

МОНОГРАФИЯ БАЗОВЫЙ ПРОЕКТ KLT-RBD ТОМ III

V*P-ФИЗИКА И СТРАТИФИЦИРОВАННОЕ ВРЕМЯ: РЕДУЦИРОВАННАЯ ПРЕДСКАЗАТЕЛЬНАЯ ПАКЕТНАЯ КОСМОЛОГИЯ

Вычислительные сертификаты, детерминированная сборка
и протокол независимого повторного запуска



Курпишев Иван Борисович

Независимый исследователь · Калининград · 2026

КОНТРОЛЬНАЯ ТОЧКА. T3-CERT-v0.14. Зафиксированы reproducibility capsule, environment/dependency locks, детерминированная сборка, SLSA-aligned local provenance, тестовые векторы и протокол независимого повторного запуска.

СТАТУСНЫЙ ЗАМОК. truth_layer_promotion=0. Локальный повторный запуск имеет статус NUM. Независимый EXT-RUN не исполнен; EMP-OBS отсутствует. Подпись или хеш подтверждают происхождение и целостность, а не физическую истинность.

Паспорт выпуска T3-CERT-v0.14

Поле	Зафиксированное значение
Автор	Курпишев Иван Борисович
Проект	ЛОГИКА КУРПИШЕВА 2 · Базовый проект KLT-RBD
Том	III · V*P-физика и стратифицированное время
Контрольная точка	T3-CERT-v0.14
Предыдущая точка	T3-RBD-v0.13
Предмет	Код, environment lock, dependency ledger, test vectors, deterministic build, independent rerun protocol
Канонический импорт	Том I v197 RU; Том II T2-FREEZE-v1.0; T3-RBD-v0.13
Математический статус	DEF / PROP / NUM; внешний trust bridge OPEN
Физический статус	MODEL / PHY-HYP; EMP-OBS отсутствует
Следующая точка	T3-END2END-v0.15

0. Реестр изменений

ID	Изменение	Статус
T3-CS-073	Введена четырёхзначная система verdict: SAME-BYTES / SAME-SEMANTICS / DIVERGED / ABSTAIN	DEF-A
T3-CS-074	Определён reproducibility capsule и обязательные поля environment/dependency lock	DEF-A
T3-CS-075	Построена детерминированная ZIP-сборка с нормализацией времени, порядка, путей и режимов	PROP/NUM
T3-CS-076	Сформирована SLSA-aligned local provenance без заявления уровня и без подписи	MODEL
T3-CS-077	Проведён аудит KLT 4.14, KLT 5.1, KLT2 v63, klt6 v50 и release v4.80	NUM-AUDIT
T3-CS-078	Обнаружено расхождение 1/17 записей MANIFEST_v4_80.json	BLOCKER
T3-CS-079	Составлен независимый rerun receipt contract	EMP-PLAN
T3-CS-080	Выполнен T3-CERT-NUM-001: 20000/20000	NUM

1. Назначение и доказательные границы

ОНТОЛОГИЯ. Вычислительный сертификат существует как связанный набор исходников, инструкций, окружения, зависимостей, тестовых векторов, артефактов, хешей и provenance.

ГНОСЕОЛОГИЯ. Воспроизводимость считается установленной только для явно перечисленных артефактов и при повторении объявленных источников, окружения и команд. Совпадение функционального результата слабее совпадения байтов.

ФЕНОМЕНОЛОГИЯ. Для исследователя сертификат проявляется как возможность взять замороженный пакет, выполнить одну объявленную процедуру и получить проверяемый verdict.

ФИЗИЧЕСКИЙ СМЫСЛ. Воспроизводимый код позволяет повторить модельное вычисление. Он не показывает, что модель описывает природу, пока не подключены измерительные данные и независимый эмпирический протокол.

ОГРАНИЧЕНИЕ. Локальная самопроверка не является независимым EXT-RUN. Hash equality не доказывает корректность формулы; подпись не доказывает истинность модели; SBOM не доказывает отсутствие уязвимостей.

АВТОРСКАЯ НОВИЗНА. В архитектуре Курпишева сертификат включён в Reper/D/Dom-цепочку и переносит статусный вектор без повышения: source-card -> proof object -> code capsule -> rerun receipt -> forecast gate.

1.1. Граница prior art

Узел	Роль	Граница
Reproducible Builds	бит-в-бит воспроизводимость при одинаковых source/environment/instructions	официальное определение
SLSA v1.2	build/source provenance и проверяемая цепочка происхождения	Linux Foundation standard
in-toto	порядок и участники шагов supply chain	CNCF prior art
SPDX 3.0.1 / CycloneDX 1.7	машиночитаемые BOM и dependency relationships	SBOM prior art
Sigstore	подпись, identity binding и transparency log	supply-chain prior art
T3-CERT	blocker-safe capsule, status firewall и EXT-RUN receipt	авторская композиция

2. Формальная модель воспроизводимой сборки

$$\text{Build} : S \times E \times I \rightarrow A$$

S = source snapshot, E = environment, I = build instructions
 $\text{Reproducible}(A) \Leftrightarrow \forall r_1, r_2 \text{ with same } (S, E, I): \text{bytes}(A_1) = \text{bytes}(A_2)$

Определение T3-CERT-1 [DEF]. Частичность сохраняет ABSTAIN при неполном входе.

Поле E включает не только название операционной системы. Для действующей сборки перечисляются версия интерпретатора или компилятора, зависимости, архитектура, locale, timezone, переменные окружения, build path policy, файловые разрешения и источник времени.

2.1. Verdict lattice

Verdict lattice for a rerun

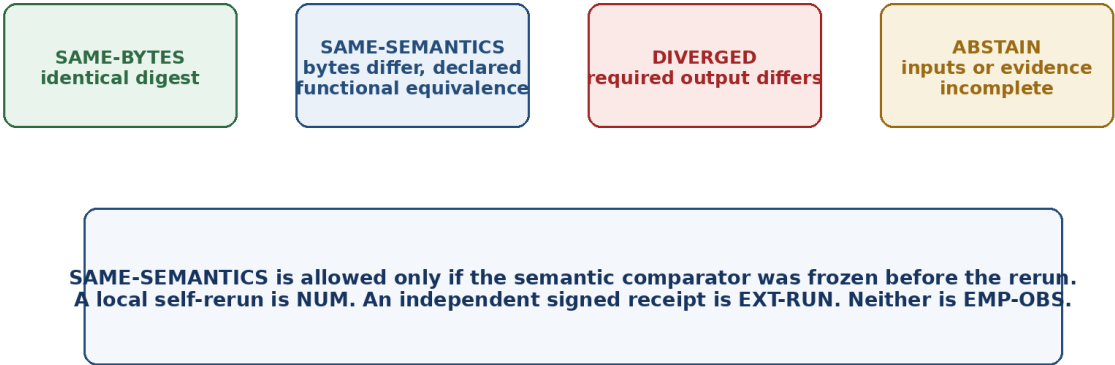


Рисунок 1. Verdict выбирается по слабейшему обязательному критерию.

Verdict	Условие	Разрешённый вывод
SAME-BYTES	SHA-256 и размер совпали	строгая локальная воспроизводимость артефакта
SAME-SEMANTICS	байты различны, заранее замороженный comparator подтверждает эквивалентность	только для объявленного semantic contract
DIVERGED	обязательный вывод или digest различается	расхождение, требующее диагностики
ABSTAIN	нет lock, Dom, input hash, build command либо receipt	вывод запрещён

3. Reproducibility capsule

Reproducibility capsule T3-CERT-v0.14

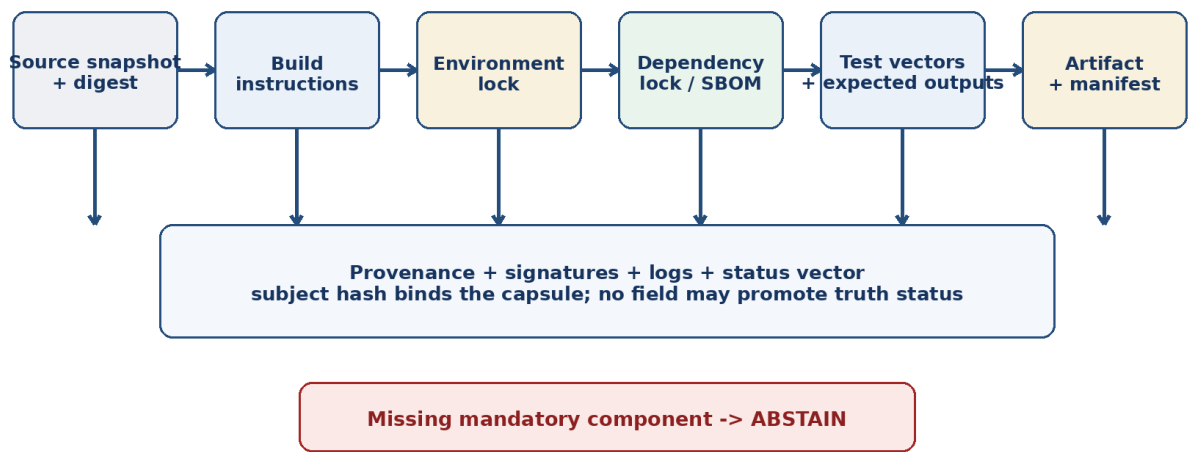


Рисунок 2. Капсула связывает все входы и результаты через subject hash.

3.1. Обязательный контракт

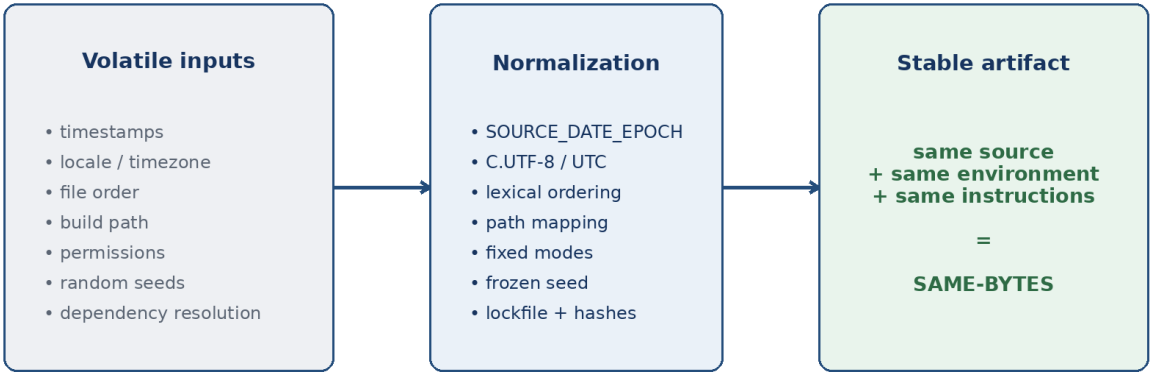
Компонент	Минимальное поле	Назначение
source	immutable snapshot + SHA-256	точно те байты, из которых выполнялась сборка
build	entrypoint, команды, flags, working directory	без устных шагов
environment	OS/runtime/architecture/locale/TZ/variables	environment lock
dependencies	точные версии + artifact hashes	lockfile или vendored bundle
test vectors	input, expected output, tolerances	предварительно заморожены
artifact	имя, media type, bytes, digest	subject сертификата
provenance	builder, parameters, dependencies, byproducts	неподписанная запись не является trust receipt
status	NUM / EXT-RUN / EMP-OBS channels	нескалярный firewall

Cert_build = (S,E,I,D,T,A,H,P,R,Status)
missing mandatory component -> ABSTAIN

D - dependency ledger; T - test vectors; P - provenance; R - external rerun receipt.

4. Детерминированная сборка и управление вариативностью

Deterministic build: variance is normalized, not wished away



Counterexample verified: changing only ZIP timestamps changed the bytes in 1000/1000 trials.

Рисунок 3. Вариативные входы нормализуются до вычисления digest.

Источник вариативности	Ремонт	Gate
Время	SOURCE_DATE_EPOCH; фиксированный ZIP timestamp	скрытая текущая дата запрещена
Порядок	лексикографический список входов и выходов	порядок обхода FS не используется
Locale/TZ	C.UTF-8 и UTC	форматы чисел и дат не зависят от машины
Путь	\${BUILD_ROOT} и path mapping	абсолютный путь не попадает в артефакт
Права	фиксированные file modes	umask не влияет на digest
Случайность	замороженный seed или криптографически зафиксированный entropy input	«случайный» тест воспроизводим
JSON	сортировка ключей, UTF-8, запрет NaN/Inf	стабильное байтовое представление

В T3-CERT-NUM-001 один и тот же capsule был собран при разных порядках исходных файлов и разных фиктивных build roots. Все 1000 пар дали идентичные байты. Контрольный наивный ZIP с различающимися timestamps разошёлся в 1000/1000 случаев.

5. Dependency lock, SBOM и environment lock

```
E_lock = (runtime, os, arch, locale, timezone, env-vars, build-path-policy)
D_lock = {(package_i, version_i, source_i, digest_i)}
```

Range constraint без разрешённого набора артефактов не является dependency lock.

5.1. Python и Dart

Для Python в текущем prior art доступен стандарт pylock.toml, предназначенный для воспроизводимой установки и содержащий точные источники и хеши пакетов. Для Dart application package официальная документация рекомендует хранить lockfile в системе контроля версий, чтобы разработка и развёртывание использовали согласованный набор зависимостей.

Корпус	Наблюдение	Verdict
KLT 4.14 Python	requirements.txt: 8 строк с >=; точных == нет; pylock.toml/lock отсутствует	ABSTAIN-DEPLOY
KLT 4.14 Flutter	7 caret constraints; pubspec.lock отсутствует	ABSTAIN-DEPLOY
KLT 5.1 Flutter	9 caret constraints; pubspec.lock отсутствует	ABSTAIN-DEPLOY
T3-CERT capsule	Python stdlib only; runtime 3.13.5 зафиксирован; third-party runtime deps отсутствуют	PASS-LOCAL

5.2. SBOM adapters

Профиль	Используемый слой	Ограничение
SPDX 3.0.1	package/component, relationship, license, external identifiers	экспорт dependency ledger
CycloneDX 1.7	components, dependencies, formulation, declarations, citations	экспорт build and BOM layer
KLT-RBD status vector	math/model/num/emp/legal channels	не заменяется полем confidence

6. Provenance, attestation и подпись

Attestation = (subject, predicateType, buildDefinition, runDetails)
subject.digest binds the artifact; builder.id identifies the asserted trust base

SLSA-aligned JSON создан локально, но SLSA level не заявлен.

Слой	Функция	Статус T3-v0.14
SLSA v1.2	где, когда и как произведён artifact; build/source provenance	модель полей и проверки
in-toto Statement	subject + predicateType + predicate	контейнер attestation
Sigstore	signature + identity + transparency log	план подписания внешнего receipt
T3 local provenance	builder=local-untrusted; signed=false; independent=false	NUM metadata only

ЮРИДИЧЕСКАЯ И НАУЧНАЯ ГРАНИЦА. Подпись подтверждает связь между ключом/идентичностью и конкретными байтами при заявленной процедуре проверки. Она не доказывает корректность алгоритма, истинность теоремы или физическую валидность результата.

Файл t3_v14_provenance.json использует in-toto Statement v1 и predicateType SLSA provenance/v1. Он сознательно содержит klt_status.signed=false, slsa_level_claimed=false и independent_runner=false. Это предотвращает превращение внутренней метаданных в фиктивный trust certificate.

7. Аудит программных пакетов KLT

Архив	Entries	Различных timestamps	Lockfiles	SHA-256
KLT 4.14	73	14	нет	5d48e7ce184527fb...
KLT 5.1	81	25	нет	d373e55bab5e63fa...
KLT6 root v50	36	4	нет	0854a90a12148620...
KLT2 v63	37	15	нет	748513c418e5ba8c...
Android release v4.80	22	7	нет	36412edaa2276bf8...

7.1. Установленные факты

- Копии KLT_4_14_CHECKED_BUILD.zip и KLT_4_14_CHECKED_BUILD (1).zip побайтово идентичны.
- Копии KLT5_1_FLUTTER_SDK_PACKAGE.zip и KLT5_1_FLUTTER_SDK_PACKAGE (1).zip побайтово идентичны.
- Ни в KLT 4.14, ни в KLT 5.1 не найден pubspec.lock; Flutter dependency resolution поэтому не заморожен.
- В KLT 4.14 requirements.txt используются восемь нижних границ $\geq$ без точных версий и хешей.
- KLT2 v63 содержит файловый manifest, но перечисление не снабжено digest для каждого объекта.
- klt6_root_v50 содержит сайт/дистрибутивный слой, но не единый environment/dependency lock.

ОГРАНИЧЕНИЕ. Аудит описывает состояние переданных архивов. Он не утверждает, что иной исходный репозиторий или CI-среда не содержит дополнительных lockfiles.

8. Release v4.80: manifest integrity и обнаруженный blocker

```
records = 17
verified = 16
mismatched = 1
Verdict = ABSTAIN-MANIFEST until repaired
```

Результат получен прямой сверкой размера и SHA-256 каждого embedded file.

Объект	Заявлено	Наблюдено	Статус
KLT v4 80 FINAL INTERNAL_RELEASE_CLOSURE_DELIVERY_MANIFEST.csv	MANIFEST_v4_80.json объявляет bytes=0 и SHA-256 пустого файла	в архиве фактически 2620 байт; SHA-256 другой	OPEN BLOCKER
Остальные 16 записей	размер и SHA-256 совпали	локальная целостность подтверждена	PASS-LOCAL

Расхождение не исправлялось молча. Требуется новая версия MANIFEST_v4_80.json либо новый delivery manifest с согласованными байтами и хешем. До change-set утверждение о полной manifest closure блокируется.

8.1. Hash map проверенных архивов

Архив	SHA-256
KLT 4.14	5d48e7ce184527fb7650ba5f5cd0ad33196d05d36f3b962f22d8a2bdbbcaefa6
KLT 5.1	d373e55bab5e63fa4c5b68688527efd6b53d30b2f71e8182716004b290f999fe
KLT6 v50	0854a90a12148620bff7e4e4cf2bd56e773692d36d9e1cc6480b7879db330e1a
KLT2 v63	748513c418e5ba8c89a8a5b931f161230f82e5be2dc95afba03aa6decb4feda7
release v4.80	36412edaa2276bf82a3b0146b49746dde0c310f1ad5d71df89d915351ac56763

9. Вычислительный сертификат T3-CERT-NUM-001

seed = 20260726
test families = 20
passed = 20000 / 20000
capsule SHA-256 =
d9edc37c9884ff6fb0c7665acf36bbf38e227c082a2bd6b402bb54977c3b3518

Сертификат использует Python standard library; внешний каталог или физические данные не анализировались.

ID	Проверка	Результат
C01	Canonical JSON independent of insertion order	1000/1000
C02	Deterministic capsule is bit-for-bit identical across roots and source order	1000/1000
C03	Artifact SHA-256 stable over repeated builds	1000/1000
C04	Counterexample: timestamp variance changes naive ZIP bytes	1000/1000
C05	Archive members have stable lexical order	1000/1000
C06	Path normalization removes dot segments without escaping root	1000/1000
C07	Path traversal is rejected	1000/1000
C08	Non-finite floats are rejected	1000/1000
C09	One-byte tampering changes artifact digest	1000/1000
C10	Embedded manifest verifies all payload entries	1000/1000

СТАТУС. Все результаты этой страницы имеют статус NUM/PROP в явно заданном synthetic carrier. Они не являются независимым EXT-RUN.

9.1. Продолжение матрицы тестов

ID	Проверка	Результат	Примечание
C11	Environment lock carries epoch, timezone and locale	1000/1000	
C12	Test vector output is stable and correct	1000/1000	
C13	Missing Dart lockfiles trigger reproducibility blocker	1000/1000	KLT 4.14 and KLT 5.1
C14	Range-only Python requirements trigger dependency-lock blocker	1000/1000	{'lines': 8, 'exact': 0, 'ranged': 8, 'caret': 0, 'sdk': 0}
C15	Embedded v4.80 manifest mismatch is deterministically detected	1000/1000	verified=16/17
C16	Declared-empty but nonempty CSV manifest is detected as blocker	1000/1000	
C17	Duplicate KLT archive copies are byte-identical	1000/1000	
C18	Unsigned local provenance cannot promote to EXT-RUN or EMP-OBS	1000/1000	
C19	Verdict engine distinguishes SAME-BYTES from divergence	1000/1000	
C20	Missing mandatory lock or external receipt yields ABSTAIN	1000/1000	

9.2. Что сертификат доказывает

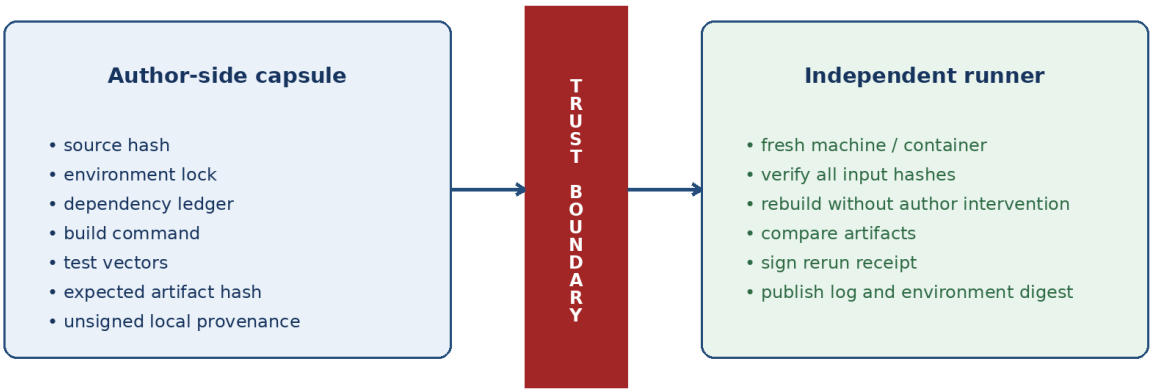
- Внутренняя reference-сборка capsule детерминирована при заявленной нормализации.
- Манифест обнаруживает изменение одного байта и проверяет payload entries.
- Gate корректно возвращает ABSTAIN при отсутствии обязательного dependency lock или external receipt.
- Переданные KLT-архивы имеют зафиксированные хеши; указанные дубликаты идентичны.
- Расхождение MANIFEST_v4_80.json воспроизводимо и локализовано в одной записи.

9.3. Чего сертификат не доказывает

- Побайтовую воспроизводимость полных KLT 4.14/5.1/Android builds: их среды и зависимости не заморожены полностью.
- Безопасность кода, отсутствие уязвимостей либо корректность всех алгоритмов.
- Независимость проверки, поскольку запуск выполнен в текущей авторской рабочей среде.
- Физическую истинность V*P, ПН.2, F-1-F-14, PredRep или космологических интерпретаций.

10. Протокол независимого повторного запуска

Independent rerun boundary



Current status: external receipt absent -> EXT-RUN OPEN; EMP-OBS remains absent.

Рисунок 4. Независимость возникает только после перехода trust boundary.

10.1. Последовательность EXT-RUN

Шаг	Действие	Выход
ER-1	Получить capsule и source snapshot из неизменяемого канала	input hashes
ER-2	Проверить SHA256SUMS и subject digest до распаковки	verification log
ER-3	Развернуть чистую VM/container по environment lock	environment digest
ER-4	Установить только разрешённые dependency artifacts	dependency receipt
ER-5	Выполнить ровно замороженную команду без консультации автора	stdout/stderr hashes
ER-6	Сравнить artifact bytes и semantic test vectors	verdict
ER-7	Подписать rerun receipt и опубликовать builder identity	signed receipt
ER-8	При divergence сохранить оба артефакта и diff report	blocker / change-set

10.2. Receipt contract

```
R_ext = (runner_id, independence_statement, input_hashes, env_digest, commands,
         outputs, verdict, signature, timestamp)
R_ext absent -> EXT-RUN = OPEN
```

Повторный запуск тем же автором или в той же непроверяемой среде сохраняет статус NUM.

11. Обновление T3-PO

Обязательство	Содержание	Статус
T3-PO-025	вычислительные сертификаты и воспроизводимость	PARTIAL-CLOSED / NUM
T3-PO-149	полный reproducibility capsule	CLOSED-LOCAL / NUM
T3-PO-150	детерминированная reference-сборка	CLOSED-PROP / NUM
T3-PO-151	environment lock schema	CLOSED-DEF
T3-PO-152	dependency lock для KLT 4.14/5.1	OPEN
T3-PO-153	SBOM export SPDX/CycloneDX	PARTIAL-MAP
T3-PO-154	SLSA/in-toto provenance	PARTIAL: unsigned local
T3-PO-155	artifact signature / trusted timestamp	OPEN
T3-PO-156	independent EXT-RUN receipt	OPEN
T3-PO-157	manifest repair v4.80	OPEN BLOCKER
T3-PO-158	full KLT deterministic build	OPEN
T3-PO-159	end-to-end operator integration	NEXT

11.1. Реестр блокиров

ID	Блокер
B-CERT-001	KLT 4.14 Python dependencies не зафиксированы exact versions/hashes
B-CERT-002	KLT 4.14 Flutter pubspec.lock отсутствует
B-CERT-003	KLT 5.1 Flutter pubspec.lock отсутствует
B-CERT-004	Flutter SDK/Android SDK/JDK/Gradle toolchain не заморожены единым lock
B-CERT-005	v4.80 JSON manifest расходится с CSV delivery manifest
B-CERT-006	KLT2 v63 manifest не содержит digest каждой записи
B-CERT-007	SLSA-aligned provenance не подписана и builder не независим
B-CERT-008	Trusted timestamp отсутствует
B-CERT-009	Независимый EXT-RUN не выполнен
B-CERT-010	EMP-OBS отсутствует; физический статус не повышается

12. Source-cards, контрольные хеши и следующая точка

Источник	Точный раздел	Идентификатор	Роль
Reproducible Builds	Definitions: “When is a build reproducible?”	official project page	prior art / definition
SLSA v1.2	Provenance and Build Provenance	Linux Foundation specification	provenance mapping
in-toto	Supply-chain integrity framework	CNCF graduated project	attestation container
SPDX	Specification 3.0.1	Linux Foundation specification	SBOM adapter
CycloneDX	Specification 1.7, Formulation and Dependencies	OWASP / ECMA-424	SBOM and build formulation
Sigstore	Overview: signing, identity, transparency log	OpenSSF project	signature plan
PyPA	pylock.toml Specification, lock-version 1.0	official packaging specification	Python dependency lock
Dart	Glossary: application packages should commit lockfiles	official Dart documentation	Flutter lock requirement

12.1. Контрольные хеши внутренних сертификатных объектов

Объект	SHA-256
t3_v14_certificate.py	fa6f94cd3ff979e8487c6259517424580f226a59a6c18c9952a168c0125d8839
t3_v14_capsule.zip	d9edc37c9884ff6fb0c7665acf36bbf38e227c082a2bd6b402bb54977c3b3518
t3_v14_dependency_ledger.json	9633213dd815883c964ce5cdd360285df9eb5a4f84aabe05b8135e6e45465d62
t3_v14_manifest.json	1b2bf0e442956c2f3658b08438a7413661105da9d61458cbdf43bdef2348a5a7
t3_v14_provenance.json	229b80dd1e703f111e5ec6659428ff69cba641f00771801c941501e28e8fee7
t3_v14_results.json	f329926cd2dd8f9f09b43a8adc4ded72379c0df1342882b1df114cb28072e62d
t3_v14_results.txt	e2a8ae1027c4acb2f5bab181aa9a94167ac1622a338b5cd03ec9ca399871f5a6
SHA256SUMS.txt	424637f5c96b379939b54362c6f17d3de43fe61514dd4a3cd8ee5ac8c58dfc93

12.2. Доказательная граница

reproducible NUM certificate != independent EXT-RUN
independent EXT-RUN != EMP-OBS
signature / registry != mathematical or physical truth

truth_layer_promotion = 0 сохраняется без исключений.

СЛЕДУЮЩАЯ ТОЧКА. T3-END2END-v0.15: композиция всех частичных операторов Тома III, перенос weakest-link status, матрица publication gate и сквозной интеграционный сертификат.