

AEROO SPACE COMPETITION

Team:Aktau-ESA 5

ТЕХНИЧЕСКАЯ ДОКУМЕНТАЦИЯ

“Satellite safety support system(S4S)”
(Система поддержки безопасности спутников)(СПБС)

Version 1.2V

Автор:Кахарман Уалихан

Казахстан. Регион: Мангистауская область
Город:Актау

Школа:”English school”

Дата:26.02.2026

Научный руководитель:Макенов Е.З

АБСТРАКТ:

Данный продукт является программным обеспечением который включает в себе 4 вспомогательных модуля затрагивающие разные области в безопасности спутников. Такими областями являются(коммуникационная, состояние спутников, поддержка в совершения маневра).

Сами модули:

- 1)Validator - уменьшает ошибки с данными в письмах. Вход:письмо
Выход:указывает на ошибки и неточности
- 2) Handoff - читая данные(переписка или итоги за день) совершает краткий пересказ по каждой части. Вход: База данных с действиями + запрос. Выход: краткий пересказ информации
- 3) Case-Finder - находит схожие прошлые события из Базы данных, определяя паттерны изменений данных и событий. Вход: айди идентификации события. Выход: айди TOP-N схожих событий
- 4)Health Status - изучает состояние спутника, находит возможные причины появлений повреждений, Указывает на опасность и критичность сбоев. Указывает ограничения на маневр
Вход:id/имя спутника. Выход: состояние спутника

Модули помогают и снижают нагрузку (операторам/инженерам, и STM части)

Ключевые параметры в моем софте являются: жёсткие правила и доказательная база для Искусственного Интеллекта(т.е минимизируется ошибки и ложные высказывания от ИИ)

Важно упомянуть, что ИИ в моем ПО является анализатором/валидатором(проверяет и анализирует) информацию, на основе доказательной Базы и порогам, он помогает операторам находить ошибки, и указывает на ключевые проблемы, возможные методы решения, опасность ошибок/проблем, и помогает в части решения маневра.

Данная система предоставляет: 1)поддержку в служебной переписке операторов при возможности Столкновений

- 2) Анализ служебной Переписки, дополнительной информации, формируя итоги действий
- 3) Анализ и поиски проблем в состоянии/здоровья Спутника, поиски возможных причин и методы решения
- 4) Поиск подобных прошлых событий по ключевому событию(для нахождения Схожих паттернов изменений данных, дополнительная информация для совершения действий)
- 5) взаимодействия всех модулей между собой, позволяя действовать сообща и с более большей точностью

Важно уточнить: 1) данные функции не рассчитывают входные данные для расчётов

2) Они **не автоматизируют** процессы совершения маневра или их планового расчета

СТР 2

ТЕРМИНЫ:

Маневр - совершение уклонения спутника от угрозы(Мусора, другого спутника)

CDM(Conjunction data message) - уже рассчитанные данные от радаров, которые включают в себя скорость, шанс столкновения спутников, расстояние минимального сближения

Retrieval - процесс поиска, получения или восстановления информации, данных или объектов

pair_a, pair_b - составляющие(обычно спутники)

Pc(Probability of Collision) - шанс столкновения между объектами(обычно имеют очень низкий шанс)

нормализация — это приведение разных значений к **единому стандарту** или шкале.

Miss distance - расстояние минимального сближения

Матрица Ковариации - погрешность данных, неопределенность.

TCA(Time Of closest approach) = время ближайшего сближения между объектами

debris - космический мусор

парсинг - превращение “сырых” данных в структурированные данные

Satellite name - имя спутника

system message - набор высокоприоритетных инструкций и контекста

Дедуплицировать - это процесс обнаружения и удаления повторяющихся (идентичных) копий данных, при котором сохраняется только один уникальный экземпляр, а дубликаты заменяются ссылками на него

Pair Key - определенный числовой айди для двух пар объектов

Json - текстовый формат

маппинг - процесс установления соответствия между элементами двух различных наборов

эмбединги - способ представления текстов, изображений, с помощью числовых векторов(Важная часть чтобы ИИ понимал все)

Телеметрия - технология автоматического сбора, передачи и анализа данных с удаленными объектами

оркестратор - программа которая координирующий и управляющий сложными компьютерными системами

dominant miss axis - анализ вероятности столкновений спутников

Векторное хранилище - БД, хранит данные в числовых кодах(удобен чтобы писать правила или casecard)

source family - источник данных, откуда он предоставляется, в него входит термины как provider- компания которая предоставляет данные.

catalog - база данных с данными объектов

object_type_pair - тип объекта, может быть спутником, мусором, полезной нагрузкой и т.д

Валидация - проверка соответствия чего-либо установленным стандартам или правилам

Handoff - Краткий и информативный пересказ итога

Триггер - Это условие срабатывания самого модуля

Теперь перейдём к основной части

ОСНОВНОЕ ОПИСАНИЕ ПО

ПЕРВАЯ ЧАСТЬ:(Назначение: зафиксировать, что представляет собой система S4S, ее границы ответственности и какие входные данные/источники используются.

Содержимое: System context (кто взаимодействует, какие сценарии запуска, какие модули существуют) и перечень data sources (операторские вводы, CDM/telemetry, rules/casecards, артефакты).

Результат для читателя: понимание “что это за система” и “на чём основаны выводы”.

Не включает: детали реализации (n8n), схемы БД и алгоритмы модулей.)

SYSTEM CONTEXT (этот раздел описывает окружение системы, участников, источники данных и систему выхода/входа)

1) Взаимодействуют: люди/сервисы(могут вызываться через запрос от человека или триггер от системы(триггерами является: сообщение от системы о проблеме здоровья спутника, запросы оператора и по времени))

2) Входные данные

Интут

Примеры

Источник

Используется

CDM DATA	Общая информация об ситуации, TCA, PC, MISS DISTANCE, SPEED, ковариации.	провайдер/База данных	Handoff, Validator
rules/Casocard (из БД, в postgres PG vector store(neon.tech))	Правила и пороги работы и анализа для ИИ. В casocard примеры как работать с информацией и прошлые ситуации которые уже решены. используется как “библиотека доказательств на примере”	Векторное хранилище	Все модули
CASE IDENTITY	Вводные данные от самого оператора, имя спутника, id	оператор/система	Handoff, Case-finder

	сообщения, данных, информации.		
health status	состояние спутника и его проблемы	База данных/Внутре нее хранилище	health_status_A I(модуль)

3) Выходные данные:

Выход

ФОРМАТ

Куда приходит

Работа валидатора	JSON/card	в чат между оператором и ИИ
Handoff summary	Текст + ключевые события(в виде карточки)	На почту или в чат
Похожесть кейсов TOP-N	Таблица + оценка схожести по баллам. Графики	На почту/База данных/В чат
Состояние спутника/огран ичения	Карточка с аргументацией и таблица с уровнями угрозы	На почту/База данных/В чат

DATA SOURCES(раздел который формально описывает, откуда система берет данные и на чём она основывает выводы)

DS1 - Operator Inputs (человек / запрос)

Тип:текст/структурированные поля

Каналы: чат(n8n workflow)

Содержимое: pair_a, pair_b, pair_key. satellite_name, message_id, case_id, communication_id.

Использует модули: Validator, Handoff, Case finder, healths_status_AI

Ограничения: Может быть не точным запрос, что приводит к неточному ответу

DS2 - CDM Data store (Neon postgres/JSONB)

Тип:Реляционная БД +JSONB

Содержимое:CDM сообщения и цепочки, поля риска: pc, TCA, miss distance, speed, covariance, source family

Использует модули:Validator, Handoff

Ограничения качества: возможны разные провайдеры/форматы/единицы => Нормализация

DS3 - Satellite health Telemetry / Health Snapshots (Neon postgres/JSONB)

тип:Json snapshot + история обновления

Содержимое:Состояние здоровья спутника/его подсистем(power/thermal/comms/prop) Потери/пропуски, числовые состояния

Используют модели: Health_status_AI (и косвенно Case-Finder, для нахождения ограничения миссии)

Ограничения:Задержка данных

DS4 - RULES & knowledge base(pgvector, Neon)

Тип: vector store:эмбединги

Содержимое:Формальные правила/пороги действий, требования к доказательствам и аргументациям, запреты/ограничения, Правила формата вывода(на каждый модуль разный)

Используют модели:Validator, Handoff, Health_Status_AI, Case-finder

ограничения: Retrieval может пропустить правила

DS5 - Casecard/Evidence library(pgvector:casecard_store)

Тип:vector store (embeddings)

Содержимое: Исторические мини кейсы/сценарии: “проблема => риск => что сделали? => результат + потенциальные угрозы” привязка к типам событий

Используют модули: Validator, Case-finder, health_status_AI

Ограничения: casecard - не факт истины, а лишь живой пример. Нельзя выдавать причину без привязки доказательствам

DS6 - Historical cases/Схожие события(Neon postgres)

тип: Таблицы кейсов + метрики/удобные записи

Содержимое: Исторически пройденные события, записи/временные ряды, pair_key object_type_pair

Использует:Case-finder

Ограничения:не всегда есть полные цепочки информации, пропуски и “дыры”

DS7 - Generated Artifacts(Outputs stored in DB)

Тип:отчеты/артефакты исполнения

Содержимое: run_id, ruleset_version, Validator_report, similarity_results, Health_verdict, handoff_summary, финальный формат ответа, логи ошибок

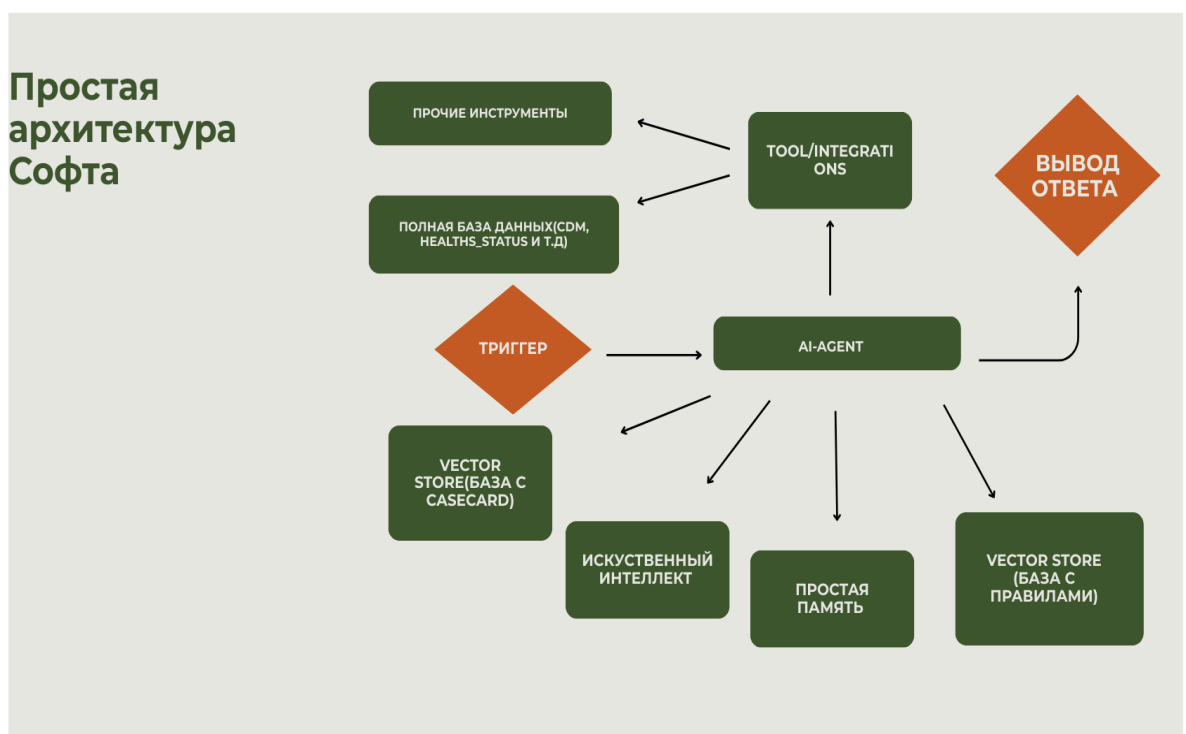
ВТОРАЯ ЧАСТЬ: (Назначение: описать техническое устройство системы на уровне компонентов и данных.

Содержимое: архитектурная схема (компоненты, связи, high-level flow) и data model description (сущности, ключи идентификации, связи, что хранится в Neon Postgres и в vector stores).

Результат: читатель понимает, где лежат данные, как они связаны, и как проходит обработка от триггера до результата.

Не включает: полный список правил/порогов и подробности каждого n8n-узла.)

АРХИТЕКТУРА(построение системы/воркфлоу)



Представлена простая схема работы Софта

1)Начало/Запуск программы = триггер(When message received/schedule trigger/Gmail event)

2) AI-agent - оркестратор и транзиттер между ИИ с БД и Инструментами(tool), использует сам ИИ и выполняет его запросы

(части связанные с модулем)

2.1) Искусственный интеллект, получает System message и задачу от модуля, после, выполняет задачу

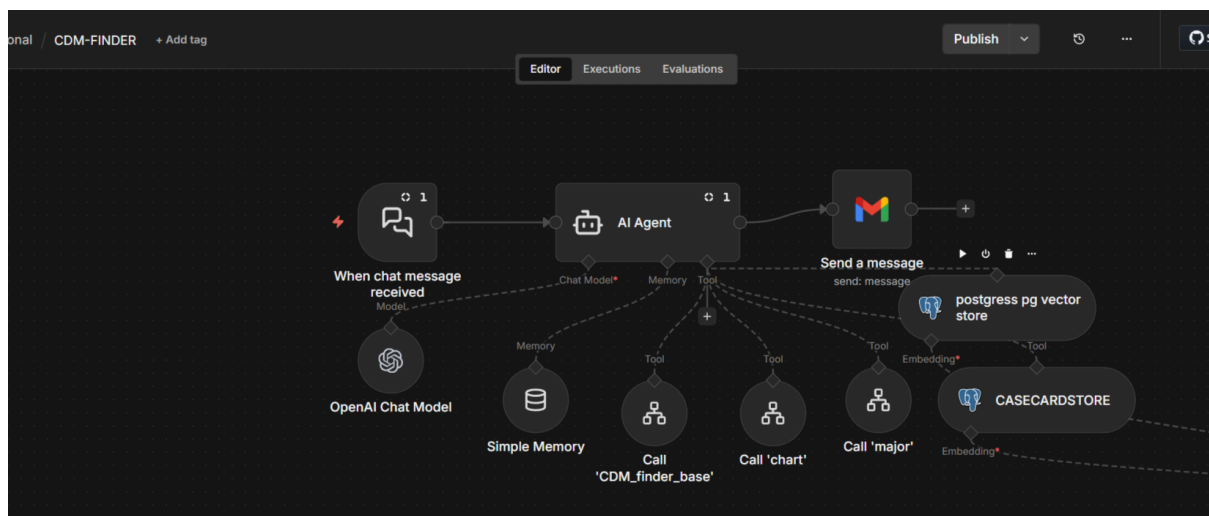
2.2) Простая память - память ИИ в диалоге(если оператор решит продолжать запросы, чтобы ИИ не потерял контекст и смысл всего)

2.3) VECTOR STORE(База с правилами и CASECARD) - База правил и порог, показательные примеры с обучением ИИ.

2.4) TOOL - инструменты ИИ для выполнении задачи и получения Базы данных для анализа.

3) Вывод данных - Выход полного ответа. Приходит либо в GMAIL почту, либо в чат переписки

НАСТОЯЩИЙ ПРИМЕР WORKFLOW



- Триггером тут является чат с ИИ(When chat message received)
- AI agent является оркестратором
- openAI Chat model является ИИ
- Postgress pg vector store/CASECARDSTORE как раз База с правилами и кейсами
- call 'CDM_finder_base' call 'chart' call 'major' являются инструментами для ИИ
- Выходом является отправкой результата на почту и такой же ответ результата в чате

Data Model - (раздел где фиксируют структуру данных)

Data model overview:

Таблицы о которых будет описание: CDM_message, CDM_source_features, Difference, messages, health_status, rules, casecard

Ключевые идентификаторы: “message_id”, “pair_key”, “time_id”, “communication_id” “rule_number”, “casecard_number”

Все эти объекты находятся в общем БД(Neon.tech) каждые объекты находятся в разных дочерних БД(для того чтобы не пересекались значения при поиске) хранятся в “jsonB”, текстом и таблицами

DATA MODEL DESCRIPTION(описание каждой таблицы)

1) CDM_message(Дочерняя БД является “VALIDATION”:

- **Цель: Хранить и показать метаданные от радаров, указать имена объектов которые имеют потенциальное сближение(Для валидации и дальнейших расчетов)**
- **Ключ идентификации:message_id**
- **Колонны:**
 - 1.1) created_at - когда было отправлена данные в БД**
 - 1.2) message_id - ключ идентификации**
 - 1.3)cdm_json - сами метаданные по объектам и ситуации, в формате JSON**
 - 1.4)creation_date - Когда данные вообще создались**
 - 1.5)Pair_a(имя ключевого спутника) pair_b(имя спутника/мусора/другого объекта)**
- **Индексы: CDM цепочку можно найти по объектам, pair_a, pair_b. а цепочка от самого первого CDM, до самого последнего можно провести по дате, creation_date**

2) CDM_source_features(Дочерний БД является “Calculaction”):

- **Цель:**Предоставляет информацию об CDM,его каталоге, провайдере, источнике данных для каждой пар объектов, типы объектов и способы расчетов (Является составной части фильтрации схожих case)
- **Ключ идентификации:**Pair_key
- **Ключевые колонны:**
 - 2.1)Message_id - Айди CDM которую рассматривают
 - 2.2)Provide - тот кто предоставляет данные
 - 2.3) signature - набор характерных признаков в данных, по которому распознают событие/объект/кейс
 - 2.4) catalogs
 - 2.5) ephemeris - это набор орбитальных данных, который задает положение (и часто скорость) объекта во времени
 - 2.6) Covariance_method - метод расчета ковариации
 - 2.7)maneuverable - возможность маневрировать объектов, для каждого
 - 2.8)object_type_pair - тип объектов, Мусор/Спутник/Полезная нагрузка
 - 2.9)dominant_miss_axis_ric - доминирующая ось RIC, по которой модуль относительного смещения на ТСА максимален.
- **Индексы:**pair_key, message_id (можно расставить данные pair_key по дате создания)

3) Difference(Дочерний БД является “Calculaction”)

- **Цель:** показать разницу значений данных между CDM, показывает эволюцию изменений данных к приближению до ТСА
- **Ключ идентификации:**pair_key, pair_a, pair_b, From_cdm_id, To_cdm_id
- **Колонны:**
 - 3.1)From_cdm_id - от чего сравниваем, to_cdm_id - с чем сравниваем
 - 3.2)old_values - значения метаданных CDM у From_cdm_id
 - 3.2)new_values - значения метаданных CDM у To_cdm_id
 - 3.3) Delta - разница значений(числовая)

3.4)Delta_ptc - процентная разница значений

3.5)Freshness - указывает старость CDM которые сравниваются. где 1 самый старый, и чем больше число тем новее

3.6)delta t - разница времени появления данных между двумя CDM

- Индексы:pair_key, Freshness. Pair_a, pair_b

4) Messages(Дочерняя БД “handoff”)

- Цель:Хранить и показывать письма операторов между друг другом, для каждого события
- Ключ идентификации: id(указывает на айди события переписки) communication_id(Айди сообщения)
- Колонны:

4.1)Id

4.2)communication_id

4.3)letter - само письмо оператора(текстовый формат)

4.4)created_at - дата отправки/Получения письма

4.5)Operator - операторы, чье является письмо, кто написал

4.6)operator change - дополнительная колонка указывающая на смену операторов

Индексы:id, communication_id(по дате так же можно сортировать)

5) Health_Status(Дочерняя БД является “FEASIBILITY”)

- цель: Хранить и показывать состояние спутника(хранится не сырые данные, а уже рассчитанные)
- ключ идентификации:time_id

5.1) time_id - айди идентификации времени, Если новые данные о спутнике придут в течении трех часов, то time_id будет постоянным, если >3 часов, то меняется time_id

5.2)info - сама информация о состоянии спутника

5.3)creation_date - дата создания Самого первого info

5.4)warning - предупреждение об аномальных значениях или же сбоев

5.5)satellite - имя рассматриваемого спутника

5.6)update_info - показывает изменения значений между обновлениями, и время обновлений

6) Rules(Находится во всех дочерних БД, все обозначены именами их бд)

- цель: дать правила и инструкцию ИИ
- Колонны являются: id, text, metadata, embeddings. Самим прочесть и понять трудно, ведь это цифровой вектор для ИИ

7) Casecard(Аналогично находится во всех дочерних БД, все обозначены именами их БД)

- цель: Показать примеры и практические части, показать алгоритм решений проблем
- Колонны такие же как у rules.

ТРЕТЬЯ ЧАСТЬ: ОПИСАНИЕ МОДУЛЕЙ

(Назначение: описать работу каждого core-модуля как отдельного функционального блока.

Содержимое: для Validator-AI, Case Finder, Health_status_AI, Handoff — триггеры, входы/выходы, условия деградации/ошибок и формат отчёта (findings + severity + evidence).

Результат: понятно, что делает каждый модуль, какие данные требуются и какой артефакт он генерирует.

Не включает: полный нодовый разбор n8n (только на уровне логики/алгоритма))

Схема работы каждого модуля

Триггер=>Инпут=>Нормализатор=>Загрузка векторного хранилище =>Работа ИИ=>Выход=> предоставление доказательств + аргументация

1) Validator-AI

Триггер: после сохранения черновика/отправки валидатору

Инпут: Само письмо, CDM из базы данных.

Нормализатор: Берет тот идентифицированный CDM который оператор предоставляет в письме(извлекается по контексту) (CDM_ID)(упорядочивание цепочки) парсинг данных, приведение времени, единиц и осей,

Загрузка векторного хранилища: Загрузка актуальной версии правил и порог проверки письма, инструкция анализа письма и его валидации

Работа ИИ: Читает и анализирует текст после сохранения письма оператором, ищет нестыковки данных и неточность формулировок. НЕ пишет за человека письмо.

Схема работы:

- Получить письмо (draft/send event)
- Извлечь идентификаторы CDM из письма (Message_id/pair_key/объекты+ТСА).
- Загрузить соответствующий CDM/цепочку CDM из БД.
- Нормализовать CDM поля (время UTC, единицы, оси, типы данных).
- Retrieve актуальных правил/порогов для валидации письма (rules_store).
- Применить правила: найти нестыковки, некорректные числа/формулировки, пропуски обязательных элементов.
- Сформировать findings с severity и evidence (ссылки на поля CDM/правила).
- Отдать результат оператору (и опционально сохранить артефакт в БД).

Выход: Предупреждение оператору об ошибке и указание на проблему, уровень ошибки(TINY/MAJOR/CRITICAL)предоставляет доказательства и аргументацию на основе правил(Rule_number N) и Базы данных

Возможные ветки отказа:При отсутствии CDM в Базы данных идет предупреждение о отсутствии такого CDM по определенному id и последующей невозможности проверки. При отсутствии доступа Правил порога, аналогично прошлому идет предупреждение о невозможности проверки.

2) Handoff Summary

Триггер: Запрос оператора пересказа на определенную смену(id). Или переписки(communication id = идентификация письменного обмена)

Инпут: Внутреннее хранилище данных писем, заметки и действий

Нормализатор:Парсинг данных, упорядочивание писем/заметок по дате и времени

Загрузка векторного хранилища: Retrieval актуальной версии правил и инструкция по анализу Инпута, формулировка и формат передачи краткого и информативного пересказа

Работа ИИ: Рассмотрение всех писем/сообщений, поиск ключевых слов, изучение статуса операторов, смены операторов, основной и важной информации о ситуации и маневров.

Схема работы :

- **Trigger:** запрос оператора / конец смены / расписание.
- **Scope:** определить communication, окно времени, связанный pair_key/message_id (если есть).
- **Load:** собрать все сообщения треда (текст + timestamps) из канала/БД.
- **Normalize:** отсортировать, дедуплицировать, выделить updates/decisions/open items.
- **Retrieve template:** загрузить шаблон handoff + правила структуры (из vector store).
- **Assemble summary:** статус → ключевые обновления → решения → открытые вопросы → next actions → refs (message_id/cdm_id).
- **Output:** выставить confidence, сохранить в БД (опц.) и отправить в chat/Gmail.

Выход: Краткий, информативный пересказ служебной переписки, которая показывает: изменения, риски, совершенные действия, решения и статусы

Возможные ветки отказа: При отсутствии переписки(отсутствие данных идентификации) или недостаточной количество данных для анализа и пересказа.

3)Case-Finder

Триггер: При запросе оператора.

Инпут: Состояние здоровья спутника, доступ к CDM данным и расчетам маневра, имеет доступ к информации об ограничениях спутника. имя каталога, имя спутника и его id, оператор должен

предоставить pair_a pair_b, на каждое событие имеется pair_key(ключ события)

Нормализатор: Данные привязываются к номеру идентификации кейса
“выбор reference_case(reference_case_id)”. Упорядочивание CDM по дате.

Загрузка векторного хранилища: Retrieval актуальной версии правил и инструкция по анализу кейсов, алгоритм нахождения схожести, и Доказательная база таковыми является

Работа ИИ: Идентификация аналогичных ситуации методом отсева для прогноза изменений обстановки/данных, и поиска эффективных предпринятых мер

Схема работы:

- Trigger: запрос оператора / системный вызов
- 1. Resolve reference: определить reference_pair_key(или pair_a/pair_b) + взять последний CDM/цепочку из БД.
- 2. Normalize features: привести время/единицы/оси, собрать ключевые части (Pc,TCA,miss,v_rel, covariance meta, source_family).
- 3. Retrieve context: загрузить актуальные rules + casecards (pgvector).
- 4. MAJOR gating: отфильтровать кандидатов по exact match: source_family, maneuverable_bucket, object_type_pair (+ желательно dominant_miss_axis).
- 5. Score & rank: посчитать similarity score, сформировать breakdown, выбрать TOP-N, Создать аргументацию на

основе доказательств

- 6. Output: вернуть TOP-N +
major_pass/missing_flags/confidence (+ опц. график) и
сохранить артефакт/отправить ответ с доказательствами.**

**Фильтрации в два этапа(Major match), сравнивают и ищут
события которые должны иметь обязательные совпадения с
данными которые определяют базовую схожесть:
source_family(catalog/provider) object_type_pair
dominant miss axis**

**Во втором этапе(AI-analyst) Анализация по схожести
трендов/эволюции данных, сами значения, ситуации и
ограничения.**

**Выход: Таблица с TOP-N схожих кейсов по действующему
событий, сравнительный анализ с действующим событием и с
схожими кейсами.Оценивание схожести по 100 Балльной
шкале**

**Возможные ветки отказа:Недостаточное количество данных, не
тот провайдер или иной источник данных. Слишком большая
матрица ковариации. Отсутствие аналогичных событий(ниже
порог 60 баллов) .**

4) Health_status_AI

**Триггер:Срочный запрос оператора, запуск по
времени(каждые 3-4 часа), запуск при выявлении критичных
проблем/сбоев/поломок (нарушение питания, батареи,
терморезимы и т.д)**

**Инпут:Полное состояние спутника и его имя. База данных
по конфликтующим сочетаниям(сочетания которые ухудшают
состояние спутника многократно при совместной поломке),**

взаимозависимым отказам(если появляется один сбой, то появится и второй)

Нормализатор: парсинг ключевых ошибок и проблем которые стоит рассмотреть в первую очередь, расстановка обновлении данных по дате

Загрузка векторного хранилища: Retrieval актуальной версии правил по выставления уровня опасности, и проверок, инструкция по обнаружении критических сбоев, поиск исторических похожих поломок, и Доказательная база.

Работа ИИ:Анализ состояния и последующей выявлении проблем, присвоение уровня критичности для поломок(CRITICAL/MAJOR/WARN), выявления ограничения для спутника(в миссий или маневра) выявление предположительных причин сбоев, на основе аргументов которые подтверждены, поиск возможных решений проблем.

Схема работы:

- **Trigger:** запрос оператора / расписание (каждые 3–4 часа) / событие критического сбоя.
 - **Load snapshot:** взять `satellite_name` и актуальный `health snapshot` (и при необходимости историю за окно времени) из БД.
 - **Normalize:** проверить полноту/свежесть данных (`staleness`, пропуски), привести поля к единому формату, собрать `missing_fields[]`.
1. **Retrieve rules:** загрузить правила/пороги + базы зависимых/конфликтующих отказов (`retrieval`).

2. **Evaluate:** применить правила к подсистемам, определить severity по каждому finding, выявить зависимости (если “А => В”).
3. сформировать ограничения для миссии/маневра (например forbid_burn, comm_unavailable).
4. **Output:** выдать карточку состояния (verdict + findings + evidence + причины/действия только при evidence), указать confidence, по желанию сохранить/отправить.

Выход: Карточка/таблица показывает состояние спутника, отдельная карточка для каждого уровня критичности демонстрирующий опасность каждой поломки, доказательства и аргументации на потенциальную опасность, возможные причины и что возможно сделать для минимизации влияния поломки.

При стабильном состоянии спутника предоставляет доклад о полной работоспособности спутника

Возможные ветки отказа: Отсутствие спутника в базы данных, устаревшая телеметрия, пропуски и потери данных(снижения доверия/ограничения в способности ИИ)

ПРАВИЛА И ПОРОГИ(Этот раздел будет посвящен объяснений правил и порогов, как они работают)

Все правила находятся в формате эмбедингов, ведь это удобно для Искусственного интеллекта и увеличивает их точность.

Стоит упомянуть что правила для каждого модуля разные, но они имеют общие черты и принципы.

общие принципы:

- Система выдает findings только по правилам
- Каждое finding обязано содержать rule_id + severity + evidence_fields + evince values.
- если evidence не хватает → то пишется “не достаточно доказательств” (а не “предположения”).
- Правила имеют версию и айди

Уровень “Важности/критичности” и составления вердикта, правила маппинга

Для всех правил есть общий уровень

Severityt level: (INFO/WARN/MAJOR/CRITICAL) (слева направо у важность/критичность)

Вердикт(итог модуля) обычно идет через правило маппинга:

- PASS - нет WARN/CRITICAL/MAJOR
- WARN, присутствует WARN, но отсутствует CRITICAL/MAJOR
- PROBLEM/FAIL: есть MAJOR или CRITICAL.

Теперь рассмотрим Ключевые правила для каждого модуля

Case-finder rules звучат так:

- 1) При первой фильтрации кандидатов, выбирать только те кейсы, у которых [major.ok](#) = true, иначе отсеивай.
- 2) Взять аргументации и доказательства из данных почему [major.ok](#) = false, почему [major.ok](#) = true
- 3) Выполнить сравнительный анализ с reference case с помощью CASE-finder base, посчитать score 0-100 с разбивкой на эволюцию значений (70%) и числовые показатели (30%), выбрать TOP-3 кандидатов, сформировать доказательства в виде trace points и mass note."

- 4) Веса для эволюции (%): { "S_cov_trend": 0.50, "S_miss_trend": 0.08, "S_pc_trend": 0.08, "S_other_trend": 0.04 } (Тут говорится о правилах расчета и анализа схожести эволюции изменений)
- 5) "Веса для числовых показателей (%): { "S_miss_num": 0.14, "S_cov_num": 0.05, "S_pc_num": 0.06, "S_vrel_num": 0.03, "S_lead_num": 0.02 } }, (Тут говорится о правилах расчета и анализа схожести числовых значений)
- 6) Создание графиков изменений значений для ТОП-3 Кейсов(изменения pc, miss distance, speed)
- 7) Создать итоговый ответ по TOP-3 с детальным обоснованием на основе правил major и Case-finder base, сформировать рекомендации и вызвать построение графиков.

Все данные правила основаны на использовании “tool”, и анализа схожести в сравнительной части и аргументации.

Health_status_AI rules:

- 1) Правило срабатывает При условии нарушений лимитов(значения вне допустимой нормы). Значимость высокая добавляет 25 баллов к severity. Ограничения: остановка операций, связанных с подсистемой с нарушением, нужна детализация параметров.
- 2) срабатывает при флаге forbid_burn = true(запреты на маневры/импульса), Значимость очень высокая, добавляет 40 severity баллов. Ограничения: полный запрет на маневры/импульсы, нужна диагностика причины
- 3) Срабатывает при включенном safe_mode, Критическая значимость, +35 баллов к severity, ограничения нагрузки, запрет маневров до стабилизации.

В общем, правила покрывают границы допустимых параметров, запреты на маневры и burn, состояние связи,

режим `safe_mode`, а также жёсткий регламент представления отчётов.

Validator rules:

- 1) Решение о типе сообщения: Если `Historical_comparison = true` и `action_intent = false` - Сообщение считается историческим, если заявлено `latest_claim = true` или `action_intent = true` сообщение считается операционным с последними данными(если не самые последние) то это просто справочная информация
- 2) Проверка актуальности данных, правило всегда выполняется, определяется является ли CDM устаревшим, если есть более новые CDM в базе данных, возвращаются поля с ID и датами создания `referenced CDM` и более новых CDM
- 3) Правила сравнений значений: Определенные пороги для сравнений и расхождений значений по разным параметрам(`TCA`, `PC`, `MISS DISTANCE`, `SPEED`) При превышении порогов срабатывают предупреждения **MAJOR/CRITICAL**
- 4) Источники **MAJOR** находок включают ошибки в конвертации единиц, значительные несоответствия данных и т.п
- 5) Источники **CRITICAL** находок: Включают в себя неоднозначность в формате времени, противоречия в намерениях маневра, отсутствие идентификаторов и другие
- 6) логика решения по итогам проверки: **CRITICAL** - необходимо исправить ошибки, отправка будет ограничена(из за потенциальной угрозы ухудшения ситуации), **MAJOR**-Можно отправить при срочности, но рекомендуется исправить и перепроверить. **INFO/TINY**-можно отправлять, рекомендуется учесть для повышения ясности

HANDOFF RULES:

- 1) Смена операторов фиксируется, если operator_id меняется в следующем письме
- 2) Изменения статуса выявляется по ключевым словам, таким как confirmed, awaiting, corrected, error
- 3) Анализ отслеживает статусы двух операторов в каждом сообщении, определяет итоговый статус, регистрирует смену оператора и причину изменения статуса
- 4) Извлекает все упоминания CDM из сообщений, где упоминаются оба объекта соответствующей пары.
- 5) Формирует список CDM с флагами по использованию и статусу передачи для каждого оператора
- 6) Сообщения анализируются на предмет переключения оператора и ключевых подтверждений по CDM.

Были показаны настоящие правила модулей(Представлены не все)

CASE CARD Examples

VALIDATOR CASE CARD	<p>CASE_ID: EV-OPS-0130 Subject: "ACTION: MIA-X01 vs 1998-067A (decision)" RAW: "TCA 12:40 tomorrow, miss 0.42 km, Pc=1e-4, GO/NO-GO in 2h, CDM attached." FINDINGS: FAIL — TCA без даты/таймзоны; нет CDM_ID/Event_ID в тексте; дедлайн "2h" не привязан к UTC; miss без типа (3D или RIC). FIX: указать TCA=2026-02-28T12:40:00Z, добавить CDM_ID/Event_ID + created_utc, задать decision_deadline_utc=...Z, уточнить miss_3D_km или miss_RIC_m. NORMALIZED: "MIA-X01 vs 1998-067A CDM_ID=... (created ...Z) TCA=...Z Pc=0.0001 miss_3D_km=0.42 decision_deadline_utc=...Z GO/NO-GO?" (краткий формат case card)</p>
HANDOFF CASE CARD	CASE_ID: HOFF-EMAIL-EX-012 TYPE:

	<p>handoff_email PRIORITY: MAJOR WINDOW: 2026-02-27 18:00Z–22:00Z ASSET: SAT-MIA-X01 CONTEXT: в 20:40–21:10Z связь деградировала, команды подтверждаются с задержкой FACTS: link_state=DEGRADED, 2× NO_LOCK по 20–30с, cmd_ack_delay ~45с, staleness ~38с DECISION: с 20:55Z остановлены non-critical команды до стабилизации RISKS: повтор NO_LOCK → риск потери командования 5–10 мин, нужен safe-mode триггер TODO: (1) монитор link_state+ack каждые 5 мин до 23:30Z (TM snap ID#4471) (2) связаться с COMMS on-call, тикет COMMS-219 до 22:30Z (3) если NO_LOCK суммарно >90с/15мин → запуск SOP S-12 немедленно DRAFT: “Тема: Handoff SAT-MIA-X01 — COMMS деградация. За смену: DEGRADED, 2×NO_LOCK, ack ~45с. Non-critical команды на паузе. Дальше: монитор до 23:30Z, COMMS-219 до 22:30Z, при >90с/15мин — SOP S-12.”</p>
CASE-FINDER CASE CARD	<p>CASE_ID: CF-LEO-0021 INPUT (reference): pair_key=MIA-X01 1998-067A, last_CDM: TCA=2026-02-28T12:40:00Z, Pc=1e-4, miss_3D_km=0.42, v_rel=12.1 km/s, maneuverable=YES, obj_type_pair=PAYLOAD vs DEBRIS, dominant_axis=I MAJOR_MATCH (фильтр): same source family (catalog/provider/cov_method/ephem_class), same maneuverable bucket, same object type pair, близкий TCA bucket, желательно same dominant axis OUTPUT (TOP-3): 1) CASE_1098 score=0.86 (Pc±30%, miss±20%, same axis I, same obj types) 2) CASE_0774 score=0.81 (Pc close, v_rel close, miss чуть меньше) 3) CASE_1412 score=0.78 (same source family + maneuverable, но axis другой) WHY_SIMILAR: совпали MAJOR-факторы + близкие Pc/miss/v_rel тренды по последним 2 CDM USE: взять из CASE_1098 “какой порог/окно решения” и “какие данные уточняли перед GO/NO-GO” (без копипасты манёвра, только шаблон действий/проверок).</p>
HEALTH_STATUS_AI CASE CARD	<p>CASE_ID: HEALTH-AI-EX-021 TYPE: health_status_ai PRIORITY: CRITICAL ASSET: SAT-MIA-X01 FACTS: COMMS NO_LOCK→comm_available=false; PROP thruster_ready=false + valve INHIBITED→forbid_burn=true; DATA staleness=44s, loss=3.6% VERDICT: burn запрещён, состояние деградирует RISKS: команда burn при inhibit / потеря командования при повтор NO_LOCK ACTIONS: запретить burn (now), эскалация PROP+COMMS (≤30 мин), при</p>

	NO_LOCK>90с/15мин или staleness>60s → SOP safe-mode.
--	--

ЧЕТВЕРТАЯ ЧАСТЬ:(Назначение: доказать работоспособность системы и воспроизводимость результатов.

Содержимое: пошаговый walkthrough запусков, примеры входов, скриншоты workflow и outputs, таблица “Input → Rule → Evidence → Output”, а также ссылки на репозиторий/материалы для воспроизведения.

Результат: читатель видит реальный запуск, реальные результаты и понимает, как повторить демо.

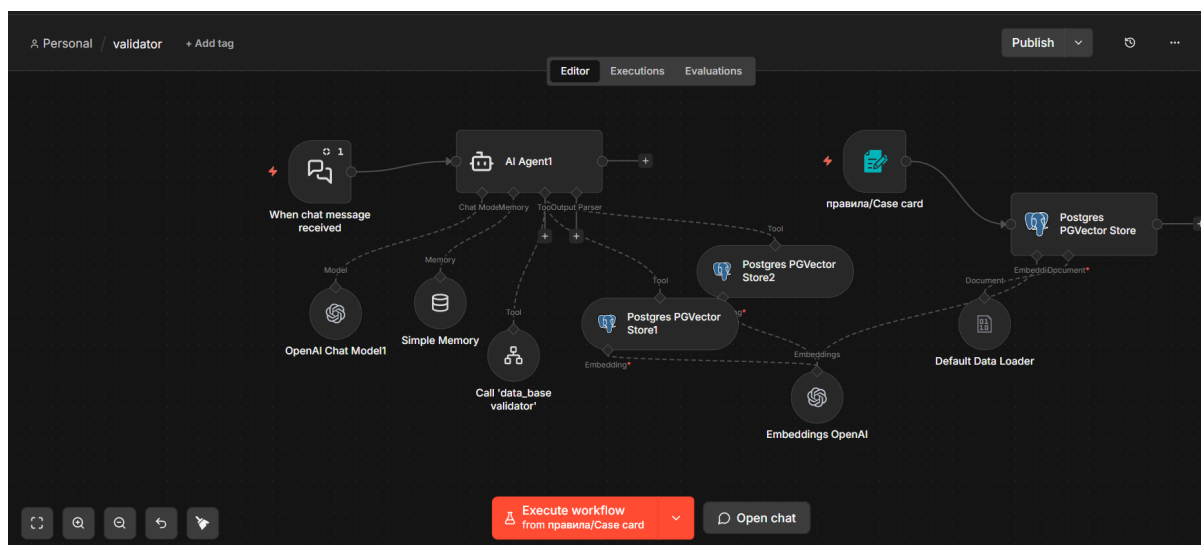
Не включает: расширенные детали инфраструктурного развёртывания (они в GitHub README).

ДЕМО/Работоспособность(Цель этого раздела показать работоспособность моих модулей)

Где запускается все модули, в платформе N8N

Источники задействованы(Neon.tech, Postgress)

1) Module:Validator AI



Триггер: When chat message received, нужно написать письмо другому “оператору”, предоставив нужные и точные информационные данные. Обязательно в письме нужно писать CDM message_id, creation_date, Имя спутников, идентифицирующие id.

Пример: “Hello CASPIAN-LX2 Operations Team,

We are notifying you of a potential close approach between our spacecraft AKTAU-NEO-1 and your spacecraft CASPIAN-LX2 per CDM (CCSDS CDM v1.0) generated by AKTAU-OPS.

Event summary (from CDM message_id: 941700101, creation_date: 2025-07-01 19:51:34 UTC):

- **Primary: AKTAU-NEO-1 (PAYLOAD, LEOLAB, maneuverable: yes), ephemeris: EPHEM-AKTAU-NEO-1-20250701**
- **Secondary: CASPIAN-LX2 (PAYLOAD, SATCAT, maneuverable: yes), ephemeris: EPHEM-CASPIAN-LX2-20250701**
- **TCA: 2025-07-06 01:47:17.618 UTC**
- **Miss distance: 1185 m**
- **Relative speed: 14610 m/s**
- **Collision probability (Pc): 1.98e-5 (method: FOSTER-1992)**
- **Relative position at TCA (RTN, m): N=+845.3, R=+34.6, T=-78.1**

- Relative velocity at TCA (RTN, m/s): N=-1422.8, R=-7.4, T=-14541.2
- Screening volume: RTN ellipsoid (X=220 m, Y=1000 m, Z=1000 m); entry/exit around TCA: 01:47:17.202–01:47:17.836 UTC

Request / coordination:

1. Please confirm that you are the responsible operator for CASPIAN-LX2, and provide your preferred coordination contact (email/phone) for this event.
2. Please share your most recent OD solution / ephemeris + covariance assumptions (or an updated CDM if available).
3. Please advise whether any maneuver is planned or constrained for CASPIAN-LX2 within the screening period leading up to TCA.

We will continue monitoring and can exchange updates on an agreed cadence until TCA (e.g., daily, then more frequently inside 24 hours). If you confirm your latest orbit solution timing, we can align update windows to reduce discrepancies.

Best regards,

AKTAU ORBITAL — Conjunction Desk / FDO

AKTAU-NEO-1 Ops Contact: ops@aktau-neo.space | +7 7122 450101”

Tool: “call ‘data_base_validator’ Это базы данных с CDM.

При выходе ИИ должен сказать итоги анализа Письма, какие ошибки присутствуют, какой их уровень severity. При отсутствии ошибок он уведомит(в формате отчета, текстом)

ChatSession: 22785...

Отчет проверки текста:

CRITICAL:

- Отсутствуют обязательные идентификаторы NORAD, COSPAR, Designator или другие уникальные каталожные номера для обоих объектов AKTAU-NEO-1 и CASPIAN-LX2. В письме присутствуют только имена и описание типов (PAYLOAD, LEOLAB, SATCAT), без указания официальных идентификаторов. По правилам это критическая ошибка (Правило R1).
- Несмотря на наличие CDM message_id (941700101), отсутствуют сведения о идентификаторах спутников в одном из столбцов, что затрудняет однозначную привязку события.
- Без этих идентификаторов невозможна автоматическая сверка с базой данных CDM или спутниковыми каталогами.

MAJOR:

- Все числовые поля указаны в СИ единицах (метры, метры в секунду), формат времени TCA задан с точностью до миллисекунд в UTC, что соответствует требованиям.
- В тексте запроса содержится призыв предоставить данные о маневрах, орбитальном решении и ковариации, что отражает корректный операционный запрос.

INFO:

- В письме не указаны конкретные NORAD или COSPAR IDs для объектов, что снижает качество коммуникации и удобство последующей обработки.
- Отсутствует информация о текущем статусе маневров (запланирован / нет маневров), это стоит уточнить в дальнейших сообщениях.
- Нет прямых ссылок или приложений с обновленным CDM или OD решениями, следует поддерживать обмен обновлениями для оптимальной координации.

ИТОГО:

- Критические ошибки по отсутствию уникальных идентификаторов спутников (NORAD ID / Designator / COSPAR) для обеих сторон.
- Рекомендуется дополнить письмо этими данными перед дальнейшей рассылкой и сверкой с базой.
- Ошибка в числовых значениях данных CDM (TCA, Pc, Miss Distance) при условии правильности CDM ID не выявлено.
- Отсутствуют MAJOR ошибки по формату единиц и запросам.
- INFO рекомендации по улучшению полноты коммуникации в части маневров и обновленных данных.

CRITICAL проблемы присутствуют.

Выход:

Отчет проверки текста:

CRITICAL:

Отсутствуют обязательные идентификаторы NORAD, COSPAR, Designator или другие уникальные каталожные номера для обоих объектов AKTAU-NEO-1 и CASPIAN-LX2. В письме присутствуют только имена и описание типов (PAYLOAD, LEOLAB, SATCAT), без указания официальных идентификаторов. По правилам это критическая ошибка (Правило R1).

Несмотря на наличие CDM message_id (941700101), отсутствуют сведения о идентификаторах спутников в одном из столбцов, что затрудняет однозначную привязку события.

Без этих идентификаторов невозможна автоматическая сверка с базой данных CDM или спутниковыми каталогами.

MAJOR:

Все числовые поля указаны в СИ единицах (метры, метры в секунду), формат времени TCA задан с точностью до миллисекунд в UTC, что соответствует требованиям.

В тексте запроса содержится призыв предоставить данные о маневрах, орбитальном решении и ковариации, что отражает корректный операционный запрос.

INFO:

В письме не указаны конкретные NORAD или COSPAR IDs для объектов, что снижает качество коммуникации и удобство последующей обработки.

Отсутствует информация о текущем статусе маневров (запланирован / нет маневров), это стоит уточнить в дальнейших сообщениях.

Нет прямых ссылок или приложений с обновленным CDM или OD решениями, следует поддерживать обмен обновлениями для оптимальной координации.

ИТОГО:

Критические ошибки по отсутствию уникальных идентификаторов спутников (NORAD ID / Designator / COSPAR) для обеих сторон.

Рекомендуется дополнить письмо этими данными перед дальнейшей рассылкой и сверкой с базой.

Ошибок в числовых значениях данных CDM (TCA, Pc, Miss Distance) при условии правильности CDM ID не выявлено.

Отсутствуют MAJOR ошибки по формату единиц и запросам.

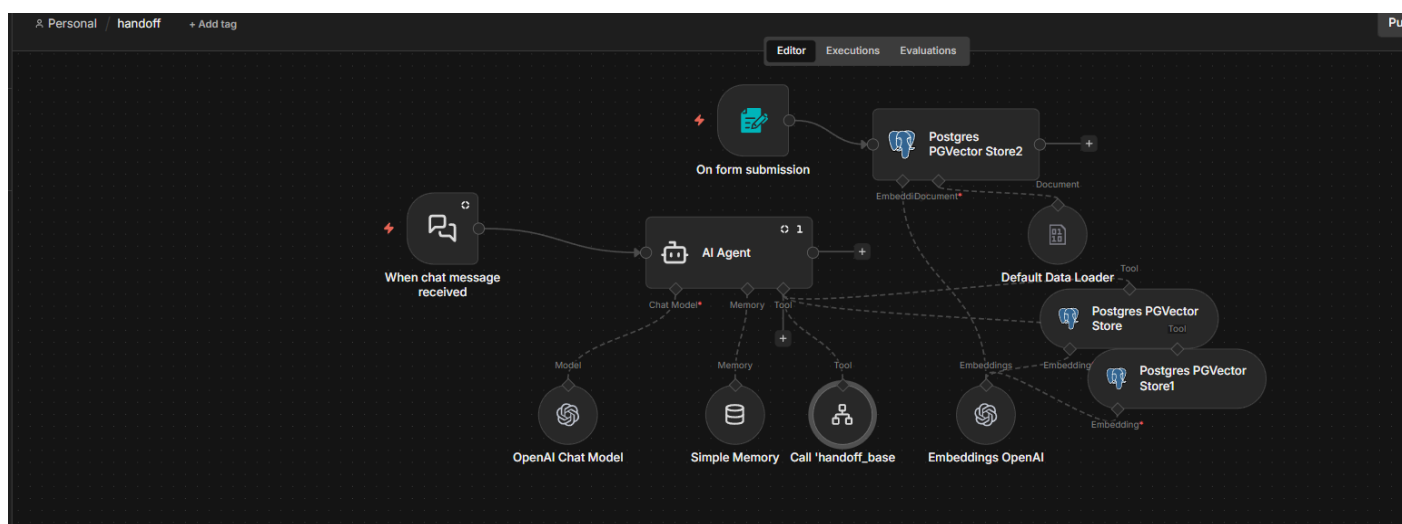
INFO рекомендации по улучшению полноты коммуникации в части маневров и обновленных данных.

CRITICAL проблемы присутствуют.

2)Module:Handoff

Триггер:When chat message received, Нужно предоставить идентификационные сведения для того чтобы ИИ смог понять для чего писать handoff.

Пример: “Мне нужен Handoff по переписке между операторами LUNASAT и MIA-CORPORATION, айди переписки ID = 1”



Tool: Call 'handoff_base' Инструмент в котором находится база данных переписок их айди идентификации.

При выходе ИИ предоставит информативный и короткий пересказ об действии и договоренности в письмах(в формате отчета, текстом)

Chat
Session: 24799...

Summary:

- Объем обсуждения: Координация маневров и обработка CDM между операторами LUNASAT и MIA-CORPORATION по событию сближения CDM 9850145153.
- Переходы операторов (если есть): оператор A — LUNASAT инициировал координацию и направил обновления; оператор B — MIA-CORPORATION подтвердил получение и уточнял вопросы.
- Обмен сообщениями (кратко по хронологии):
 - LUNASAT передал исходные данные CDM и запросил подтверждение возможности маневрирования.
 - MIA-CORPORATION подтвердил получение и уточнил ответственность и сроки маневров.
 - LUNASAT предложил координацию с заморозкой несущественных маневров и инструкции по пересылке новых CDM.
- Ссылки на CDM/обновления: Активно используется CDM 9850145153, который координирует LUNASAT; CDM 9850145154 выявлен как неиспользуемый и требует внимания.
- Новые CDM: новых CDM, кроме упомянутых, не обнаружено.
- Оперативный статус (наша сторона vs контрагент): LUNASAT ожидает подтверждения; MIA-CORPORATION подтвердил отсутствие необходимости в маневрах.
- Запланированные действия:
 - LUNASAT контролирует подтверждения и замораживает маневры.
 - Решить вопрос по статусу CDM 9850145154.
- Ожидания / пункты мониторинга:
 - Ожидание подтверждения статуса маневров от MIA-CORPORATION.
 - Уточнение обработки не переданного CDM 9850145154.

Выход:Объем обсуждения: Координация маневров и обработка CDM между операторами LUNASAT и MIA-CORPORATION по событию сближения CDM 9850145153.

Переходы операторов (если есть): оператор A — LUNASAT инициировал координацию и направил обновления; оператор B — MIA-CORPORATION подтвердил получение и уточнял вопросы.

Обмен сообщениями (кратко по хронологии):

LUNASAT передал исходные данные CDM и запросил подтверждение возможности маневрирования.

MIA-CORPORATION подтвердил получение и уточнил ответственность и сроки маневров.

LUNASAT предложил координацию с заморозкой несущественных маневров и инструкции по пересылке новых CDM.

Ссылки на CDM/обновления: Активно используется CDM 9850145153, который координирует LUNASAT; CDM 9850145154 выявлен как неиспользуемый и требует внимания.

Новые CDM: новых CDM, кроме упомянутых, не обнаружено.

Оперативный статус (наша сторона vs контрагент): LUNASAT ожидает подтверждения; MIA-CORPORATION подтвердил отсутствие необходимости в маневрах.

Уточнение обработки не переданного CDM 9850145154.

“Call’chart”” это алгоритм который создаёт графики эволюции и значений метаданных CDM(PC, miss distance, covariance)

“Call’major”” это инструмент отсева вообще не схожих кейсов/событий

“call ‘AI FEASIBILITY’” это вызов Health_status_AI чтобы получить данные и ограничения спутника текущего эталонного кейса, и трех самых схожих

примеры chart: График изменений PC текущего кейса и схожего

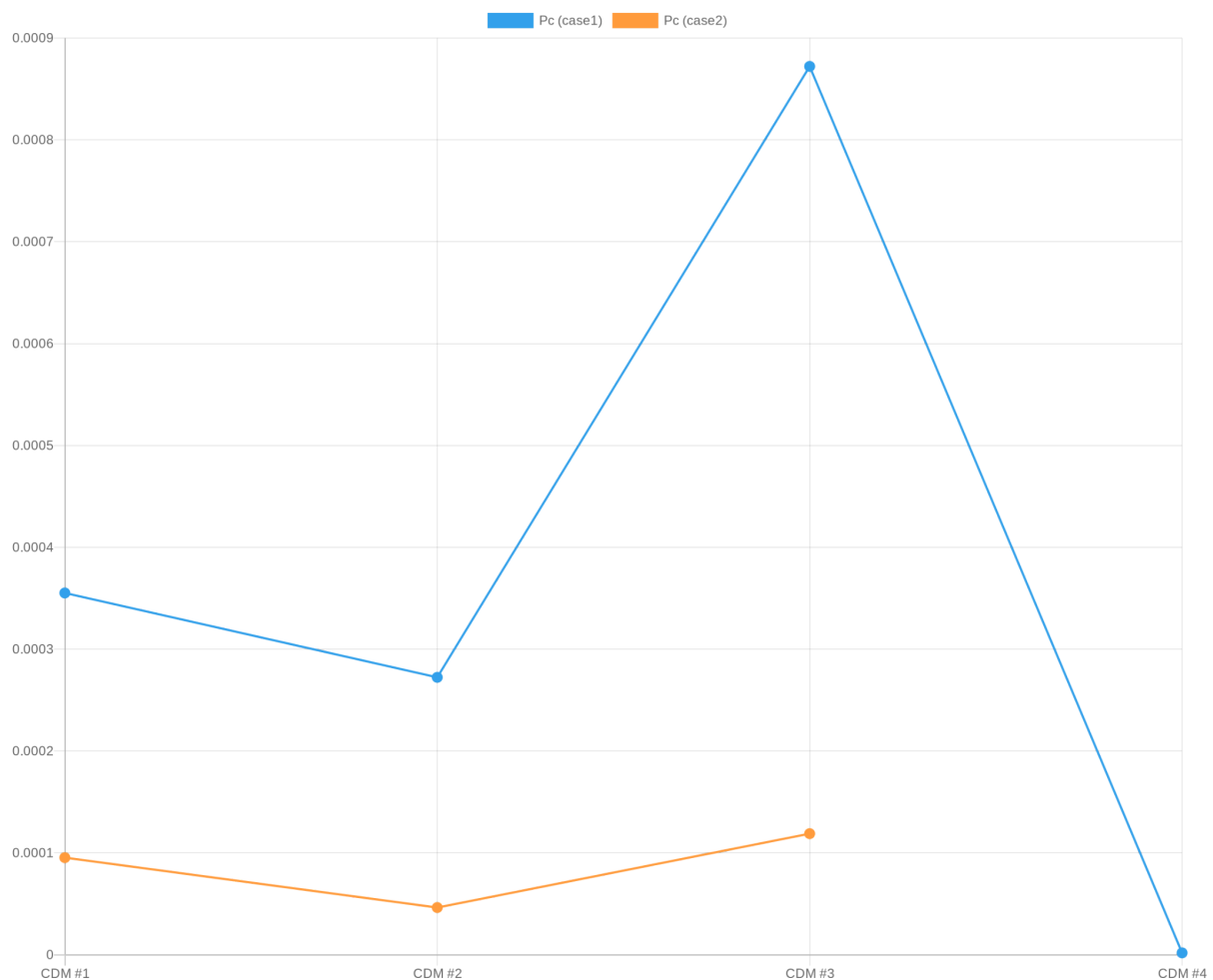


График изменений ковариации

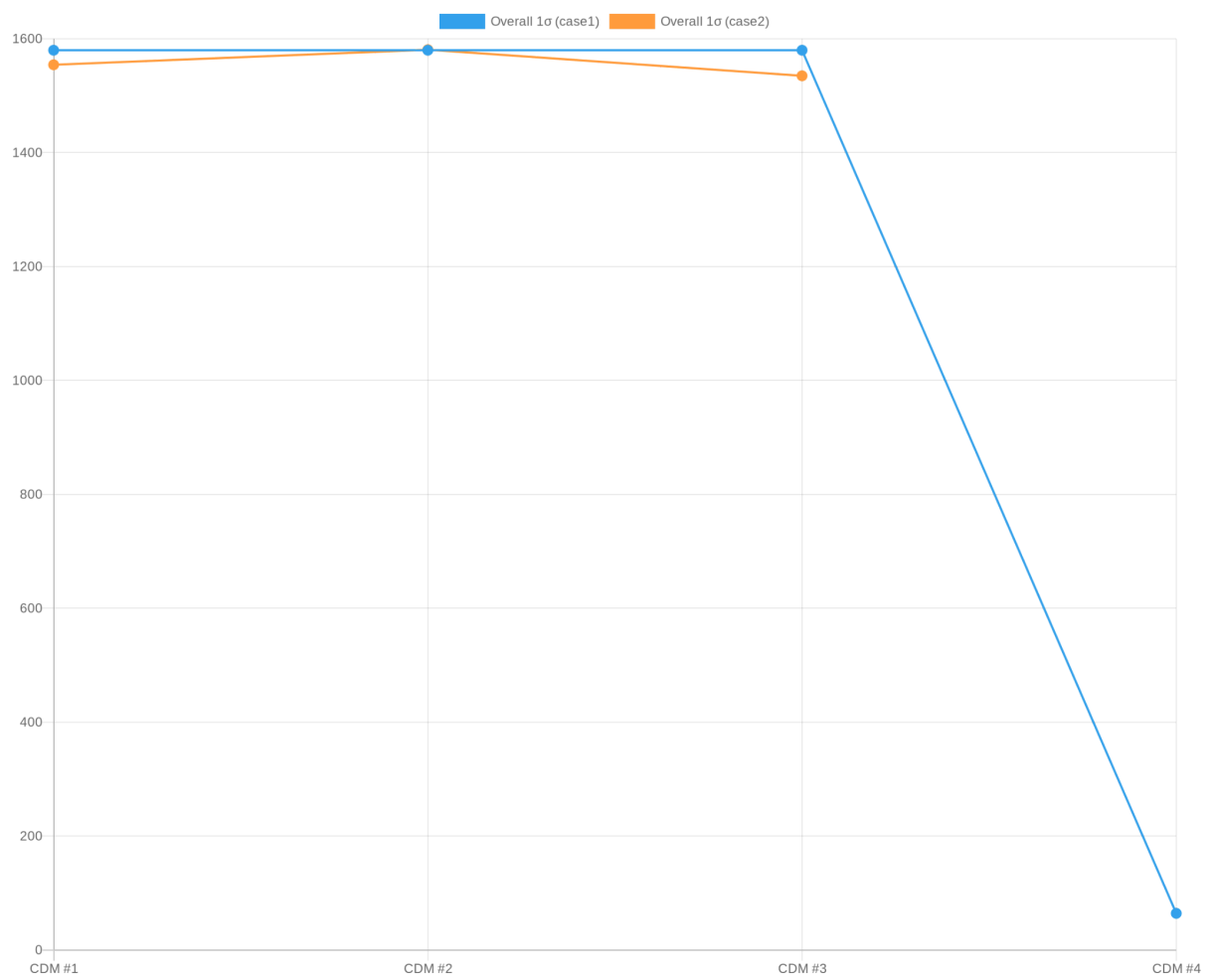
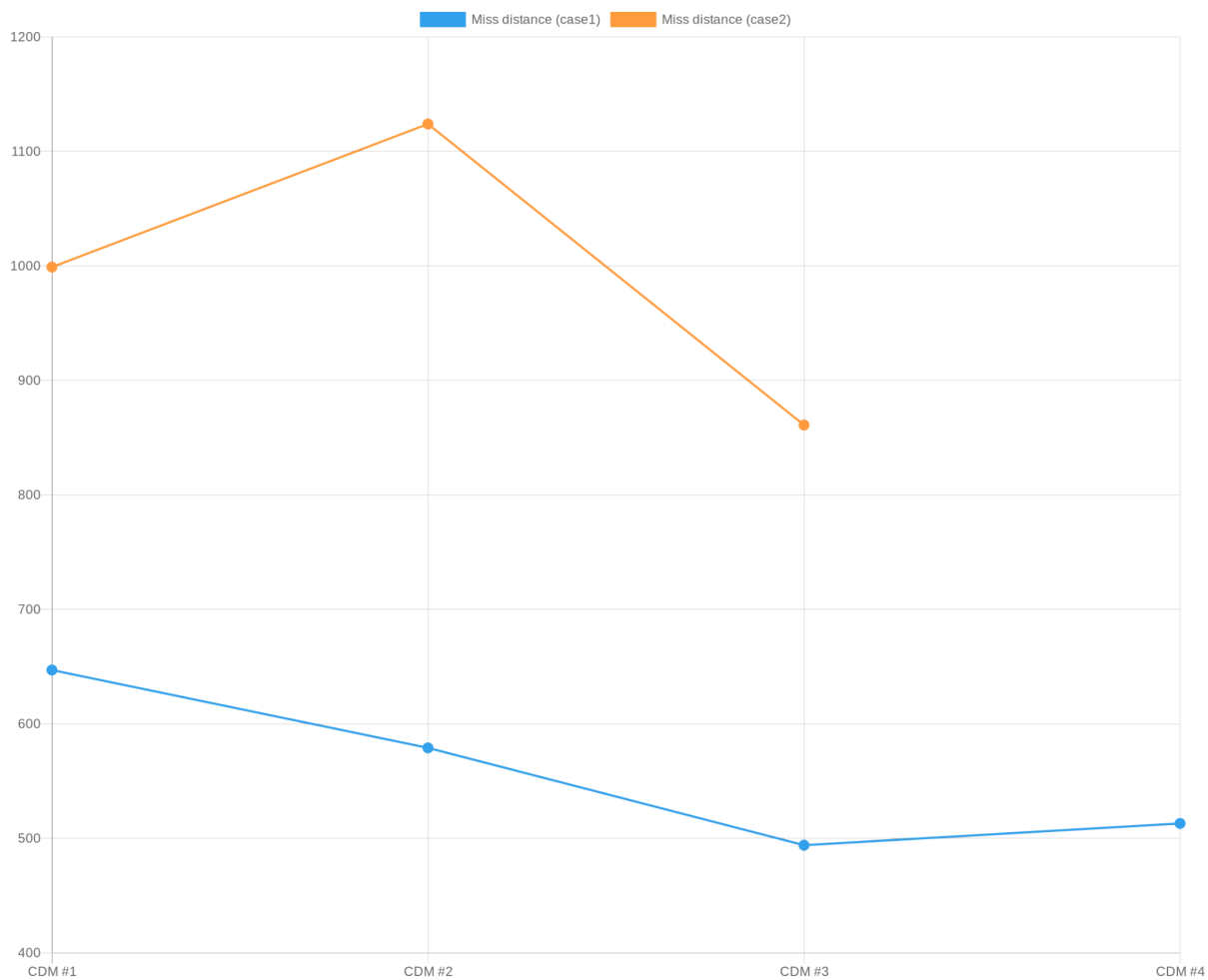


график изменений Miss distance



**При Выходе выдаст формат: машинный “report object”
(JSON-вывод)
выводит топ схожих кейсов событии, и аргументирует это
доказательствами**

```
Chat Session: 140f3...

Мне нужно узнать схожие события которые имеются в базе данных. Вот ключ самого события, pair_key = 540074022 , имена объектов pair_a = AKTAU-NEO-1, pair_b = CASPIAN-LX2

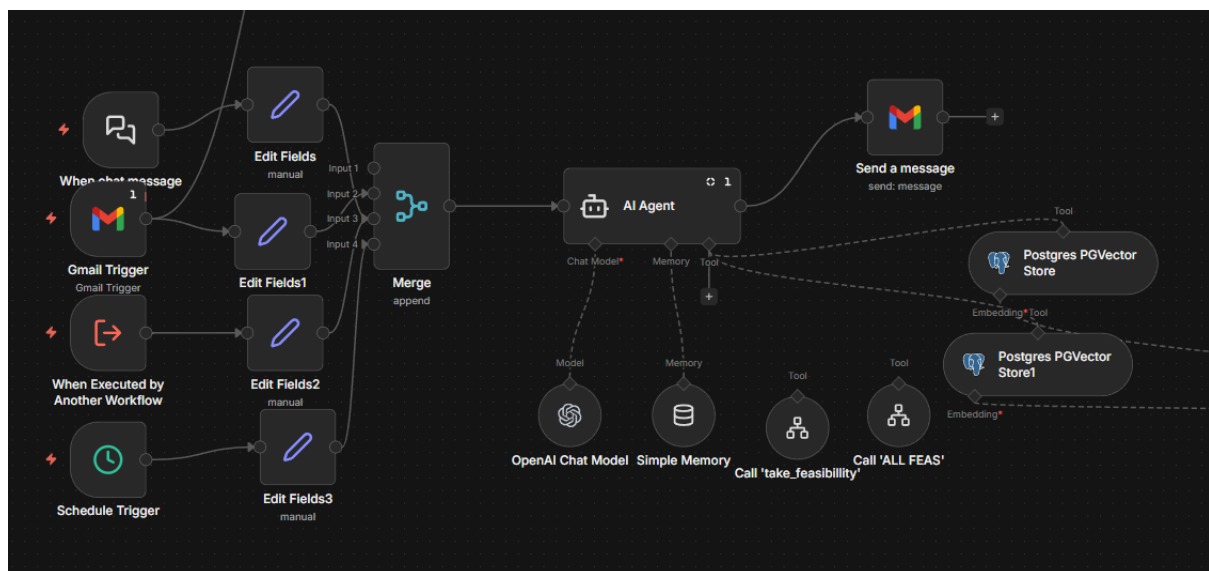
{ "summary": { "reference_pair_key": "540074022", "top_n": 5, "candidates_considered": 5, "tool_status": { "pg_vector": "ok", "casecardstore": "ok", "major": "ok", "precalc": "ok", "chart": "fail" }, "confidence": "Med" }, "top3_table": [ { "rank": 1, "case_id": "3418091776", "pair_key": "3418091776", "score": 0.95, "label": "VerySimilar", "major_pass": true, "missing_flags": [] }, { "rank": 2, "case_id": "2333789955", "pair_key": "2333789955", "score": 0.67, "label": "Weak", "major_pass": true, "missing_flags": [] }, { "rank": 3, "case_id": "1482759335", "pair_key": "1482759335", "score": 0.49, "label": "Weak", "major_pass": true, "missing_flags": [ "pc_unknown", "missing_dt" ] }, "evidence_packs": [ { "rank": 1, "why_similar": [ "Δ mass 17.3 kg (4.64%) close", "Overall similarity 95%", "pc similarity 96%", "miss distance similarity 97%", "relative speed similarity 90%", "covariance similarity 89%", "timing similarity 100%", "Reference pair AKTAU-NEO-1|CASPIAN-LX2 vs candidate ALMATY-OBS-7|TENGRI-NAV-2" ], "rules_fired": [ { "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" } ], "charts": [], "how_it_can_help": [ "Similar dynamic and covariance profile; applicable cross-checks for maneuverability and risk", "Strong match in approach timing and miss distance aspects" ], "mismatches_and_risks": [ "Mass difference 17.3 kg but within acceptable threshold", "Slight variations in covariance metrics", "Provider consistency maintained" ] }, { "rank": 2, "why_similar": [ "Δ mass 48.2 kg (12.94%) bigger difference", "Overall similarity lower at 67%", "pc similarity moderate 73%", "miss similarity 62%", "relative speed 57%", "covariance 79%", "timing similarity 96%", "Candidate lacks object pair names" ], "rules_fired": [ { "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" } ], "charts": [], "how_it_can_help": [ "Moderate similarity case; could help in understanding variability over broader object parameters" ], "mismatches_and_risks": [ "Missing object names in candidate", "Significant mass difference", "Lower pc and miss distance similarities" ] }, { "rank": 3, "why_similar": [ "Mass unknown (0 kg) - missing data", "Overall similarity 49%", "Low pc similarity 40%", "Miss similarity 63%", "No timing or covariance metrics available", "Candidate missing trace steps and some critical fields" ], "rules_fired": [ { "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" } ], "charts": [], "how_it_can_help": [ "Potentially relevant for incomplete data cases; caution advised due to missing metrics" ], "mismatches_and_risks": [ "Missing critical metrics like covariance data", "Incomplete candidate trace data", "Mass difference unknown" ] } ], "recommendation": { "best_candidate_pair_key": "3418091776", "what_to_reuse_from_casecard": [ "maneuverability assessments", "risk metrics related to covariance and miss distance", "timing profile analysis" ], "what_to_check_next": [ { "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "check": "Verify if covariance method differences occur in operational scenarios" }, { "rule_id": "RULE_05_NO_RAW_OUTPUT_FINAL", "check": "Confirm data completeness for object names and trace step details in candidates" } ], "workflow_inputs_for_chart": { "pair_a": "AKTAU-NEO-1", "pair_b": "CASPIAN-LX2", "pair_a(other)": "ALMATY-OBS-7", "pair_b(other)": "TENGRI-NAV-2", "pair_a(other1)": "KAZORBIT-OBS-12", "pair_b(other1)": "ORION-DEMO-7", "pair_a(other2)": "SOLARIS-NEO-1", "pair_b(other2)": "LYRA-T90" } }
```

Выход: { "summary": { "reference_pair_key": "540074022", "top_n": 5, "candidates_considered": 5, "tool_status": { "pg_vector": "ok", "casecardstore": "ok", "major": "ok", "precalc": "ok", "chart": "fail" }, "confidence": "Med" }, "top3_table": [{ "rank": 1, "case_id": "3418091776", "pair_key": "3418091776", "score": 0.95, "label": "VerySimilar", "major_pass": true, "missing_flags": [] }, { "rank": 2, "case_id": "2333789955", "pair_key": "2333789955", "score": 0.67, "label": "Weak", "major_pass": true, "missing_flags": [] }, { "rank": 3, "case_id": "1482759335", "pair_key": "1482759335", "score": 0.49, "label": "Weak", "major_pass": true, "missing_flags": ["pc_unknown", "missing_dt"] }, "evidence_packs": [{ "rank": 1, "why_similar": ["Δ mass 17.3 kg (4.64%) close", "Overall similarity 95%", "pc similarity 96%", "miss distance similarity 97%", "relative speed similarity 90%", "covariance similarity 89%", "timing similarity 100%", "Reference pair AKTAU-NEO-1|CASPIAN-LX2 vs candidate ALMATY-OBS-7|TENGRI-NAV-2"], "rules_fired": [{ "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" }], "charts": [], "how_it_can_help": ["Similar dynamic and covariance profile; applicable cross-checks for maneuverability and risk", "Strong match in approach timing and miss distance aspects"], "mismatches_and_risks": ["Mass difference 17.3 kg but within acceptable threshold", "Slight variations in covariance metrics", "Provider consistency maintained"] }, { "rank": 2, "why_similar": ["Δ mass 48.2 kg (12.94%) bigger difference", "Overall similarity lower at 67%", "pc

similarity moderate 73%", "miss similarity 62%", "relative speed 57%", "covariance 79%", "timing similarity 96%", "Candidate lacks object pair names"], "rules_fired": [{ "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" }], "charts": [], "how_it_can_help": ["Moderate similarity case; could help in understanding variability over broader object parameters"], "mismatches_and_risks": ["Missing object names in candidate", "Significant mass difference", "Lower pc and miss distance similarities"] }, { "rank": 3, "why_similar": ["Mass unknown (0 kg) - missing data", "Overall similarity 49%", "Low pc similarity 40%", "Miss similarity 63%", "No timing or covariance metrics available", "Candidate missing trace steps and some critical fields"], "rules_fired": [{ "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "name": "MAJOR gate + аудит причин + Proof#1" }], "charts": [], "how_it_can_help": ["Potentially relevant for incomplete data cases; caution advised due to missing metrics"], "mismatches_and_risks": ["Missing critical metrics like covariance data", "Incomplete candidate trace data", "Mass difference unknown"] }], "recommendation": { "best_candidate_pair_key": "3418091776", "what_to_reuse_from_casecard": ["maneuverability assessments", "risk metrics related to covariance and miss distance", "timing profile analysis"], "what_to_check_next": [{ "rule_id": "RULE_01_MAJOR_GATE_AUDIT", "check": "Verify if covariance method differences occur in operational scenarios" }, { "rule_id": "RULE_05_NO_RAW_OUTPUT_FINAL", "check": "Confirm data completeness for object names and trace step details in candidates" }] }, "workflow_inputs_for_chart": { "pair_a": "AKTAU-NEO-1", "pair_b": "CASPIAN-LX2", "pair_a(other)": "ALMATY-OBS-7", "pair_b(other)": "TENGGRI-NAV-2", "pair_a(other1)": "KAZORBIT-OBS-12", "pair_b(other1)": "ORION-DEMO-7", "pair_a(other2)": "SOLARIS-NEO-1", "pair_b(other2)": "LYRA-T90" } } }э

4)Module Health_status_AI

Триггер: When chat message received, Gmail trigger, Schedule trigger, when executed by another workflow. Обычно идет обновление и проверка состояния спутника по времени, если это срочно, то вызывается командой оператора, требуется лишь написать имя спутника, `sattelite_name`. в Gmail trigger присылается предупреждение о сбое, или значениях которые находятся в не нормы. “when executed by another workflow” это вызов от Case-finder. Нужно чтобы Case-finder смог найти какие ограничения были у определенного спутника в определенное время



tool: CALL “take_feasibility” База данных с последним обновлением состояния здоровья спутника. call “ALL FEAS” база с состоянием ВСЕХ спутников за ВСЕ время

При выходе выдает формат отчета, говорится о состоянии спутника за последние 3 часа, в зависимости от запроса может быть анализ состояния спутника за определенный промежуток времени. Выдается либо в чат либо на почту

Personal / AI FEASIBILITY + Add tag

When chat message 1

Gmail Trigger

Edit Fields manual

Execute workflow from On form submission

Chat

- satellite: MIA-X01
- окно анализа: текущее состояние + изменения за последние 3 часа
- источники: rules + casecards

2. КОРОТКИЙ ВЕРДИКТ

- Состояние: деградирует
- Главная причина(ы) вердикта:
Причина #1: Подсистема связи (COMMS) в состоянии MAJOR, состояние канала связи NO_LOCK, связь недоступна (comm_available=false) (E1, E2)
Причина #2: Подсистема пропульсии (PROP) в состоянии MAJOR, thruster_ready=false, valve_status=INHIBITED, запрещён burn (forbid_burn=true) (E3-E5)
Причина #3: Качество данных деградировано: высокая задержка данных (staleness=44c), пропуски телеметрии (4 gaps), потери пакетов 3.6% (E6-E9)

3. ТЕКУЩЕЕ СОСТОЯНИЕ СПУТНИКА

- Ключевые симптомы/флаги/режимы:
 - comms.link_state=NO_LOCK, comm_available=false (E1, E2)
 - prop.thruster_ready=false, valve_status=INHIBITED, forbid_burn=true (E3-E5)
 - data_quality validity=DEGRADED, staleness=44s, telemetry gaps=4, packet loss=3.6% (E6-E9)
 - CPU load OBC=34% (E10)
- Что хорошо:
 - ADCS, Power, Thermal подсистемы улучшились по состоянию (MAJOR→NOMINAL) (E14)
- Что плохо:
 - Ограничены командование и манёвренность спутника (см. Причина #1, #2)

4. ЧТО ИЗМЕНИЛОСЬ ЗА ПОСЛЕДНИЕ 3 ЧАСА

- Изменение #1: связь ухудшилась с OK→NO_LOCK, уровень серьезности с NOMINAL→MAJOR — потеря канала связи критична для управления (E11)
- Изменение #2: thruster_ready изменилось с true→false, valve status NOMINAL→INHIBITED — включена блокировка работы двигателей (E12)

Выход: “ИДЕНТИФИКАЦИЯ

time_id: 3

satellite: MIA-X01

окно анализа: текущее состояние + изменения за последние 3 часа

источники: rules + casecards

КОРОТКИЙ ВЕРДИКТ

Состояние: деградирует

Главная причина(ы) вердикта:

Причина #1: Подсистема связи (COMMS) в состоянии MAJOR, состояние канала связи NO_LOCK, связь недоступна (comm_available=false) (E1, E2)

Причина #2: Подсистема пропульсии (PROP) в состоянии MAJOR, thruster_ready=false, valve_status=INHIBITED, запрещён burn (forbid_burn=true) (E3-E5)

Причина #3: Качество данных деградировано: высокая задержка данных (staleness=44c), пропуски телеметрии (4 gaps), потери пакетов 3.6% (E6-E9)

ТЕКУЩЕЕ СОСТОЯНИЕ СПУТНИКА

Ключевые симптомы/флаги/режимы:

comms.link_state=NO_LOCK, comm_available=false (E1, E2)

prop.thruster_ready=false, valve_status=INHIBITED,

forbid_burn=true (E3–E5)

data_quality validity=DEGRADED, staleness=44s, telemetry

gaps=4, packet loss=3.6% (E6–E9)

CPU load OBC=34% (E10)

Что хорошо:

ADCS, Power, Thermal подсистемы улучшились по состоянию (MAJOR→NOMINAL) (E14)

Что плохо:

Ограничены командование и манёвренность спутника (см.

Причина #1, #2)

ЧТО ИЗМЕНИЛОСЬ ЗА ПОСЛЕДНИЕ 3 ЧАСА

Изменение #1: связь ухудшилась с OK→NO_LOCK, уровень серьезности с NOMINAL→MAJOR — потеря канала связи критична для управления (E11)

Изменение #2: thruster_ready изменилось с true→false, valve status NOMINAL→INHIBITED — включена блокировка работы двигателей (E12)

Изменение #3: качество данных ухудшилось с GOOD→DEGRADED: увеличилась задержка, количество потерь и пропусков в телеметрии (E13)

Изменение #4: ADCS, Power и Thermal подсистемы улучшились с MAJOR→NOMINAL (УЛУЧШЕНИЕ) (E14)

ПРОБЛЕМЫ И НА ЧТО СМОТРЕТЬ В ПЕРВУЮ ОЧЕРЕДЬ

P0: восстановить связь, подтвердить корректность телеметрии (см. Причина #1–#3)

P1: диагностика и устранение inhibit в пропульсии (см. Причина #2)

P2: мониторинг улучшения ADCS/Power/Thermal и оценка влияния на общую стабильность (E14)

ЧТО ЭТО ОЗНАЧАЕТ

Интерпретация 1: потеря связи и блокировка пропульсии ограничивают командные операции и манёвры, что

существенно снижает операционную готовность спутника (E1–E5)

Интерпретация 2: деградация качества телеметрии снижает точность контроля состояния спутника и усложняет диагностику (E6–E9)

ОГРАНИЧЕНИЯ, КОТОРЫЕ МОГУТ ПОЯВИТЬСЯ, И ПОЧЕМУ

Ограничение 1: burn запрещён из-за состояния пропульсии, ограничена возможность выполнения манёвров (см. Причина #2)

Ограничение 2: активные операции без восстановления связи не рекомендуется выполнять во избежание потерь управления (см. Причина #1)

ЧТО НЕ РЕКОМЕНДУЕТСЯ ДЕЛАТЬ СЕЙЧАС

Не рекомендуется 1: запускать операции, требующие команды через канал связи (см. Причина #1)

Не рекомендуется 2: пытаться сжечь топливо или включать двигатели пока сохраняется inhibit (см. Причина #2)

ОТКУДА ОБЫЧНО БЫВАЮТ ТАКИЕ ПРОБЛЕМЫ

Типовая причина: потеря связи вызвана отказом аппаратных средств или внешними помехами; пропульсия блокируется при неисправностях клапанов или бортовых защитах; ухудшение телеметрии связано с деградацией каналов передачи (E3–E5, E6–E9)

НАСКОЛЬКО ЭТО СЕРЬЁЗНО (Severity)

Severity: 80/100 — RED, Major критичные флаги по связи и пропульсии, запрет burn, деградация качества телеметрии (E1–E9)

ЧТО НУЖНО УТОЧНИТЬ

Не хватает данных: детали ошибки workflow для feasibility, логи с ошибками связи и пропульсии

Вопрос оператору #1: есть ли доступ к физической диагностике связи и пропульсии?

Вопрос оператору #2: были ли произведены операции командования за последние 3 часа?

EVIDENCE MAP:

E1=feas.subsystem_health.comms=MAJOR

E2=feas.key_metrics.comms.link_state=NO_LOCK

E3=feas.subsystem_health.prop=MAJOR
E4=feas.key_metrics.prop.thruster_ready=false
E5=feas.derived_flags.forbid_burn=true
E6=feas.data_quality.validity=DEGRADED
E7=feas.data_quality.staleness_s=44
E8=feas.data_quality.telemetry_gaps_last_10m=4
E9=feas.data_quality.packet_loss_estimate_pct_last_10m=3.6
E10=feas.key_metrics.obc.cpu_load_pct=34
E11=feas.update_info.comms_change='OK→NO_LOCK;
NOMINAL→MAJOR'
E12=feas.update_info.prop_change='thruster_ready true→false;
valve NOMINAL→INHIBITED'
E13=feas.update_info.data_quality_change='GOOD→DEGRADED;
staleness/loss/gaps↑'
E14=feas.update_info.recovery='adcs/power/thermal
MAJOR→NOMINAL''

ЗАКЛЮЧЕНИЕ ТЕХДОКУМЕНТАЦИИ:

Данный документ фиксирует текущее состояние разработки Satellite Safety Support System (S4S) как программной системы поддержки безопасности спутниковых операций в LEO. Внутри документа последовательно описаны: системный контекст и границы ответственности, используемые источники данных, модель данных и структура хранилищ (операционные данные и knowledge base), а также логика работы ключевых модулей, которые закрывают основные зоны операционного риска: корректность служебной коммуникации, консистентность CDM-информации, анализ готовности/ограничений спутника по телеметрии и формирование устойчивого handoff между сменами.

Архитектурно S4S реализован как оркестрируемый pipeline (n8n), где каждый модуль работает по одинаковому принципу: (1) получение триггера и входного контекста, (2) нормализация и приведение данных к единому формату, (3) retrieval актуальных правил/шаблонов и casecards из векторного

хранилища, (4) вычисление findings по формальным условиям и порогам, (5) сборка результата в стандартизированный отчёт с уровнем критичности и доказательной частью. Такой подход делает систему воспроизводимой, проверяемой и пригодной для операционного применения: каждый вывод опирается на конкретные поля входа/БД и конкретные правила, а не на “общие рассуждения”.

S4S сознательно позиционируется не как “автопилот” и не как замена специалиста. Система не выполняет орбитальную динамику и расчёт манёвра с нуля, не отправляет команды на аппарат, не принимает окончательные решения вместо оператора. Её задача — снижать вероятность ошибок и пропусков: быстро выявлять несостыковки и рискованные формулировки в переписке, поднимать релевантные исторические прецеденты по схожим CDM-паттернам, указывать ограничения по здоровью спутника, а также структурировать ход событий и незакрытые пункты для передачи смены. При недостатке входных данных, отсутствующих записях в БД или невозможности извлечь релевантные правила система обязана переходить в режим INSUFFICIENT_INPUT / INSUFFICIENT_EVIDENCE / DEGRADED_MODE, явно указывая, какие поля или источники необходимы для корректной проверки — это ключевое условие доверия к результатам.

Раздел демо демонстрирует работоспособность системы на типовых сценариях: запуск по триггеру, получение контекста из хранилищ, выполнение модулей, формирование результатов (verdict/findings/evidence/constraints), а также доставка ответа оператору и (при наличии) сохранение артефактов выполнения для аудита и повторной проверки. Исходные материалы и реализация (workflow-экспорты, структура данных, ruleset/casecards, примеры запусков) опубликованы в GitHub-репозитории, что обеспечивает прозрачность и воспроизводимость.

Дальнейшее развитие проекта связано не с “добавлением текста”, а с расширением покрытия и устойчивости: увеличение набора правил/порогов и калибровка скорингов, расширение совместимости с разными источниками CDM/форматами телеметрии, усиление контроля качества данных (staleness, конфликты источников), рост библиотеки casecards на реальных прецедентах и добавление регрессионных тестов, которые гарантируют стабильность выводов при обновлении ruleset. В таком виде S4S может масштабироваться от прототипа к более зрелой системе поддержки решений, оставаясь evidence-based и операционно проверяемой.