишева: KLT 4.14, KLT 5.1, RBD/RPD, реальный импорт данных, сайт, public/private split, депозитные материалы и осторожную карту программа - база данных - изобретательский кандидат. Том является техническим и редакционным binder, а не заменой внешней юридической проверки перед подачей.

Abstract

Volume VI assembles the software and registration contour of the Kurpishev Doctrine: KLT 4.14, KLT 5.1, RBD/RPD, real-data import, site publication, public/private split, deposit materials, and a cautious software-database-invention boundary. It is a technical and editorial binder, not a substitute for live legal filing review.

Словарь обозначений

- C@C - событие@состояние.
- Rep(R,I,U;D) - Reper-четвёрка.
- λ - гармоническая мера согласованности.
- CGI - индекс причинно-графического разрыва.
- RBD/RPD - Reper Database / Reper-Projective Database.
- Evidence-D - достаточное основание, пригодное для аудита.
- public/private split - разделение публикуемых и контролируемых материалов.

0. Редакционный статус и точка v8.1

Том VI фиксирует прикладной, программный и регистрационный контур Доктрины Курпишева. Его задача - собрать в одном месте KLT 4.14, KLT 5.1, RBD/RPD, импорт реальных артефактов, сайт, депозитные материалы и осторожную правовую карту: что относится к программе, что к базе данных, что может быть только кандидатом на компьютерно-реализованное техническое решение.

Контрольная точка:

KLT-DOCTRINE-FINAL-MONOGRAPH-VOLUME-VI-KLT-RBD-SOFTWARE-LEGAL-RU-v8.1

Том не является заменой юридической экспертизы перед подачей. Он формирует математико-технический корпус, карту источников, границы утверждений, карту депонирования и QA-контур. Актуальные формы, пошлины и процедурные детали ФИПС/Роспатента подлежат отдельной live-проверке непосредственно перед подачей.

1. Сквозная формула программно-регистрационного слоя

Главная цепь Тома VI:

`Object -> C@C_x -> Rep_x(R,I,U;D) -> λ_x -> CGI_x -> Status_x -> RBD_x -> publication/deposit route`.

Для программного объекта это читается так: файл, документ, пакет, страница сайта, таблица или сборка сначала фиксируется как событие@состояние. Затем ему назначается Reper-четвёрка: R - фактическое содержимое и работоспособность; I - проектная идея; U - поле допустимых применений; D - достаточное основание: источник, спецификация, проверка, контрольная сумма, журнал запуска или документ.

Программный truth-status невозможен без Dom и D. Если отсутствует предметный домен, контрольная среда, исходный файл, проверочный журнал, версия или связка с манифестом, объект получает gap-status: не ошибка уничтожения, а очередь пересборки.

2. KLT 4.14 как проверенный демонстрационно-аудиторский слой

KLT 4.14 в этом томе фиксируется как проверенная программная сборка переходного типа. Она сохраняет учебно-аудиторскую линию KLT 4.13, но расширяет её в сторону универсального импорта документов, архивов, веб-оболочки и мобильного контура.

Основной технический Reper KLT 4.14:

``Rep_KLT414 = (R_build, I_lambda_check, U_import_export, D_verification)``.

Содержательно это означает:

- R_build: Python-пакет, CLI, web shell, шаблоны DOCX, примеры и selftest-вывод;
- I_lambda_check: идея проверки работы через lambda-чек-лист, сравнение с идеальным ответом и ведомость;
- U_import_export: TXT/DOC/DOCX/HTML/PDF/изображения/TEX/ODT/EPUB/архивы и массовые выгрузки;
- D_verification: selftest.log, CHECK_RESULT, рендеры, шаблоны, демонстрационные результаты и контрольные файлы.

Положение о статусе: KLT 4.14 является программной реализацией прикладного аудита. Она не заявляет авторство на классические математические объекты, а фиксирует авторскую архитектуру их программного применения в KLT/RBD-контуре.

Таблица 1. KLT 4.14 - состав программного слоя

Компонент	Назначение	D-основание
CLI Python engine	локальная проверка работ и сравнение с идеальным ответом	README, app.py, tests, selftest
Universal import	TXT/DOCX/HTML/PDF/изображения/TEX/ODT/EPUB/архивы	importers.py, archives.py
DOCX reports	лямбда-чек-листы, ведомости, plagiarism sheet	templates, reports.py
Web shell	drag-and-drop интерфейс	web/index.html, server.py
Flutter shell	мобильная оболочка Android/iOS	mobile_flutter_shell

3. KLT 5.1 как SDK и мост к проектному графу

KLT 5.1 закрепляется как расширенный SDK-слой. Его назначение - объединить линию KLT 5.0, проектный construction graph engine, PEAKS/PIX-модель, lambda-дефект, финансово-проектные показатели и визуальные выходы в кроссплатформенную оболочку.

Формула SDK-слоя:

``KLT5.1 = Flutter_UI + Python_reference_core + Project_JSON + lambda_engine + SVG/JSON/MD reports + verification``.

Расчётный центр:

``lambda = ((U - R)(I - D)) / ((U - D)(I - R))`,
`delta = |lambda + 1|`,
`Auth = 1/(1 + delta)``.

Критическое отличие KLT 5.1 от чистого текста: объектом аудита становится проектная конфигурация - работы, ресурсы, поставки, сметы, договорные параметры, финансовые отклонения, технологические наложения и графы риска. Поэтому KLT 5.1 занимает место между математическим Rепер-ядром и прикладным цифровым продуктом.

Таблица 2. KLT 5.1 - SDK-архитектура

Компонент	Назначение	D-основание
Flutter app	трёхязычный интерфейс и визуальные вкладки	app/lib, pubspec.yaml
Python reference core	проверяемое ядро расчётов	core_python/klt51_core.py, tests
Project JSON	works/resources/deliveries/costs/finance	examples/demo_project.json
Lambda engine	lambda, delta, Auth, D*, MSI, DI, CRI	docs/TZ, source code
SVG/MD/JSON outputs	схемы, графики, отчёт, результат	examples/out

4. RBD/RPD как база реперов и граф доказательной прослеживаемости

RBD/RPD в Томе VI определяется как база реперных объектов, источников, версий, страниц, программных сборок, депозитных материалов и proof-status очередей. Это не плоская библиография и не каталог файлов. Это вычисляемая карта оснований.

Минимальная схема:

``source -> artifact -> C@C -> Rep -> lambda_audit_item -> evidence_D -> status -> release/deposit/publication link``.

После реального импорта артефакты получают не только имя файла, но и роль: site_page, fips_package, publication_package, software_build, public_private_split, legal_review_event, lambda_audit_item. Отсюда появляется ключевой принцип: публикация и регистрация не совершаются напрямую из папки файлов. Они проходят

через RBD-узел, статус приватности, legal-review queue и проверку достаточного основания.

Таблица 3. RBD/RPD - импортированные классы

Класс	Смысл	Статус
import_artifacts	реальные обнаруженные артефакты	контролируемый реестр
site_pages	страницы сайта и зеркала	публичный/контролируемый split
fips_packages	пакеты заявочной подготовки	live-check queue
software_builds	программные сборки	sha/checksum binding
lambda_audit_items	узлы lambda-аудита	gap/review mechanism
rbd_release_links	связь артефакта с выпуском	rollback and release map

5. Реальный импорт данных и очереди контроля

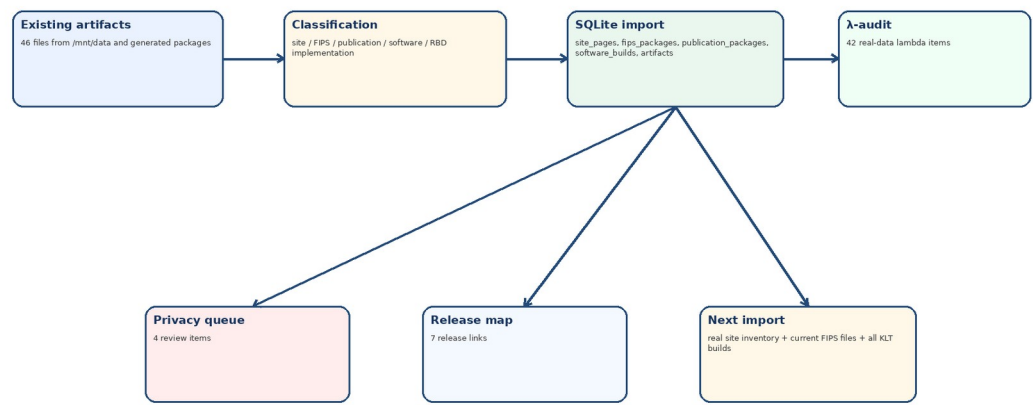
Пакет `KLT-RBD-SCHEMA-EXTENSION-REAL-DATA-IMPORT-v1.0` переводит реальные артефакты проекта в SQLite-базу. Тем самым сайт, PDF/HTML/MD/CSV/JSON/SQL/ZIP/checksum-файлы, FIPS-пакеты, программные сборки и publication packages становятся объектами запроса, аудита и дальнейшей сборки.

Важнейшая дисциплина этого слоя: оставшиеся очереди не трактуются как дефект системы. FIPS live-check queue, privacy review queue, lambda gaps и public/private split являются частью корректной RBD-архитектуры. Они показывают, какие элементы нельзя публиковать, подавать или выдавать как завершённые без дополнительного основания.

Теорема прослеживаемости: если каждый публикуемый или депонируемый объект имеет source-id, artifact-id, Reper-id, checksum, privacy-status, legal-review-status и release-link, то публикационный пакет восстанавливает свою цепочку происхождения без обращения к неформальной памяти проекта.

Рис. 1. Схема реального импорта данных в RBD/RPD.

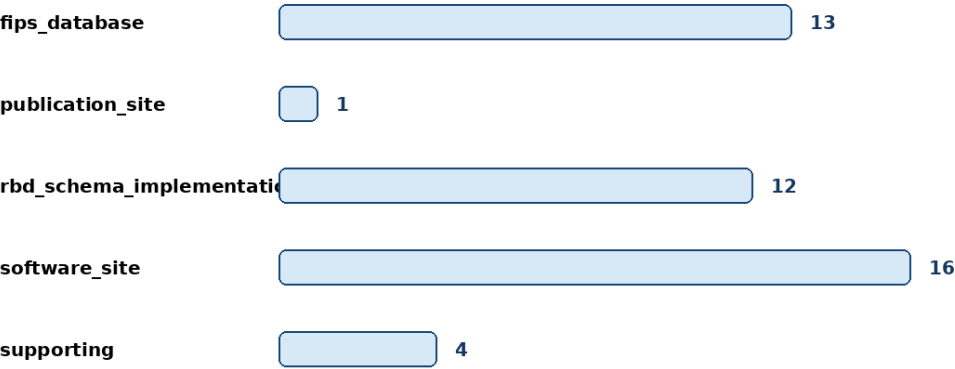
RBD/RPD Real Data Import v1.0



Result: generated artifacts are now indexed as real database objects with checksums, privacy decisions, release links and lambda traceability.

Рис. 2. Распределение импортированных артефактов по классам.

Imported artifacts by class



6. Сайт как публикационный слой, а не как источник истины

Сайт kurpishev.com в данной сборке рассматривается как публичный маршрут, но не как первичный источник truth-status. Первичным остаётся связка: исходник, манифест, RBD-узел, контрольная сумма, proof-status и QA.

Публичный маршрут имеет вид:

`source package -> site route -> public annotation -> download map -> checksum -> rollback point`.

Для KLT 4.14/5.1 сохраняется дисциплина коротких путей: `/ru/klt/k414.html`, `/ru/klt/k51.html`, папки `a/`, `d/`, `s/`, `v/`, а также карты `manifest.txt`, `map.json`, `nav.json`, `sitemap.xml`. Это не косметическая мера: короткие имена становятся техническим условием публикационной воспроизводимости.

7. Граница программа - база данных - изобретение

Для правового контура вводится осторожная трехчастная карта.

1. Программный контур: KLT 4.14 и KLT 5.1 как конкретные реализации в виде исходников, интерфейсов, reference-core, тестов, документации, примеров и визуализаций.

2. База данных: RBD/RPD как структурированный набор таблиц, графов, записей, связей, импортов, evidence-binding, release links и поисково-аудиторских представлений.

3. Изобретательский контур: только кандидатная линия компьютерно-реализованного способа/системы обработки разнородных цифровых документов и проектных данных. В этой линии не заявляются как таковые чистые формулы, математический метод или абстрактная идея; предметом анализа может быть техническая архитектура: адаптеры ввода, графовый аудит, локализация разрывов, генерация отчёта и управляемая пересборка.

В этом томе изобретательская часть получает статус `candidate / requires external legal-patent review`, а не статус готовой заявки.

Таблица 4. Правовая граница

Слой	Маршрут	Ограничение
KLT 4.14 / KLT 5.1	программа для ЭВМ / software dossier	не заявляет классическую математику как собственность
RBD/RPD	база данных	персональные и mixed artifacts требуют privacy review
KLT algorithmic method	кандидат на компьютерно-реализованный способ	нужна внешняя патентная экспертиза
Site layer	публикация и публичная аннотация	не заменяет источник истины и официальный депозит

8. Депозитный пакет и public/private split

Депозитный пакет должен быть разделён на публичные, контролируемые и закрытые элементы.

Публичные элементы: сайт-ready страницы, публичные аннотации, краткие README, демонстрационные JSON/SVG/MD, общие схемы и маршруты загрузки.

Контролируемые элементы: исходные zip-пакеты, SQLite-базы, CSV-экспорты, SHA256, манифесты, проверочные журналы, технические задания, заявочные черновики и карты соответствия полей.

Закрытые или требующие проверки элементы: персональные данные, mixed artifacts, приватные документы, материалы с неочищенными реквизитами, промежуточные legal-review заметки, а также всё, что может раскрыть лишнюю информацию до подачи.

Правило: `public route` не может быть создан раньше `public/private split`. Если split отсутствует, объект не публикуется, а переводится в очередь review.

9. QA-контур Тома VI

QA-контур Тома VI состоит из пяти проверок.

- Source QA: перечислены источники, роли и SHA256.
- Document QA: созданы DOCX, PDF, TEX, MD.
- Render QA: выполнен рендер DOCX и PDF в PNG-страницы, собраны contact sheets.
- Package QA: создан ZIP, сгенерирован SHA256 и выполнен `unzip -t`.
- Legal-technical QA: подготовлены legal boundary matrix и deposit materials map; финальная официальная процедура оставлена для live-проверки перед подачей.

Итоговый статус v8.1: `ready as monograph volume and technical-registration binder; not final legal filing without live official-form verification`.

10. Основные определения и теоремы

Определение 10.1. Software-Reper есть четверка `Rep_SW = (R_code, I_method, U_runtime, D_verification)`, где D_verification включает исходник, контрольную сумму, тест, документацию и среду запуска.

Определение 10.2. Database-Reper есть четверка `Rep_DB = (R_records, I_schema, U_queries, D_integrity)`, где D_integrity включает схему, экспорт, checksum, SQL-запросы и журнал integrity-check.

Определение 10.3. Deposit-Reper есть четверка `Rep_DEP = (R_materials, I_claim_route, U_registration_routes, D_formal_basis)`, где D_formal_basis не заменяется внутренним желанием подать заявку, а требует актуальной формы, перечня материалов и проверки статуса персональных данных.

Теорема 10.4. Если программный или базовый объект имеет C@C-фиксацию, Reper-четверку, контрольную сумму, проверочный журнал и RBD-связку, то он допускает воспроизводимый технический audit trail.

Доказательство. C@C фиксирует событие и состояние. Reper задаёт четыре обязательных компонента. SHA256 фиксирует неизменность файла. Проверочный журнал связывает файл со средой. RBD-связка помещает объект в граф происхождения. Следовательно, объект может быть восстановлен как цепочка source -> artifact -> evidence -> status.

Следствие 10.5. Отсутствие любого из этих элементов не уничтожает объект, но переводит его из статуса готового депонирования в гар-очередь.

Приложение А. Карта источников**Таблица А1. Source binding**

Источник	Размер	SHA256	Роль
monograph5_0_ru.pdf	14300260	ea84bab141342eb6...	Master corpus 5.0: C@C, Rep, lambda, KLT/RBD applications, source-preservation discipline.
KLT_4_14_CHECKED_BUILD.zip	1170911	5d48e7ce184527fb...	Checked KLT 4.14 source/build: universal import, web shell, mobile shell, checklist/report workflow.
KLT5_1_FLUTTER_SDK_PACKAGE.zip	3019939	d373e55bab5e63fa...	KLT 5.1 Flutter SDK: cross-platform application, Python reference core, examples and verification.
klt_rbd_schema_extension_real_data_import_v1_0_fixed.zip	11969199	406632a118589a45...	RBD/RPD schema extension and real-data import: SQLite, imported artifacts, site pages, FIPS queues, lambda audit items.
kurpishev_com_klt414_fixed_site_package.zip	4714585	9b334cdfa2bcdf2d...	Short-name website logistics for KLT 4.14/5.1.
kurpishev_site_responsive_polished_pack.zip	18759229	89713f360a21100a...	Responsive site publication layer with KLT pages and download assets.
KLT DOCTRINE MONOGRAPH_SOURCE_v7_0_PACKAGE (1).zip	35008439	b50a0408fa8f745d...	Doctrine monograph source skeleton and assembly protocol v7.0.

Приложение В. Контрольные CSV

Созданы контрольные файлы: LEGAL_BOUNDARY_MATRIX_VOL6_RU_v8_1.csv и DEPOSIT_MATERIALS_MAP_VOL6_RU_v8_1.csv. Они включены в транспортный ZIP и используются как индекс правовой границы и депозитных материалов.

Следующая точка

KLT-DOCTRINE-FINAL-MONOGRAPH-VOLUME-VI-KLT-RBD-SOFTWARE-LEGAL-EN-v8.2

Английская редакция Тома VI: Software, RBD/RPD, site publication and legal-technical registration contour.