

Приложение В: Zulip АйТек — сущностная модель и жизненный цикл проекта (заказа)

Приложение В к РГ-СМК-Zulip-01.

Дополнение к карте каналов. Здесь — скелет, на котором держится вся логика: знание о **контрагентах** и **объектах**, и проходящий через стадии **проект** (= заказ).

Терминология (важно)

Слово	Значение в АйТек	Где
Проект	заказ клиента — поставка НКУ/РУСН на конкретный объект, со своим жизненным циклом	сквозной объект потока, канал <code>#проекты</code> , тема <code>Проект: ...</code>
Развитие	внутренние инициативы и изменения: цифровизация, интеграция, изменения СМК	канал <code>#развитие</code> (<code>Инициатива: </code>), владелец — LeanTech + СМК

«Проект» закрепляем за заказом (язык бизнеса). Внутренние инициативы переименовываем в `#развитие`, чтобы одно слово не означало две разные вещи.

Сущностный спайн

Логика навешана не на стадии потока, а на сущности и их связи:

Контрагент —1:N→	Объект —1:N→	Проект (=заказ) —1:N→	Изделия
заказчик,	площадка/	конкретная	НКУ 0,4 кВ,
генподрядчик,	сооружение	поставка на объект	РУСН-10,
поставщик,	(ПС, завод,	со своим ЖЦ	подстанции
субподряд	ЖК, цех)	(G0...G8)	

Это граф, а не строгое дерево: у одного объекта может быть несколько контрагентов (заказчик, генподряд, проектировщик), а проект тянет ещё и поставщиков. Поэтому модель живёт как связанные записи, а не как вложенные папки.

Где что живёт (четыре слоя):

- **CRM/ERP** — источник истины по сущностям: карточки Контрагентов, Объектов, Проектов и связи между ними (граф сущностей).
- **KB (BookStack)** — источник истины по документам: регламенты, стандарты КД, методики; коммерческий раздел держит профили контрагентов и объектов как знание.
- **Zulip** — обсуждение, навешанное на сущности через карточку темы и нейминг. Чат не хранит сущности, он на них ссылается.
- **Onyx** — индексирует все три источника; сущности становятся **якорями retrieval**, поэтому «знание об объекте/контрагенте» отвечается поверх записей, документов и обсуждения.

Тема проекта в Zulip

Чтобы тема несла объект и контрагента (а не только номер), первое сообщение темы = карточка проекта со ссылками в структурный слой.

Имя темы (коротко, стабильно): Проект: <код объекта> / <№> Пример: Проект: ПС-Северная / 2026-312

Карточка (первое сообщение):

Контрагент:	000 «Ромашка» (заказчик) · ГК «Стройсервис» (генподряд)
Объект:	ПС 110/10 кВ «Северная», г. Подольск
Проект №:	2026-312 · Договор: №10/02
Состав:	НКУ 0,4 кВ — 12 шкафов · РУСН-10 — 3 ячейки
Срок:	отгрузка до 2026-09-15
Записи:	CRM/ERP ↔ <ссылка> · KB ↔ <ссылка>
Статус:	G2 — договор подписан, в проектировании

Эта карточка даёт и людям, и Опух три якоря сразу — контрагент, объект, проект — по любому из которых ищется контекст.

Жизненный цикл проекта (шлюзы)

Тема **Проект:** — сквозная, идёт через стадийные каналы. На каждом шлюзе статус обновляется в карточке; стандартная работа стадии остаётся в стадийном канале.

Проект: <Объект> / <№>		
G0	лид / запрос на объект	#продажи → квалификация
▼		
G1	расчёт себестоимости + КП	#продажи → отправлено клиенту
▼		
G2	договор подписан	#продажи → проект «в работу»
▼		
G3	КД готова	#проектирование → запуск в производство
▼		
G4	материалы обеспечены	#снабжение
▼		
G5	изделие собрано	#производство (Подольск / Кострома)
▼		
G6	ОТК / ПСИ пройдены ■шлюз	#качество → готово к отгрузке
▼		
G7	отгружено на объект	#отгрузка
▼		
G8	ПНР / сдача	#сервис → проект закрыт

Стадийный канал ведёт повторяющуюся работу стадии (мощности, нормы, типовые проблемы), а конкретный проект — сквозную тему; из стадии на проект ссылаемся через **#проекты>Проект: ...**. Решение по проекту фиксируем **Решение:** внутри его темы; рекламацию — **Рекламация:** в **#качество** со ссылкой на объект и проект.

Что это открывает (запросы к Onyx)

Сущности-якоря превращают разрозненную переписку в наведируемое знание:

- **По контрагенту:** «все проекты ООО Ромашка», «история рекламаций по этому заказчику».
- **По объекту:** «что у нас по ПС Северная за всё время», «какие изделия туда уходили».
- **По проекту:** «статус 2026-312», «что решили по составу щитов».
- **Срез по шлюзу:** «какие проекты застряли на G4 (снабжение)».

Без сущностного спайна эти запросы недоступны — есть только тысячи сообщений. Со спайном — это и есть «знание об объектах и контрагентах», ради которого всё строится.

Связь с базой знаний (KB)

Контрагент / Объект / Проект — это коммерческий раздел KB в действии: именно эти сущности делают базу знаний наведируемой, а не складом документов. Один словарь сущностей между CRM/ERP, KB и Zulip-темами — обязательное условие, чтобы Onyx видел единый граф, а не три разных языка для одного объекта.

“ Нейминг един во всём комплекте: Проект: = заказ (#проекты), Инициатива: = изменение (#развитие). Это приложение В; грамматика тем — приложение А (itech-zulip-ontology.md), карта потока — приложение Б (itech-zulip-channels-valuestream.md); ядро — регламент РГ-СМК-Zulip-01 (itech-zulip-reglament-start.md).

Revision #2

Created 2026-06-19 03:53:44 UTC by Левицкий Алексей

Updated 2026-06-19 11:41:50 UTC by Claude Ops