

Почему машины ошибаются

*Как инженеры учат автоматические системы
замечать сбои и действовать разумно*

А. В. Повный, ТГ: @AndyPovny

Сайт: <https://electricalschool.info/>

Канал в ТГ: <https://t.me/electricalschool>

Пролог. Коробка, которую робот не увидел

До конца смены оставалось меньше часа. Коробки шли по конвейеру с привычным промежутком. У направляющей каждая немного выравнивалась, после чего робот снимал её с ленты и переносил дальше. Оператор уже смотрел на следующую партию, когда заметил движение, которого не должно было быть.

Одна коробка прошла место захвата.

Робот остался над конвейером. Не дёрнулся, не попытался догнать изделие. Коробка скользнула вдоль направляющей, слегка развернулась и остановилась там, где могла помешать следующей. Оператор остановил участок с пульта и дождался, пока движение прекратится. Только тогда он подошёл ближе.

На экране появилась строка: «Нарушение цикла».

Он перечитал её. Нарушение было видно и без экрана. Сообщение не говорило, почему робот пропустил коробку и что успело произойти до остановки.

Начальник смены подошёл через несколько минут. Оператор показал на коробку, стоявшую под углом к направляющей.

- Датчик не увидел?
- Не знаю. Коробка здесь, робот её не взял.

Это были два достоверных наблюдения. Между ними пока не хватало нескольких событий.

Для оператора происшествие началось с пропущенной коробки. Он видел конвейер, робота и то, как быстро к месту остановки приближалось следующее изделие. Экран показывал состояние участка после нарушения цикла, но не воспроизводил всё, что система успела решить до него.

Инженер по надёжности открыл журнал событий. Там не было записи «робот не увидел коробку». Были изменения сигналов, команды и отметки времени. Одна строка сообщала о появлении изделия у входа в зону. Другая - о смене состояния датчика ближе к месту захвата. Затем шли запись о переходе к следующему шагу и сообщение, которое теперь висело на экране.

Инженер прокрутил журнал назад.

- Если датчик был неисправен, почему его сигнал менялся?
- Может, сработал поздно, - сказал оператор. - Я видел, что коробку повело у направляющей.

Оба объяснения подходили к тому, что они знали. Ни одно пока не объясняло всю последовательность.

Сама система не знала, что перед ней «коробка, которую пропустил робот». Датчик мог сообщить, что предмет перекрыл луч. Камера могла передать результат обработки изображения. Привод конвейера - сведения о движении ленты. Значение «коробка находится в месте захвата» возникало только тогда, когда программа сопоставляла эти сообщения по заранее заданным правилам.

В обычном цикле различие почти незаметно. Изделие приходит, сигналы совпадают с ожиданиями, робот выполняет захват. Человеку кажется, что машина видит коробку так же, как он. Но в журнале инженер искал не коробку. Он искал момент, когда поступивших сведений оказалось достаточно для следующего действия - или недостаточно, чтобы его начинать.

Начальник смены посмотрел на остановленный конвейер.

- Коробку уберём. После этого можно запускать?

Вопрос нельзя было отложить до спокойного разбора на следующий день. Участок простаивал, но ошибочный повторный пуск мог затронуть коробки, оставшиеся на линии, и заставить механизмы не в том состоянии, которое ожидала программа.

Инженер закрыл окно с сообщением об аварии, чтобы видеть записи перед ним.

- Пока неизвестно.

Он не знал, сместилась ли коробка до сигнала датчика или после. Не знал, что именно успела подтвердить камера. И не знал, завершился ли предыдущий шаг так, как предполагала программа. Удалить застрявшую коробку было необходимо, но этого ещё не хватало, чтобы считать участок готовым к работе.

За спинами людей молчала линия. Оператор видел коробку. Начальник смены видел остановленное производство. Инженер видел несколько строк, порядок которых ещё предстояло проверить. У машины тоже были свои сведения - только ни одно из них само по себе не означало: «коробка на месте, можно захватывать».

На следующий день им предстояло выяснить, почему эти разные картины разошлись.

Посмотрите как инженер. Какая запись или наблюдение помогли бы проверить версию о неисправном датчике, не принимая её заранее за ответ?

Глава 1. Машина работает не в том мире, который нарисован на схеме

Утром наладчик положил на стол распечатку схемы участка. На ней коробка двигалась по прямой линии. Напротив датчика её контур занимал одно определённое положение; в точке захвата робот встречал изделие в той же ориентации, в какой оно покинуло направляющую.

Разработчик провёл пальцем вдоль стрелок.

- Здесь коробка входит в зону. Датчик подтверждает её появление. Система определяет момент захвата, робот переносит коробку, камера

проверяет результат. Если подтверждения нет, цикл не должен продолжаться.

На схеме всё выглядело последовательным. Наладчик не стал спорить. Он попросил пройти к конвейеру.

Участок ещё не запустили. Коробку, из-за которой остановилась линия, убрали из прохода, но оставили рядом для осмотра. Один её угол был слегка примят. Наладчик положил изделие на неподвижную ленту и показал, как оно касается боковой направляющей.

- Нарисовано так, будто каждая идёт ровно. А эта упирается углом.

Он чуть сдвинул коробку. Та повернулась на несколько градусов.

- Обычно лента протаскивает её дальше, и она выравнивается. Иногда нет.

Разработчик посмотрел на расстояние между коробкой и датчиком. Даже в перекошенном положении изделие должно было оказаться в его зоне. По крайней мере, так следовало из схемы.

- Если она прошла мимо робота, датчик всё равно должен был её обнаружить.

- Должен, - согласился наладчик. - А вот когда именно и в каком положении она оказалась дальше, надо проверять.

Удобный мир чертежа

Программу управления невозможно написать для каждой царапины на направляющей и каждого неровного края коробки. Инженер выбирает признаки, которые важны для работы: где появляется изделие, с какой

скоростью движется лента, какой сигнал разрешает роботу начать действие. Из этих признаков складывается рабочее описание участка.

Такое описание необходимо. Без него нельзя задать последовательность операций и проверить, что должно происходить при штатной работе. Но схема всегда проще самого цеха.

На линии коробки немного различаются по размеру и жёсткости. После очистки направляющая может стоять на долю миллиметра иначе, чем до неё. Пыль и обрывки упаковочного материала меняют то, как изделие скользит вдоль борта. Эти различия не обязательно становятся неисправностями. Пока система справляется с ними, никто не замечает, сколько вариантов поведения скрывается за словом «коробка».

Наладчик провёл рукой вдоль направляющей, не касаясь механизмов.

- Этот узел снимали на прошлой неделе. После установки я проверял проход изделий. Задеваний не было.

- Положение проверяли измерением? - спросил инженер по надёжности.

- По контрольной коробке и по работе линии.

Ответ не означал, что регулировку выполнили неправильно. Он означал, что у них пока не было записи, с которой можно сравнить положение узла до обслуживания и после него. Практический опыт наладчика подсказывал, где искать возможный сдвиг, но не доказывал, что именно он вызвал вчерашнюю остановку.

Они осмотрели крепление. Болты стояли плотно. На одном краю направляющей виднелся след от картона, заметнее обычных потёртостей. Наладчик показал его остальным.

- Здесь коробка могла развернуться.
- Эта? - спросил оператор.
- Не знаю. След не подписан.

Инженер сфотографировал место. Новая подробность делала версию о перекосе разумной, но не превращала её в установленную причину.

Что должно было произойти

Разработчик попросил оператора описать обычный цикл без слов из программы.

- Коробка доходит до направляющей, прижимается к ней, потом оказывается под роботом. Робот берёт её. Если не взял, я вижу, что она ушла дальше.

- А если коробка идёт немного боком?
- Чаще всего выпрямляется.

Это «чаще всего» было важнее, чем казалось. Программа не могла опираться на него как на подтверждение положения каждой отдельной коробки. Для человека допустимое отклонение узнаётся по множеству мелочей: по движению изделия, звуку, зазору у направляющей. Система располагала лишь теми сведениями, которые ей передавали установленные устройства.

Разработчик открыл на ноутбуке схему предусмотренного цикла. В ней положение коробки к моменту захвата выводилось из обнаружения изделия и движения конвейера. Если начальное положение определено неверно или коробка повернулась позже, расчёт мог описывать уже не ту картину, которую видел оператор.

- То есть робот мог получить правильную команду для коробки, стоящей ровно? - спросил начальник смены.

- Мог, - ответил разработчик. - Но мы пока не знаем, какую команду он получил вчера. И получил ли команду на захват вообще.

Начальник смены посмотрел на оставленную рядом коробку.

- Тогда проверяем направляющую?

- Проверяем, - сказал инженер по надёжности. - Но отдельно от журнала событий. Если сейчас найдём отклонение, это не докажет, что вчера оно возникло до пропуска коробки.

Они договорились не регулировать узел до замеров. Иначе положение, в котором линия остановилась, исчезло бы раньше, чем его успели описать. Наладчик подготовил контрольные изделия; разработчик - записи сигналов за несколько циклов до остановки. Инженер хотел сопоставить их с тем, что происходило на самом участке.

У первой версии появилась проверяемая форма: коробка могла изменить положение возле направляющей, и предусмотренный цикл не учитывал этого изменения. Осталось выяснить, произошло ли оно вчера и могло ли объяснить поведение робота.

Когда разработчик снова открыл журнал, инженер указал на строку от датчика. Сигнал там был. Он не выглядел так, как ожидали увидеть при полностью отказавшем устройстве.

Посмотрите как инженер. Что подтвердило бы версию о смещении коробки у направляющей, а какое наблюдение заставило бы искать причину в другом месте?

Что осталось за кадром. Пока герои не измерили положение направляющей в момент происшествия и не восстановили движение той коробки. След на металле показывает, где изделие могло касаться узла, но не сообщает, когда он появился.

Глава 2. Что машина считает правдой

Разработчик вывел на экран два окна. В одном был журнал вчерашнего цикла, в другом - текущие сигналы датчиков. На остановленном конвейере оба окна казались одинаково понятными: коробки нет, луч датчика свободен, робот ожидает разрешения на работу.

Оператор принёс сохранённую коробку и положил её перед датчиком. В текущем окне один из сигналов изменился.

- Значит, он работает, - сказал оператор.
- Сейчас - да, - ответил разработчик. - Вчера это ещё надо проверить.

Наладчик попросил слегка повернуть коробку. Оператор сделал это, не запуская ленту. Сигнал оставался активным, пока боковая сторона закрывала луч, и исчез, когда край изделия вышел из зоны датчика.

Датчик не сообщал системе: «Коробка стоит ровно». Он сообщал только о состоянии луча. Ровная коробка, перекошенная коробка и рука наладчика могли вызвать одинаковое изменение сигнала. Понять, что именно находится перед датчиком, по одному такому сообщению было нельзя.

Один участок, разные сведения

Камера над линией давала другую информацию. Её изображение помогало определить видимое положение изделия в тот момент, когда был сделан снимок. Но оно не показывало, что происходило за пределами кадра и как коробка двигалась после съёмки.

Сведения о движении конвейера отвечали на третий вопрос: насколько продвинулась лента. Если система знала, где находилась коробка раньше, она могла рассчитать её ожидаемое положение позже. Такой расчёт полезен, пока коробка движется вместе с лентой так, как предполагает программа. Если изделие задело направляющую и повернулось, пройденный лентой путь уже не описывал его положение полностью.

Оператор посмотрел на три окна.

- Я вчера видел коробку у направляющей. Почему камера не показала её там?

Разработчик открыл сохранённый кадр. На нём коробка ещё приближалась к месту захвата. Её передний край был виден отчётливо, задний частично закрывал элемент ограждения.

- Камера показала её здесь, - сказал он. - Где она оказалась после этого снимка, по нему не видно.

Для оператора коробка существовала непрерывно: он видел, как она подошла, повернулась и прошла дальше. У системы непрерывной картины не было. Она получала отдельные сведения и связывала их расчётом. Если между двумя наблюдениями что-то менялось, программа могла узнать об этом лишь при наличии подходящего подтверждения.

Инженер по надёжности записал на листе три короткие фразы: «Луч перекрыт». «На снимке виден край коробки». «Лента прошла заданное расстояние». Рядом он оставил пустое место.

- Что мы вправе написать здесь? - спросил он.
- Что коробка подошла к захвату, - предложил оператор.
- Возможно. Но это уже вывод из трёх сообщений, а не содержание какого-то одного.

Он не оспаривал то, что оператор видел своими глазами. Для расследования было важно другое: какую картину могла составить сама система и на чём эта картина основывалась.

Неисправен или недостаточен?

Разработчик вернулся к вчерашнему журналу. На входе участка датчик присутствия сообщил о коробке. Позже датчик у направляющей тоже сменил состояние. Запись не походила на полный отказ, при котором устройство всё время молчало бы.

Начальник смены, слушавший разговор, указал на вторую строку.

- Если сигнал есть, почему мы вообще говорим о датчике?

- Потому что сигнал мог прийти не тогда, когда его ожидали, - ответил инженер. - Или программа могла придать ему не то значение. Но пока ни того ни другого мы не доказали.

Наладчик предложил проверить датчик на нескольких коробках, поставленных с разным перекосом. Разработчик согласился, но попросил сначала сохранить исходные настройки и положение узла. Проверка после регулировки показала бы, как датчик работает в новых условиях, а им требовалось разобраться со вчерашними.

На контрольной коробке сигнал появлялся устойчиво. При небольшом повороте он тоже возникал, хотя момент его появления менялся. Оператор повторил проход вручную ещё раз и посмотрел на экран.

- Получается, датчик видит коробку, но не говорит, как она стоит.
- Верно, - сказал разработчик. - А программа может ожидать, что после этого сигнала она движется определённым образом.

Он открыл участок кода, где из сигнала датчика и сведений о движении ленты формировалось условие для следующего шага. По отдельности эти строки ничего не объясняли. Надо было установить, какие значения программа получила в тот цикл и какое решение из них вывела.

Первое удобное объяснение пришлось отложить. Датчик нельзя было объявить полностью неисправным: в журнале сохранилось изменение его сигнала. Но и исправный сигнал ещё не означал, что система знала положение коробки достаточно точно для захвата.

Оператор снова взглянул на сохранённый кадр. На фотографии коробка была. В журнале был сигнал. На линии робот её пропустил. Теперь требовалось выяснить, на каком шаге сведения об изделии перестали соответствовать тому, что происходило в цехе.

Посмотрите как инженер. Какое наблюдение позволило бы отличить неверный сигнал датчика от верного сигнала, которому программа придала слишком широкое значение?

Для любознательных. Датчик присутствия обычно отвечает на узкий вопрос, например перекрыт ли луч в заданной точке. Чтобы определить положение изделия, одного такого ответа может не хватить: нужны другие измерения или допущение о том, как изделие двигалось после обнаружения. Чем важнее это допущение для следующего действия, тем полезнее проверять его отдельно.

Что осталось за кадром. Журнал показывает изменения сигналов, но пока не даёт полной картины движения коробки между ними. Контрольные проходы помогают понять возможности датчика; они не воспроизводят автоматически условия вчерашнего происшествия.

Глава 3. Три лица ошибки

На следующий день у инженера по надёжности лежали три распечатки одного и того же цикла. В каждой были те же сигналы и та же остановка. Различались пометки на полях.

На первой он написал: «Устройство». На второй: «Вывод». На третьей: «Решение».

- Три разных неисправности? - спросил начальник смены.
- Три места, где результат мог разойтись с действительностью. Не обязательно мы найдём неисправность в каждом.

Он разложил листы так, чтобы все видели одну строку: сигнал датчика у направляющей изменился. Вчера это лишило их простого ответа «датчик совсем не работал». Но не ответило на вопрос, почему робот пропустил коробку.

Когда подводит устройство

Наладчик начал с того, что можно было проверить руками и измерением. Он осмотрел крепление датчика, разъём, кабель и место, через которое проходил луч. Если корпус сдвинулся, соединение нарушилось или поверхность загрязнилась, сигнал мог перестать соответствовать обычному положению изделия.

Проверка не обнаружила явного обрыва. Датчик переключался, когда контрольная коробка пересекала луч. Это был полезный результат, но наладчик не стал ставить на первой распечатке крест.

- Сейчас работает. Вчерашнее состояние мы этим не восстановили.

Физический отказ не всегда выглядит как устройство, которое навсегда перестало отвечать. Контакт может нарушаться при движении кабеля, загрязнение может мешать лишь при определённом положении изделия. А иногда устройство исправно, но механизм доставляет изделие к нему не так, как было предусмотрено.

На распечатке «Устройство» инженер отметил то, что они знали: изменение сигнала было записано; при нынешней проверке датчик переключался. Рядом осталось то, чего они не знали: в какой точке находился край вчерашней коробки в момент переключения и сохранялось ли положение датчика на протяжении всей смены.

Когда сигнал понят неверно

Разработчик взял вторую распечатку.

- Допустим, датчик передал именно то, что должен: луч перекрыт. Что программа решила после этого?

В предусмотренном цикле появление сигнала означало, что изделие достигло определённого участка пути. Дальше его положение оценивалось с учётом движения конвейера. Такой расчёт работал для коробки, которая после обнаружения двигалась предсказуемо.

Инженер передвинул контрольную коробку вдоль направляющей, затем слегка развернул её.

- Луч перекрыт в обоих случаях. А место, за которое роботу надо брать коробку, уже другое.

Ошибка здесь не в самом сигнале. Датчик мог верно сообщить о перекрытии луча, а программа - сделать из этого сообщения вывод, для которого не хватало оснований. Чтобы доказать такую версию, нужно было сравнить вычисленное положение изделия с его действительным положением в тот же момент. Последнего наблюдения у героев пока не было.

Оператор посмотрел на сохранённый кадр камеры.

- Значит, по фотографии тоже нельзя сказать, что программа ошиблась?

- Можно сказать, где коробка была на снимке, - ответил инженер. - Нельзя без проверки приписать ей то же положение позже.

Вторая версия оставалась открытой.

Когда вывод верен, а действие нет

На третьей распечатке разработчик обвёл условие перехода к следующему шагу.

- Представим, что система правильно определила: положение коробки не подтверждено. Что она должна сделать?

Оператор ответил первым:

- Не пытаться брать её наугад.

- Согласен. Но этого недостаточно, - сказал начальник смены. - Если каждый раз останавливать весь участок при коротком пропадании сигнала, мы можем получить остановки без всякой необходимости.

У обоих была причина говорить именно так. Захват без подтверждения мог повредить изделие или оборудование. Бессмысленная остановка тоже требовала вмешательства людей и разбирательства. Выбор действия зависел от того, какие сведения отсутствовали, что ещё продолжало двигаться и какие меры были предусмотрены для такого случая.

Можно правильно распознать ситуацию и всё же выбрать для неё неподходящее действие. Например, зафиксировать, что положение коробки

не подтверждено, но разрешить шаг, рассчитанный только на подтверждённое положение. Это была третья версия. Вчерашний журнал ещё не позволял установить, случилось ли именно так.

Начальник смены указал на экран:

- А сообщение «Нарушение цикла»?
- Оно относится к тому, что система обнаружила позже, - сказал инженер. - Нам надо выяснить, что она решила раньше.

Карта, по которой пойдёт расследование

Инженер взял чистый лист и нарисовал пять клеток.

Сигнал → интерпретация → решение → действие → проверка результата.

Первая клетка отвечала на вопрос, что сообщили датчик, камера и устройства конвейера. Вторая - какой смысл программа придала этим сведениям. Третья - какой шаг она выбрала. Четвёртая - что сделали робот и конвейер. Пятая - было ли подтверждено, что получилось именно то, чего ожидали.

- Мы всё время перескакивали от первой клетки к четвёртой, - заметил разработчик. - Увидели пропущенную коробку и сразу спросили, сломался ли датчик.

Теперь у каждой версии появилось своё место. Отказ устройства следовало искать прежде всего там, где возникает сигнал. Неверный вывод - там, где сигналы превращаются в представление о положении коробки. Неудачное правило действия - между представлением и командой. Даже

выполненная команда не завершала расследование: робот мог двигаться правильно, но не получить ожидаемого результата, и это требовалось заметить.

На схеме у инженера не получилось провести жирную черту вокруг одной клетки. Данных для этого не хватало. Зато стало ясно, какие вопросы задать журналу: что было известно системе перед пропуском коробки, какой переход она разрешила и чем проверяла его итог.

Разработчик увеличил фрагмент записи с двумя соседними событиями. Сигнал датчика и переход к следующему шагу стояли почти рядом. По первому взгляду нельзя было уверенно сказать, какое событие система обработала раньше.

- Тут, похоже, дело во времени, - сказал он.

Инженер не стал менять пометок на распечатках. Пока они выяснили лишь то, что одна видимая ошибка могла возникнуть в разных местах. Следующей уликой был порядок событий.

Посмотрите как инженер. Если датчик действительно сообщил о коробке, какая запись помогла бы различить две оставшиеся версии: неверный вывод о её положении и неверное действие при правильно установленной неопределённости?

Что осталось за кадром. Карта помогает не смешивать причины, но сама не устанавливает их. Одно происшествие может затронуть несколько её звеньев; для каждого вывода нужны отдельные наблюдения или испытания.

Глава 4. Время, которого не видно на фотографии

Разработчик увеличил журнал так, что на экране остались две соседние строки. В первой датчик у направляющей изменил состояние. Во второй программа перешла к следующему шагу цикла.

- Сигнал был раньше, - сказал оператор.

Инженер по надёжности придвинулся к экрану.

- Раньше по отметкам в этом журнале. Нам ещё нужно понять, когда программа получила его для обработки.

На сохранённой фотографии коробка приближалась к месту захвата. Фотография показывала положение изделия в один момент. По ней нельзя было определить, успела ли программа получить подтверждение до того, как приняла решение. Разница между этими событиями могла быть короткой, но именно на ней теперь держалась версия происшествия.

Два события и одно решение

Разработчик открыл подробную запись работы контроллера. Он пояснил, что сигнал от устройства и решение программы не всегда возникают в одну и ту же секунду, даже если на экране журнала стоят рядом. Сначала меняется состояние входа. Затем система считывает его, выполняет предусмотренные проверки и обновляет состояние цикла.

Обычно этот промежуток не мешает работе: изделие движется в пределах условий, для которых настроен участок. Но если подтверждение приходит у самой границы допустимого времени, порядок становится важнее самого факта появления сигнала.

- Представим, что программа ждёт подтверждение до перехода, - сказал разработчик. - Оно появляется сразу после того, как решение уже принято. В журнале сигнал есть. Для того решения его ещё не было.

- Значит, робот мог пропустить коробку, хотя датчик сработал? - спросил оператор.

- Мог. Но мы пока не знаем, так ли произошло вчера.

Наладчик напомнил о коробке, которую разворачивало у направляющей. Если её край достиг луча позже обычного, сигнал тоже мог задержаться. Получалась цепочка, которую стоило проверить: изменение положения изделия сдвинуло момент обнаружения, а программа продолжила цикл, не дождавшись нужного подтверждения.

Это было объяснение, а не установленная причина. След от картона на направляющей не сообщал, какая коробка его оставила. Журнал не показывал положение вчерашней коробки в каждый момент её движения.

Что отсчитывает таймаут

В программе нашлось условие ожидания. После одного события система давала следующему подтверждению ограниченное время. Если подтверждения не было, выполнялся другой переход.

Такой предел называют таймаутом. Он нужен, чтобы машина не ждала бесконечно, когда коробка не пришла или устройство перестало отвечать. Но таймаут имеет смысл только вместе с ответами на два вопроса: от какого события начат отсчёт и что система делает, когда время истекло.

Разработчик показал условие инженеру. Отсчёт начинался после появления коробки на входе участка. Предполагалось, что за отведённое время она пройдёт путь до датчика у направляющей.

- Если коробку притормозило у борта, время могло истечь раньше подтверждения, - сказал наладчик.

- А если лента двигалась иначе, чем мы предполагаем, тоже, - добавил инженер.

Он попросил не менять настройку таймаута до завершения проверки. Простое увеличение времени могло помочь при запаздывающем сигнале, но могло и заставить систему дольше ждать там, где изделия вообще нет. Важнее было понять, почему подтверждение и переход оказались так близко друг к другу и какую реакцию предусматривала программа.

Почему часы могут спорить

Камера хранила время снимка в собственном архиве. Контроллер записывал события в своём журнале. Оператор добавил в сменную запись время, которое видел на экране пульта. Все три отметки относились к одному происшествию, но инженеру нельзя было просто расположить их по числам и объявить получившийся порядок точным.

Часы разных устройств могут быть настроены не вполне одинаково. Кроме того, отметка может относиться к разным этапам: получению сигнала, обработке сообщения или сохранению записи. Разработчик начал с тех событий, порядок которых можно было установить внутри контроллера, а снимок камеры пока оставил как отдельное свидетельство.

Начальник смены посмотрел на три окна.

- Выходит, даже время в журнале не обязательно говорит правду?

- Говорит о том, что было записано и когда именно эта система поставила отметку, - ответил инженер. - Для сравнения с другим устройством надо знать, как устроена запись.

Ему не требовалось доказать, что все часы показывали неверное время. Достаточно было не выдавать их показания за более точную картину, чем они позволяли получить.

К концу разбора разработчик нашёл важную подробность. В проблемном цикле подтверждение от датчика у направляющей было зарегистрировано очень близко к моменту, когда ожидание завершилось. В обычных сохранённых циклах между этими событиями оставался заметный запас.

Это не объясняло, почему сигнал запоздал. Причиной могло быть движение коробки, работа устройства или сочетание условий, которых пока не воспроизвели. Но направление расследования изменилось. Герои искали уже не только неисправную деталь: им предстояло понять, как небольшое отклонение во времени повлияло на следующий шаг системы.

Инженер закрыл фотографию и оставил на экране последовательность событий. На снимке коробка просто стояла боком. В журнале она успела стать опоздавшим подтверждением.

Посмотрите как инженер. Какую проверку нужно провести, прежде чем сравнивать время на снимке камеры со временем перехода, записанным контроллером?

Для любознательных. Таймаут не определяет причину задержки. Он лишь устанавливает предел ожидания после выбранного события. Один и тот же истёкший таймаут возможен, если изделие движется медленнее обычного, датчик не передал сигнал или программа не получила его вовремя.

Что осталось за кадром. Записи позволяют сузить промежуток, в котором возникло расхождение, но не дают непрерывного изображения вчерашнего движения коробки. Пока нельзя утверждать, что её перекося вызвал задержку сигнала, а не совпал с ней.

Глава 5. Как маленькое отклонение становится остановкой

Наладчик поставил контрольную коробку на ленту с небольшим перекосом. Участок работал в проверочном режиме: движения выполнялись под наблюдением, без попытки воспроизвести вчерашнее происшествие на полной скорости. Оператор следил за коробкой, разработчик - за сигналами.

Коробка коснулась направляющей углом. На мгновение её передний край пошёл вдоль борта, а задний ещё двигался с лентой. Затем изделие выровнялось и прошло дальше.

- Вчера она могла не выровняться, - сказал оператор.
- Могла, - ответил наладчик. - Но контрольная прошла. Это пока показывает только то, что перекося меняет её движение.

На экране появление сигнала у направляющей при таком проходе отличалось от обычного. Не настолько, чтобы само по себе вызвать остановку. Однако теперь было легче представить, почему сигнал в проблемном цикле оказался почти у границы времени ожидания.

Цепочка вместо одной причины

Герои вновь прошли по записям, не пропуская шагов. Коробка вошла в участок. Подтверждение у направляющей появилось позже, чем в нескольких предшествующих циклах. Ожидание следующего шага к тому времени завершалось. Система перешла дальше, а поздний сигнал не вернул её к прежнему состоянию.

После этого камера не дала однозначного подтверждения ожидаемого положения изделия. Робот не выполнил захват. Коробка ушла за место, где её должны были снять с ленты, и оператор остановил участок.

Теперь у происшествия появилась последовательность. Но последовательность ещё не равнялась полному объяснению. Записи подтверждали задержку и переход программы; они не показывали с достаточной точностью, что именно задержало коробку. Пробный проход с перекосом делал версию о направляющей правдоподобной, но не доказывал, что вчера всё происходило так же.

Начальник смены указал на схему.

- Если программа не получила подтверждения вовремя, почему она вообще перешла дальше?

Разработчик вернулся к условию, которое они нашли накануне. Оно не разрешало роботу захват наугад. После истечения времени ожидания

система прекращала попытку обработать эту коробку и готовилась к следующему шагу конвейера. Предполагалось, что неподтверждённое изделие не окажется в месте, где помешает дальнейшей работе.

- То есть робот сделал то, что ему предписали? - спросил начальник смены.

- Похоже на то, - ответил разработчик. - Вопрос в том, годится ли это правило для ситуации, которая возникла здесь.

Это различие изменило разговор. Они уже не искали команду, из-за которой робот якобы «не захотел» брать коробку. Они проверяли допущение: можно ли после запоздавшего подтверждения считать путь свободным для продолжения цикла.

Где следовало заметить расхождение

Инженер по надёжности положил рядом две распечатки. На одной было предусмотренное движение коробки, на другой - восстановленная последовательность сигналов.

В предусмотренном цикле подтверждение положения приходило с запасом времени. Затем система принимала решение о захвате и проверяла результат. Во вчерашнем цикле запас исчез. Подтверждение пришло на границе перехода, а позднее камера не смогла подтвердить ожидаемое положение.

- Здесь у нас ещё нет доказательства, что коробка стоит правильно, - сказал инженер, показывая на момент перехода. - А здесь уже есть признак, что ожидаемого результата не получилось.

Между этими точками система продолжала работу так, будто неподтверждённая коробка больше не требует внимания. Оператор заметил её только после того, как она прошла мимо робота.

Наладчик снова посмотрел на направляющую. Даже если вчерашний перекос начался именно там, он был небольшим. На линии случались коробки, которые подходили к борту неидеально и всё равно успешно обрабатывались. Сам по себе такой случай не должен был приводить к остановке участка.

Начальник смены предложил два быстрых исправления: точнее выставить направляющую и увеличить время ожидания. Наладчик был готов проверить положение узла. Разработчик согласился оценить настройку. Инженер попросил не считать ни одно из этих действий окончательным ответом.

- Если мы выровняем направляющую, другая коробка всё равно может прийти позже обычного. Если увеличим время, система будет дольше ждать, но по-прежнему должна решить, что делать, когда подтверждения нет. Нам нужно проверить это решение.

Начальник смены не возражал против проверки. Ему нужен был понятный ответ, когда участок можно будет вернуть в работу, а не ещё одна остановка через час после пуска.

Перед уходом инженер дополнил карту расследования. Возле «сигнала» он отметил запаздывающее подтверждение. Возле «интерпретации» - неизвестное положение коробки после перехода. Возле «действия» - продолжение цикла без захвата. Последнюю клетку, «проверка

результата», он обвёл: система обнаружила нарушение, но лишь после того, как коробка оказалась за местом обработки.

На доске остался вопрос для следующего этапа работы: какое расхождение можно было заметить раньше и что оно на самом деле позволило бы заключить?

Посмотрите как инженер. Если подтверждение положения не пришло в ожидаемое время, что уже известно системе, а какую причину задержки она ещё не вправе считать установленной?

Что осталось за кадром. Герои восстановили порядок записанных событий и нашли слабое место в реакции системы. Причина запаздывания пока не доказана: контрольный проход показывает возможный механизм, но не заменяет наблюдения за вчерашней коробкой.

Интерлюдия. Лифт

На совещании разработчик снова показал журнал упаковочной линии. Подтверждение от датчика было записано, но пришло слишком близко к переходу программы. Начальник смены всё ещё смотрел на эту строку с недоверием.

- Если сигнал в журнале есть, почему его нельзя считать достаточным?

Инженер по надёжности взял лист бумаги и нарисовал кабину лифта с дверью.

- Представьте два сообщения: «дверь закрыта» и «кабина получила разрешение на движение». В журнале оба есть. Достаточно ли этого, чтобы понять, всё ли произошло правильно?

- Нужно знать порядок, - ответил оператор. - Сначала подтверждение, потом разрешение.

- И ещё нужно знать, к какой двери и к какому состоянию относится подтверждение, - добавил инженер.

На упаковочной линии вопрос был похожим, хотя задача и оборудование были другими. Запись «коробка обнаружена» имела значение для решения о захвате только в том случае, если система получила её вовремя и могла связать с нужным изделием. Сигнал, пришедший после перехода к следующему шагу, не превращал уже принятое решение в обоснованное.

Начальник смены вернулся к экрану.

- Значит, нам мало проверить, был ли сигнал. Надо проверить, что программа знала в тот момент, когда решила продолжать.

Инженер кивнул. Он не предлагал управлять конвейером по правилам лифта. У разных машин различаются устройство, опасности и необходимые меры защиты. Пример помогал задать более точный вопрос к вчерашнему журналу: **какое подтверждение было доступно системе до решения, а не появилось в записи позже?**

Разработчик вывел рядом время ожидания и момент перехода. Теперь им предстояло определить, как система должна реагировать, если ожидаемое подтверждение так и не пришло.

Что осталось за кадром. Схема лифта здесь служит только пояснением порядка событий. По этому примеру нельзя выбирать алгоритм защиты ни для лифта, ни для упаковочной линии.

Часть II. Как заметить расхождение

Глава 6. Ожидание против наблюдения

После совещания инженер по надёжности попросил выгрузить записи не одного происшествия, а нескольких предшествующих смен. Он искал циклы, в которых подтверждение у направляющей приходило позже обычного.

Разработчик сперва открыл только аварии.

- Здесь больше ничего похожего нет.
- Посмотрим и успешные циклы, - сказал инженер. - Если коробку успели захватить, задержка всё равно могла быть.

Такие циклы нашлись. В двух из них сигнал у направляющей появился ближе к концу отведённого времени, чем в большинстве остальных. Робот обработал коробки, камера подтвердила результат, и никто на участке не заметил отклонения. Система тоже не сообщила о нём: предусмотренный шаг завершился успешно.

Наладчик изучил даты.

- В эти дни я не записывал неисправностей.
- И правильно, если их не было, - ответил инженер. - Но теперь видно, что вчерашний цикл не возник из совершенно неизменной картины.

Записи не доказывали постепенный износ и не указывали на направляющую. Задержки могли иметь разные причины. Они показывали

другое: система уже получала подтверждения с меньшим запасом времени, хотя результат оставался нормальным.

Что именно ожидала система

Разработчик написал на доске условие обычного цикла словами: после появления коробки на входе подтверждение у направляющей должно поступить до окончания времени ожидания. Это ожидание не означало, что каждая коробка обязана прийти в одну и ту же долю секунды. Оно задавало предел, в котором программа рассчитывала продолжить работу по обычному сценарию.

Рядом инженер записал вчерашнее наблюдение: подтверждение не было доступно системе к моменту перехода.

Разница между этими фразами казалась небольшой. Для расследования она была решающей. «Подтверждение не пришло вовремя» описывало обнаруженное расхождение. «Сломался датчик» называло его возможную причину. Чтобы перейти от первого утверждения ко второму, требовались дополнительные данные.

- А если коробки вообще не было? - спросил начальник смены.
- Тогда мы тоже могли бы не получить подтверждение, - сказал разработчик.
- А если она застряла перед датчиком?
- Результат ожидания был бы тем же.

Один пропущенный сигнал не позволял выбрать между отсутствующей коробкой, изменившимся движением изделия, неисправностью устройства и ошибкой в обработке сигнала. Зато он позволял сказать: обычная картина движения больше не подтверждена.

Проверка на линии

Наладчик предложил провести ещё один контролируемый проход. На этот раз целью было не повторить вчерашнюю остановку, а посмотреть, какое расхождение система способна показать до неё. После согласования порядка проверки они подавали контрольные коробки и наблюдали за временем между входным сигналом и подтверждением у направляющей.

Несколько изделий прошли с обычным запасом. Одно слегка коснулось борта и задержалось заметнее других, но уложилось в отведённое время. На экране появился только итог успешного цикла.

- Для программы они одинаковые, - заметил оператор.

Разработчик открыл подробные записи.

- Результат одинаковый. Время между сигналами разное.

Оператору не требовалось видеть длительность каждого прохода во время нормальной работы. Но для поиска повторяющегося отклонения эти значения были полезнее, чем один признак «успешно» или «авария». Они показывали, насколько близко цикл подошёл к границе, после которой система меняла своё поведение.

Инженер предложил сохранять этот промежуток для анализа и отдельно определить, когда его изменение требует внимания. Он не

предложил немедленно останавливать линию при каждой задержке. Сначала нужно было выяснить, какие колебания обычны для участка и что происходит с коробкой, если подтверждение запаздывает.

Начальник смены поддержал первую часть предложения, но насторожился из-за второй.

- Если вы сделаете из этого новую тревогу, оператор будет подходить к линии каждые полчаса. А вчерашнюю коробку мы всё равно можем пропустить.

- Поэтому пока разделим наблюдение и действие, - ответил инженер. - Мы можем заметить, что время меняется, не объявляя каждое изменение поломкой.

Расхождение ещё не диагноз

К вечеру у героев появился способ точнее описывать проблему. Вместо «датчик иногда не работает» они могли сказать: «Подтверждение положения приходит с разным запасом времени; в проблемном цикле система перешла дальше, не получив его вовремя».

Такую формулировку можно было проверять. Разработчик мог проследить правило перехода. Наладчик - сопоставить задержанные проходы с поведением изделий у направляющей. Оператор - наблюдать, когда коробки начинают идти иначе, чем обычно. Ни одному из них не приходилось заранее соглашаться с чужой версией причины.

Однако впереди возникло новое решение. Если система научится замечать задержку раньше, о каждом ли таком случае нужно сообщать оператору? И когда одного сообщения мало, а работу следует прервать?

Начальник смены попросил прежде всего поднять журнал остановок. Он хотел знать не только цену пропущенной коробки, но и то, как часто участок уже прекращал работу из-за тревог, после которых неисправность не находили.

Посмотрите как инженер. Какие разные причины могут привести к одному наблюдению - подтверждение у направляющей не поступило вовремя? Какое дополнительное сведение помогло бы различить хотя бы две из них?

Для любознательных. Время между двумя ожидаемыми событиями можно отслеживать даже тогда, когда цикл закончился успешно. Но само по себе увеличение этого времени не доказывает износ или приближение отказа: сначала нужно понять обычный разброс значений и условия, при которых они получены.

Что осталось за кадром. Найденные в архиве задержки похожи по записи, но не обязательно одинаковы по причине. Пока герои не сопоставили их с конкретными изделиями и состоянием участка, называть их ранними проявлениями вчерашнего сбоя нельзя.

Глава 7. Цена тревоги и цена молчания

Начальник смены принёс журнал остановок. На нескольких страницах повторялась похожая запись: участок остановлен, коробка осмотрена, неисправность не подтверждена, работа возобновлена.

- Вот почему я не хочу ещё одну тревогу на каждую задержку, - сказал он. - Оператор идёт к линии, проверяет изделие, зовёт наладчика. Потом оказывается, что всё прошло нормально.

Разработчик открыл график времени между появлением коробки на входе и подтверждением у направляющей.

- Если ждать до самого конца допустимого времени, мы можем снова узнать о проблеме слишком поздно. Можно поставить предупреждение раньше.

На графике это выглядело просто: провести черту и отмечать все циклы выше неё. Но черта не знала, сколько раз оператору придётся реагировать на исправную коробку. И не знала, какую именно коробку линия пропустит, если провести её выше.

Два способа ошибиться

Инженер по надёжности предложил разобрать не настройки, а последствия решений.

Представим, что система сообщает о возможном отклонении, хотя коробка благополучно доходит до места захвата. Это ложная тревога. Она отнимает время и внимание. Если таких сообщений много, люди постепенно перестают воспринимать их как признак необычного события.

Другой случай хуже замечен в журнале тревог: коробка действительно идёт не так, как ожидается, но предупреждения нет. Линия продолжает работу, и отклонение обнаруживается только после пропуска изделия или при следующей проверке. Это пропущенная проблема.

- Значит, опустим порог и поймаем всё раньше, - сказал разработчик.

Начальник смены показал на журнал.

- Тогда сначала посмотри, сколько исправных циклов окажется за этой чертой.

Разработчик передвинул порог на графике. Число отмеченных проходов выросло. Он поднял его - предупреждений стало меньше, но вчерашний цикл оказался ближе к моменту, когда уже трудно что-либо изменить. Они не выбирали между «строгой безопасностью» и «удобством производства». Они пытались понять, в какой момент сообщение ещё помогает принять решение и насколько часто оно требует внимания без достаточного основания.

Не всякая тревога требует остановки

Оператор указал на задержанный, но успешный цикл из предыдущей смены.

- Если коробка дошла до робота и результат проверен, зачем мне сообщение на пульте?

- Возможно, тебе оно и не нужно в этот момент, - ответил инженер. - Но для наладчика повторяющиеся задержки могут быть поводом проверить участок.

Они разделили два события, которые раньше смешивали. Первое - время прохода стало необычным, хотя нужные подтверждения получены и изделие обработано. Такое отклонение можно сохранить для наблюдения и проверки, не прерывая каждый цикл. Второе - к моменту решения

обязательного подтверждения нет. Здесь уже нельзя делать вид, что положение коробки известно.

Разработчик добавил на схему разные реакции. Для одного случая - запись и понятное уведомление при повторении. Для другого - запрет действия, которое требует неподтверждённого положения изделия. Что должно происходить с конвейером и соседними коробками, ещё предстояло проверить на самом участке.

Начальник смены прочитал подписи.

- Это лучше, чем одна красная лампа на всё. Но мне нужно знать, какая задержка считается обычной.

Инженер согласился. Порог нельзя было выбрать по одному вчерашнему происшествию. Нужны записи нормальных проходов при разных коробках и условиях работы, затем проверка того, какие реальные отклонения система обнаруживает и сколько бесполезных сообщений создаёт. Важна была не только частота срабатываний, но и цена каждого пропуска.

Цена зависит от следующего действия

На столе лежала вчерашняя коробка с примятым углом. Если подобное изделие пройдёт без обработки, последствия зависят от устройства всей линии: его могут заметить при последующем контроле, а могут не заметить. Если же робот попытается захватить коробку в неверном положении, возникнет уже другая ситуация. Одинаковой реакции на обе возможности никто не стал назначать заранее.

- Нам нужно знать, что линия делает после предупреждения, - сказал оператор. - Если она только пишет «задержка», а коробка всё равно идёт дальше, я могу не успеть её увидеть.

Эта фраза остановила спор о положении черты на графике. Предупреждение полезно, только если за ним следует действие, которое можно выполнить вовремя. У системы должно быть достаточно сведений, чтобы выбрать свой следующий шаг; у человека - достаточно времени и ясная задача, если решение передают ему.

Герои решили сначала испытать предлагаемые пороги на сохранённых записях, не подключая новые предупреждения к работающей линии. Так можно было увидеть, какие прошлые циклы они отметили бы и сколько раз подняли бы тревогу без видимого нарушения. После этого предстояло проверить реакции на контролируемых отклонениях.

Разработчик снова открыл вчерашний цикл. На графике он выделялся, но не так резко, как ожидали. По одной только задержке его было трудно отделить от нескольких успешно обработанных коробок.

- Значит, времени недостаточно, - сказал он.

Инженер кивнул.

- Возможно. У нас ещё есть камера и сведения о движении. Посмотрим, что они говорили об этой же коробке.

Посмотрите как инженер. Если снижение порога предупреждения позволит заметить больше задержанных коробок, какой показатель нужно проверить прежде, чем выводить каждое такое предупреждение на пульт оператора?

Что осталось за кадром. Герои пока сравнили записи и возможные реакции, а не утвердили настройку для эксплуатации. Без испытаний нельзя обещать, что выбранный порог обнаружит будущие сбои или что число ложных тревог останется приемлемым.

Глава 8. Когда данные спорят

Разработчик поместил на один экран три записи проблемного цикла. По сигналу датчика коробка достигла направляющей. По движению конвейера она должна была приблизиться к месту захвата. На снимке камеры изделие находилось заметно в стороне от расчётного положения.

- Что-то из этого неверно, - сказал начальник смены.

Инженер по надёжности не стал выбирать подозреваемого. Он попросил подписать каждую запись так, чтобы было ясно, о чём она сообщает.

Над сигналом датчика разработчик написал: «Луч перекрыт». Над данными конвейера: «Лента прошла это расстояние». Над снимком: «Коробка видна здесь в момент съёмки».

Начальник смены прочитал подписи ещё раз.

- Они даже не утверждают одно и то же.

Что измеряли приборы

Датчик у направляющей находился в одной точке. Если край коробки перекрыл его луч, сигнал менялся независимо от того, стояло ли изделие ровно. Датчик подтверждал присутствие предмета в своей зоне, а не положение всей коробки перед роботом.

Данные конвейера относились к движению ленты. На их основе программа оценивала путь изделия. Для такой оценки требовалось допущение: коробка движется вместе с лентой предсказуемо и не меняет заметно своего положения относительно неё.

Камера показывала видимую часть коробки на отдельном снимке. Она могла помочь проверить положение изделия, но снимок относился к своему моменту времени. Сравнивать его с расчётом для более позднего момента без поправки было бы ошибкой.

Разработчик сначала совместил записи по отметкам времени. Потом вернулся к журналу контроллера и проверил, к какому событию привязано сохранение кадра. Выяснилось, что камера получила команду на съёмку до того, как датчик у направляющей зарегистрировал коробку. На прежнем экране кадр стоял рядом с поздним расчётным положением, и это создавало впечатление прямого противоречия.

- Значит, камера ни при чём? - спросил оператор.
- Она показала то, что было раньше, - ответил разработчик. - Это не отменяет разницы с расчётом. Но сначала нужно сравнить сведения об одном и том же времени.

Он пересчитал ожидаемое положение коробки для момента съёмки. Расхождение стало меньше. Полностью оно не исчезло.

Противоречие как улика

Наладчик снова взглянул на коробку, стоявшую у стены.

- Если её разворачивало у направляющей, лента могла пройти своё расстояние, а сама коробка оказаться не там, где её ждали.

- Да, - сказал инженер. - Но снимок сделан до сигнала у направляющей. Он не показывает сам разворот.

Оставшееся различие могло возникнуть по нескольким причинам. Коробка уже шла с небольшим перекосом. Оценка её начального положения была неточной. Изображение и данные движения относились к моментам, которые они ещё не сумели сопоставить достаточно точно. Нельзя было объявить камеру неисправной только потому, что её сообщение не совпадало с расчётом.

И всё же несовпадение было полезным. Система ожидала, что разные источники сложатся в одну картину движения изделия. В этом цикле картина не складывалась без дополнительных допущений. Такое расхождение можно обнаружить, даже не зная его причины.

- Почему программа не показала это оператору? - спросил начальник смены.

Разработчик открыл настройки проверки. В предусмотренном цикле каждый источник выполнял свою задачу. Датчик участвовал в переходе между шагами, сведения конвейера - в расчёте положения, камера - в проверке на своём этапе. Отдельного правила, которое сопоставляло бы все три сообщения перед дальнейшим действием, не было.

Это не означало, что нужно просто добавить правило «если данные различаются, остановить всё». Сначала следовало установить, какое различие допустимо из-за времени съёмки и погрешностей измерения, а

какое говорит, что положение изделия больше нельзя считать подтверждённым.

Оператор указал на два обычных цикла, где снимок и расчёт тоже не совпадали в точности.

- Иначе мы опять получим тревогу на каждую коробку.
- Поэтому проверять надо не совпадение до последней цифры, - сказал инженер. - Надо определить, можем ли мы при этом расхождении безопасно сделать следующий шаг.

Чего не хватало вчера

В проблемном цикле подтверждение у направляющей пришло поздно. Камера показывала положение, отличавшееся от расчётного, хотя часть разницы объяснялась временем съёмки. Каждое наблюдение по отдельности ещё позволяло спорить о причине. Вместе они уже не поддерживали уверенный вывод: коробка находится там, где её ожидают к моменту захвата.

Разработчик записал новую задачу для проверки алгоритма: при расхождении источников система должна не выбирать привычно «главный» датчик, а уточнять, какое положение действительно подтверждено и можно ли действовать без него. Иногда для этого хватит следующего наблюдения. Иногда нужный момент пройдёт раньше, чем появятся новые сведения.

Начальник смены посмотрел на экран с тремя подписями.

- Можно ли сделать так, чтобы она сообщала не «камера ошиблась», а «положение не подтверждено»?

- Это точнее, - ответил инженер. - Но нам придётся определить, когда такое сообщение остаётся предупреждением, а когда запрещает следующий шаг.

Расследование больше не сводилось к поиску устройства, которому следует верить всегда. Источники рассказывали о разных частях одного движения. Вчера система не смогла сложить их в достаточно надёжную картину, но продолжила выполнять предусмотренные переходы. Теперь героям предстояло решить, как машина должна обозначать предел собственной уверенности.

Посмотрите как инженер. Прежде чем признать показания камеры и расчётного положения противоречащими друг другу, какие два уточнения нужно сделать о снимке и расчёте?

Что осталось за кадром. Сопоставление трёх источников выявило расхождение, но не показало точную траекторию вчерашней коробки. Причину оставшейся разницы ещё предстоит проверить; нельзя автоматически считать один прибор верным, а другой неисправным.

Глава 9. Может ли система сказать «я не уверена»?

На следующем совещании разработчик вывел на экран изображение коробки. Поверх него программа отметила предполагаемое место захвата. Рядом стояло число: 94%.

- Камера уверена почти полностью? - спросил начальник смены.
- Это показатель, который выдаёт алгоритм, - ответил разработчик. - Я бы не называл его вероятностью того, что захват будет успешным.

Инженер по надёжности попросил открыть несколько соседних кадров. На одном коробка стояла ровно, на другом её край закрывала часть направляющей. Показатель оставался высоким в обоих случаях. Между тем второй кадр оставлял важный вопрос: где находилась скрытая часть изделия и каким будет его положение к моменту захвата?

Число выглядело точным. Сведения, на которых оно основывалось, полными не были.

Уверенность в ответе и основания для действия

Алгоритм камеры выбирал положение коробки по изображению. Его показатель описывал, насколько определённым получился этот выбор по правилам самого алгоритма. Он не учитывал автоматически всё, что происходило за пределами кадра, после съёмки или в механике участка.

Разработчик закрыл число ладонью.

- Допустим, мы его не видим. Что нам нужно знать, прежде чем дать роботу команду?

Наладчик ответил: положение коробки у направляющей. Оператор добавил: успела ли она дойти до места захвата. Инженер уточнил ещё одно: подтверждают ли датчик, камера и расчёт положения одну и ту же картину в нужный момент.

Высокий показатель алгоритма не восполнял отсутствие этих сведений. Он мог быть полезен, когда условия съёмки и изделия похожи на те, на которых проверяли распознавание. Но даже тогда решение о движении робота требовало подтверждений, предусмотренных для самого действия, а не только убедительного числа возле снимка.

- Выходит, система должна писать «не уверена» всякий раз, когда источники расходятся? - спросил начальник смены.

- Не обязательно писать именно так, - сказал инженер. - Но она должна различать подтверждённое положение и положение, которое только предполагает.

У неопределённости бывают разные последствия

Герои рассмотрели вчерашний цикл с этой точки зрения. Камера показывала изделие на одном кадре. Датчик позже подтвердил перекрытие луча. Расчёт по движению ленты давал ожидаемое положение. Ни одно из сообщений само по себе не позволяло уверенно указать место захвата в нужный момент.

Если изделие ещё далеко, система может дожидаться следующего наблюдения и уточнить положение. Если коробка уже подходит к месту, где решение нельзя откладывать, ожидание перестаёт быть нейтральным: конвейер продолжает движение. Тогда требуется заранее определённая реакция, которая не опирается на неподтверждённый захват.

Оператор посмотрел на схему.

- А если камера сомневается, но датчик и движение ленты совпадают?
- Зависит от того, что именно камера не смогла подтвердить, - ответил разработчик. - Для одного шага этого может быть достаточно. Для другого - нет.

Сомнение само по себе не задаёт команду «стоп». Сначала нужно назвать неизвестное. Неясно, есть ли коробка? Неясно её положение?

Неизвестно, удерживает ли её робот после захвата? Для разных вопросов полезны разные проверки, а последствия ошибочного действия тоже различаются.

Разработчик предложил на испытаниях отмечать не только итоговый ответ камеры, но и случаи, когда видимость коробки ограничена или её положение расходится с другими сведениями. Инженер попросил отдельно проверить, как ведёт себя алгоритм на изделиях, которые отличаются от привычных. Если условия изменились, высокий показатель мог остаться высоким, даже когда ответ перестал заслуживать доверия.

Начальник смены вернулся к числу на экране.

- Значит, 94% не даёт мне разрешения на пуск?
- Нет, - сказал разработчик. - Оно относится к одному выводу из одного изображения. Для пуска нам нужен ответ о состоянии всего участка.

Эта разница была неудобной: одно число легко поместить на экран, а границы его применимости - гораздо труднее. Герои решили не показывать оператору показатель без пояснения, к какому наблюдению он относится. Сообщение должно было говорить прежде всего о том, что система смогла подтвердить и какое действие из-за отсутствующих сведений не выполняет.

Разговор прервал наладчик. Он принёс выписки о времени движения одного из узлов за несколько недель. В последние дни оно менялось понемногу, без аварий и резких скачков.

- Возможно, мы искали задержку только возле коробки, - сказал он. - А часть времени теряли ещё здесь.

Посмотрите как инженер. Почему высокий показатель распознавания на снимке не отвечает на вопрос, можно ли начинать захват движущейся коробки?

Для любознательных. Показатель уверенности алгоритма и проверенная частота правильных ответов - не одно и то же. Чтобы понять, насколько числу можно доверять, алгоритм испытывают на подходящих примерах и отдельно смотрят, что происходит при изменении изделий, освещения и условий съёмки.

Что осталось за кадром. Герои пока не проверяли камеру на изменённых условиях и не установили, влиял ли её вывод на вчерашнее решение о захвате. Число на экране служит поводом уточнить основания решения, а не доказательством ошибки алгоритма.

Глава 10. Медленный сбой

Наладчик принёс не график аварий, а записи о работе привода подающего участка. В них отмечалось, сколько времени проходило от команды на движение до достижения заданного положения. За несколько недель это время понемногу увеличилось.

Ни в один отдельный день изменение не выглядело необычным. Привод выполнял команду, линия выпускала продукцию, сообщение о неисправности не появлялось. Если смотреть только на вчерашнюю остановку, эти записи легко было пропустить.

- Почему ты обратил на них внимание? - спросил разработчик.

- После обслуживания узел двигался чуть резвее. На слух я разницы теперь не замечаю, а по журналу она есть.

Инженер по надёжности попросил наладчика не связывать находку с пропущенной коробкой раньше времени. Они ещё не знали, менялась ли скорость движения именно в нужной части цикла и могла ли эта перемена повлиять на момент появления сигнала у направляющей.

Когда каждый цикл исправен

Разработчик вывел значения по дням. Отдельные измерения колебались: иногда привод доходил до положения быстрее, иногда медленнее. Но если сравнить начало и конец периода, сдвиг был виден.

Система, которая проверяет лишь факт выполнения команды за максимально допустимое время, могла считать все эти циклы одинаково успешными. Для поиска медленного изменения полезен другой взгляд: не только завершилось ли движение, но и как менялось время его выполнения при сходных условиях.

Наладчик предложил проверить крепления, состояние механизма и нагрузку привода. Он не утверждал, что нашёл изношенную деталь. Увеличение времени могло быть связано не только с износом: на измерение влияют режим работы, изделие и состояние участка. Сначала следовало убедиться, что сравнивают сопоставимые циклы.

Начальник смены посмотрел на график.

- Если это началось давно, почему никто не получил предупреждение?

- Потому что предусмотренный предел ни разу не был превышен, - ответил разработчик. - Программа отвечала на вопрос «выполнено ли движение вовремя?». Вопрос «меняется ли его время неделя за неделей?» мы ей не задавали.

Оператор вспомнил, что иногда коробки подходили к направляющей чуть иначе, чем он ожидал. Он не записывал это как неисправность: линия продолжала работать. Теперь его наблюдение стало поводом для проверки, но не подтверждением связи с графиком привода. Память о нескольких проходах не заменяла измерений, а измерения не объясняли сами собой увиденное оператором.

Предупреждать, не обещая дату отказа

Инженер предложил сохранять времена движения вместе с условиями работы и разбирать устойчивые изменения при обслуживании. Такой подход мог помочь назначить проверку раньше, чем механизм перестанет выполнять команду.

Начальник смены задал вопрос, который возник бы у любого, кому предстоит планировать работу:

- По этому графику можно сказать, сколько привод ещё проходит?
- Нет, - ответил наладчик. - Мы даже не установили причину изменения.

Продолжить линию графика вперёд было бы легко. Полученная дата выглядела бы убедительно, но держалась бы на предположении, что механизм и условия продолжают меняться тем же образом. В реальной работе

износ может ускориться, замедлиться или оказаться вовсе не причиной замеченного сдвига. Тренд давал основание осмотреть узел, а не обещание точного дня поломки.

Для вчерашнего происшествия находка оставалась одной из проверяемых версий. Если подающий участок начал менять движение коробок, он мог уменьшить запас времени до подтверждения. Чтобы это установить, нужно было сопоставить записи привода и датчиков в одних и тех же циклах, а затем проверить механизм. Пока такой связи не было.

Инженер добавил к рабочей схеме ещё одну пометку: ожидания системы могут перестать соответствовать участку не только внезапно. Иногда оборудование меняется понемногу, а каждое отдельное действие ещё укладывается в допустимые пределы.

Перед уходом разработчик показал результаты повторной проверки упаковочной линии. После вчерашнего разбора они умели раньше замечать задержку и несогласованность сведений. Но оставался более трудный вопрос: что делать, когда расхождение уже обнаружено, а его причина ещё неизвестна? Остановить участок полностью, дать закончить текущий цикл или перейти к ограниченной работе - одинакового ответа для всех случаев не было.

Посмотрите как инженер. Какие данные нужно сопоставить с увеличившимся временем движения привода, чтобы проверить, связано ли оно с запаздывающим подтверждением коробки?

Что осталось за кадром. График показывает изменение времени движения, но пока не устанавливает ни износ, ни его связь со вчерашней

остановкой. Для вывода нужны сопоставимые условия измерения и проверка самого узла.

Интерлюдия. Станок

Инженер по надёжности принёс на совещание другую запись: «Цикл завершён без ошибки». Она относилась не к упаковочной линии, а к станку, который обработал деталь.

- По этой строке можно выпускать деталь? - спросил он.
- Если станок закончил обработку, почему нет? - ответил начальник смены, но тут же посмотрел на инженера. - Вы собираетесь показать размер вне допуска.

Именно это и произошло. Станок выполнил предусмотренные движения, не остановился и не сообщил об аварии. Позже измерение показало, что один размер детали не соответствует требованию. Запись о завершённом цикле была верной, но отвечала на более узкий вопрос: закончила ли система заданную последовательность. Она не подтверждала качество полученного изделия.

Оператор вспомнил вчерашнюю коробку.

- У нас проверка тоже была. Камера ведь смотрела на результат.
- Была, - согласился разработчик. - Только надо уточнить, какой результат она могла подтвердить и когда. Если коробка уже прошла место захвата, поздняя проверка обнаружит нарушение, но не вернёт её назад.

Пример со станком не подсказывал, какой контроль поставить на упаковочной линии. Для детали важен размер, для коробки - положение и итог обработки. Общим был вопрос, который легко пропустить за строкой «цикл завершён»: **проверила ли система тот результат, ради которого выполняла действие?**

Разработчик вернулся к схеме расследования и задержал палец на последней клетке - «проверка результата». До сих пор герои уделяли больше внимания сигналам перед захватом. Теперь им предстояло решить, что линия должна делать, когда ожидаемый результат нельзя подтвердить.

Что осталось за кадром. Способ проверки зависит от изделия и процесса. Из того, что станку потребовалось измерение детали, не следует, что упаковочной линии нужен такой же прибор или такой же порядок остановки.

Часть III. Что делать, когда ясности нет

Глава 11. Остановиться, продолжить, замедлиться

Утром начальник смены попросил героев ответить на практический вопрос. В новой версии программы коробка снова приходит к месту захвата без достаточно надёжного подтверждения положения. Что делает линия?

Разработчик написал на доске три предложения: остановить участок, продолжить цикл без изменений, перейти к ограниченной работе с дополнительной проверкой.

- Третье звучит разумнее остальных, - сказал оператор.

- Пока мы не объяснили, что именно разрешено в этом режиме, оно ничего не значит, - возразил наладчик.

На вчерашней линии слово «продолжить» скрывало несколько разных движений. Можно не давать роботу команду на захват, но оставить конвейер работающим. Можно остановить подачу новых коробок, однако дать уже начатому перемещению завершиться. Можно попросить оператора осмотреть изделие. Каждое решение меняло состояние участка, а не просто положение переключателя на экране.

Когда остановка помогает

Инженер по надёжности предложил начать с самого трудного случая: система не знает, где находится коробка относительно зоны действия робота. Для захвата это сведение необходимо. Если траектория рассчитана для предполагаемого положения, продолжать движение на таком основании нельзя.

Остановка, однако, тоже требовала продуманного порядка. Что происходит с коробками, уже находящимися на конвейере? Какие механизмы завершают движение, а какие прекращают его? Остаётся ли изделие у направляющей, где его может задеть следующий пуск? Нажать «стоп» и получить безопасное, понятное состояние - не одно и то же, если заранее не определить поведение оборудования.

- Я вчера остановил участок, когда увидел коробку дальше робота, - сказал оператор. - После этого мне всё равно пришлось выяснять, где остальные.

Его решение предотвратило дальнейшее развитие видимого нарушения. Но оно не сообщило системе задним числом, какие операции были завершены и что осталось внутри участка. Эти вопросы нельзя было оставлять до повторного пуска.

Почему продолжение не бывает нейтральным

Разработчик показал старое правило перехода после истечения времени ожидания. Робот не делал захват вслепую. Зато конвейер продолжал следующий шаг, исходя из того, что неподтверждённое изделие не создаст помехи. Вчера это предположение оказалось недостаточно проверенным.

- То есть мы уже пробовали второй вариант, - сказал начальник смены.
- Не в виде осознанного решения для перекошенной коробки, - уточнил разработчик. - В программе это было обычным переходом после отсутствия подтверждения.

Продолжение работы удобно описать как отсутствие вмешательства. На деле это тоже команда системе: двигать изделия и механизмы дальше при определённом наборе неизвестных. Если неизвестное касается положения предмета на пути робота или следующей коробки, бездействием такое решение назвать нельзя.

Ограниченный режим нужно спроектировать

Третью возможность герои разобрали внимательнее. Допустим, положение отдельной коробки не подтверждено, но система может прекратить подачу новых изделий и удержать робота от захвата. После этого

оператору можно сообщить, какой участок требует осмотра. В некоторых обстоятельствах часть функций могла бы оставаться доступной - например, отображение состояния и сохранение записей.

Но нельзя просто «замедлить конвейер» и считать проблему решённой. Меньшая скорость сама по себе не подтверждает положение коробки и не гарантирует, что человек успеет заметить её вовремя. Для ограниченного режима требовалось заранее определить, при каких условиях он включается, какие движения допускает, как контролируется результат и когда работа должна прекратиться полностью. Затем это поведение нужно испытать на данном участке.

Начальник смены перечитал три предложения на доске.

- Получается, мы не выбираем любимый режим. Сначала выясняем, что может случиться в каждом.
- И что система знает перед выбором, - добавил инженер.

Они договорились описать реакции отдельно для нескольких ситуаций: подтверждение запоздало, источники дали несовместимые сведения, результат захвата не установлен. Пока никто не разрешал ограниченную работу на действующей линии. На бумаге она выглядела привлекательной; в цехе её ещё требовалось проверить.

Разработчик стёр с доски слова «остановить всё» и «продолжить как прежде». Вместо них он начал записывать конкретные действия и необходимые для них подтверждения. Так выбор переставал зависеть от общего ощущения, что одна реакция слишком сурова, а другая слишком удобна.

Посмотрите как инженер. Если система не подтвердила положение коробки, какие движения участка нельзя разрешать лишь на основании того, что робот не получил команду на захват?

Что осталось за кадром. Для упаковочной линии пока обсуждают возможные реакции, а не готовые инструкции по безопасности. Допустимость каждого режима зависит от устройства участка, расположения людей и оборудования, предусмотренных мер защиты и результатов испытаний.

Глава 12. Риск важнее удобства

На следующем листе разработчик изобразил коробку в трёх положениях: перед камерой, у места захвата и за роботом. Под каждым рисунком он написал одно и то же: «Положение не подтверждено».

Начальник смены посмотрел на лист.

- Сообщение одинаковое. А делать будем разное?
- Именно это и нужно проверить, - ответил инженер по надёжности.

Пока коробка находилась до места захвата, у системы ещё могла быть возможность получить дополнительное наблюдение. Когда изделие подошло к роботу, решение о его движении уже зависело от подтверждённого положения. Если коробка прошла дальше, задача менялась: требовалось установить, где она остановилась и не мешает ли дальнейшей работе.

Одни и те же слова на экране скрывали разные последствия ошибочного решения.

Что можно исправить позже

Герои начали с качества продукции. Если коробка не была обработана и это надёжно обнаружили до выхода изделия с участка, её судьбу можно решить отдельно. Возможно, потребуется остановить подачу, отвести изделие по предусмотренному маршруту или передать его на проверку. Какой вариант доступен, зависело от конструкции линии.

Другое дело - команда на захват, когда положение коробки неизвестно. Если робот начнёт движение к неверно рассчитанному месту, последующая проверка качества уже не отменит самого движения. Поэтому подтверждение, необходимое для захвата, должно существовать до команды, а не появиться в отчёте после неё.

Наладчик добавил ещё один сценарий: перекошенная коробка могла застрять между направляющей и соседним механизмом. Здесь речь шла не только о пропущенной операции, но и о возможном повреждении оборудования. Продолжать подачу новых изделий, не выяснив положение застрявшего, означало бы увеличивать последствия отклонения.

Оператор указал на место, откуда он вчера заметил коробку.

- А если мне предлагают подойти и посмотреть, линия должна быть в понятном состоянии. Иначе «проверьте коробку» - плохая инструкция.

Инженер записал это отдельно. Решение, которое передают человеку, меняет и его действия. Нельзя оценивать реакцию системы только по судьбе изделия, не учитывая, куда и при каком состоянии оборудования она направляет оператора.

Риск - это не название аварии

Разработчик предложил назначить одну реакцию на сообщение «Положение не подтверждено». Так было бы проще запрограммировать и объяснить. Инженер попросил сначала уточнить, что именно может произойти после каждого разрешённого действия.

Если система ошибочно задержит исправную коробку, последствием станет ненужное вмешательство и простой. Если она пропустит необработанное изделие, последствие зависит от того, существует ли дальше надёжный контроль. Если разрешит движение механизма при неизвестном положении предмета, оценивать придётся уже возможность столкновения и то, может ли человек оказаться в опасной зоне. Эти случаи нельзя считать равноценными только потому, что начались с одного недостающего сигнала.

Начальник смены вернулся к спору о чувствительности контроля.

- Выходит, стоимость ложной тревоги тоже надо учитывать, но не одинаково во всех ситуациях.

- Да, - сказал инженер. - Иначе мы либо останавливаем всё при любом сомнении, либо одинаково спокойно относимся к сомнению перед записью в журнал и перед движением робота.

Они решили описывать не абстрактную «неопределённость», а конкретное неизвестное и следующее действие. Можно ли получить подтверждение до движения? Можно ли позже обнаружить и исправить ошибку? Что произойдёт, если предположение окажется неверным? Эти вопросы помогали обсуждать реакцию без формулы «чем больше тревог, тем безопаснее».

Решение до ответа

Для вчерашней коробки окончательная причина задержки всё ещё оставалась открытой. Но отсутствие диагноза не мешало принять часть решений. Герои уже знали, что захват при неподтверждённом положении нельзя обосновывать одним ожидаемым временем прибытия. Они знали и то, что сообщение после пропуска коробки появляется слишком поздно, чтобы предотвратить этот пропуск.

Оставалось определить, как система передаст человеку ситуацию, которую сама не смогла разрешить. Разработчик открыл действующее сообщение: «Нарушение цикла». Оператор прочитал его и покачал головой.

- Вчера оно не сказала мне, какую коробку искать. И не сказала, что робот уже сделал.

На этот раз спор закончился не выбором порога и не решением об остановке. Герои начали переписывать сообщение.

Посмотрите как инженер. Почему проверка изделия после работы робота может быть достаточной для одной задачи, но не заменяет подтверждения положения перед командой на захват?

Что осталось за кадром. Герои сравнивают возможные последствия, не присваивая им универсальные оценки. Конкретные меры защиты и допустимые режимы работы должны определяться для устройства и условий именно этой линии; пример с коробкой не заменяет такой проверки.

Глава 13. Кто принимает решение после тревоги

Разработчик вывел на экран прежнее сообщение крупным шрифтом:

«Нарушение цикла».

Оператор посмотрел на него, затем на остановленную линию.

- Это я и без экрана вижу.

Инженер по надёжности попросил его вспомнить первые минуты после вчерашней остановки. Что пришлось выяснять самому? Оператор назвал положение пропущенной коробки, состояние робота и вопрос о том, успела ли система начать захват. Начальника смены интересовало другое: какие изделия уже обработаны и можно ли готовить участок к повторному пуску.

Одно сообщение не обязано объяснять всё происшествие. Но оно должно помогать принять ближайшее решение, а не оставлять человека один на один с остановленным оборудованием и загадочной надписью.

Что случилось, а что только предполагается

Разработчик предложил заменить старую строку на «Неисправен датчик коробки». Формулировка была короткой и понятной.

- Только мы этого не установили, - напомнил инженер.

В журнале был сигнал датчика. Вопрос заключался в том, что подтверждение не оказалось доступным в нужный момент и положение изделия перед следующим шагом осталось неясным. Назвать датчик неисправным означало бы превратить одну из версий в готовый диагноз. Оператор мог бы начать проверять датчик, тогда как коробка находилась бы дальше по линии.

На экране появилась другая запись: «Положение коробки перед захватом не подтверждено».

- Уже лучше, - сказал оператор. - Но что сделала линия?

Разработчик открыл подробный журнал. Для обсуждаемого случая он мог подтвердить, что команда на захват не выдавалась, подача новых коробок была прекращена, а сведения о последнем цикле сохранены. Он добавил эти сведения в проект сообщения. Отдельной строкой указал зону, где следовало проверить наличие изделия.

Теперь текст сообщал о трёх разных вещах: что обнаружено, чего система не знает и какие действия уже выполнены. Предположения о причине в нём не было.

Передать управление - не значит убрать автоматику

Начальник смены предложил добавить кнопку «Ручной режим».

Оператор возразил:

- Я не смогу принять решение по одной кнопке. Мне нужно понимать, какая коробка не обработана и где она может быть.

Ручное управление не восполняет недостающие сведения. Если человеку передают ситуацию после тревоги, он должен знать состояние оборудования в пределах того, что система действительно может подтвердить. Ему также нужно ясно видеть, чего система подтвердить не смогла. Иначе оператору приходится угадывать по общему виду линии то, что программа уже записала - или, наоборот, доверять экрану там, где запись неполна.

Разработчик предложил показывать вместе с сообщением схему участка с отмеченной зоной неопределённости. Инженер попросил не

ставить на ней точку в предполагаемом положении коробки: точка выглядела бы как измеренная координата. Участок, в пределах которого изделие требовало проверки, честнее отражал доступные сведения.

- И без сотни параметров на первом экране, - добавил начальник смены.

Подробный журнал был нужен наладчику и разработчику. Оператору в момент остановки требовалось другое: понять, какое действие не выполнено, что уже остановлено и какой участок нельзя считать свободным без проверки. Полные записи при этом должны оставаться доступными для расследования.

Проверка текста в цехе

Герои показали черновик сообщения оператору другой смены, который не участвовал во вчерашнем разборе. Он прочитал его и спросил:

- «Подача прекращена» - это про входной конвейер или про всю линию?

Разработчик исправил формулировку. Проверка оказалась полезнее спора у ноутбука: автор сообщения знал, какие механизмы имел в виду, а человек у пульта не обязан был догадываться.

Затем оператор заметил ещё одну неточность. Надпись «Проверьте коробку у робота» звучала как указание сразу идти к механизму. Её заменили сообщением о необходимости осмотра зоны после приведения участка в состояние, предусмотренное для такого вмешательства. Порядок безопасного доступа не стали заменять одной строкой на экране.

К концу встречи сообщение перестало изображать диагноз, которого у системы не было. Оно называло обнаруженное расхождение, границу известных сведений и выполненные действия. Начальник смены прочитал его ещё раз.

- С таким текстом я хотя бы понимаю, почему одного удаления коробки недостаточно для пуска.

Инженер согласился, но указал на вчерашнюю сменную запись. В ней говорилось, что коробки «иногда идут у направляющей неровно». Никто не уточнил, когда это началось и кому об этом сообщили.

Экран можно было сделать яснее. Но часть сведений о линии всё равно оставалась у людей.

Посмотрите как инженер. Почему сообщение «Неисправен датчик» могло бы помешать разбору вчерашнего случая, даже если оператору оно кажется понятнее, чем «Положение не подтверждено»?

Что осталось за кадром. Текст сообщения пока проверен только как черновик. До применения на участке нужно подтвердить, что описанные в нём действия автоматики действительно выполняются при каждом соответствующем сценарии, а порядок вмешательства согласован с устройством линии.

Глава 14. Человек тоже работает с неполной информацией

Инженер по надёжности открыл сменный журнал. За несколько дней до остановки там появилась короткая запись: «Коробки иногда идут неровно у направляющей». Ни времени, ни номера партии, ни пояснения, что именно видел оператор, рядом не было.

- Я это писал, - сказал оператор предыдущей смены, которого пригласили на разбор. - Две коробки повернуло, но робот обе взял. Потом всё шло нормально.

Наладчик вспомнил разговор возле линии. Он предложил понаблюдать за направляющей, однако заявку на проверку не оформил: оснований останавливать участок тогда не увидел. Новый оператор запись прочитал, но не знал, что подобных проходов было два и что наладчик уже обратил на них внимание.

Каждый передал часть картины. Вместе эти части могли бы подсказать, где искать изменение поведения линии, но до происшествия они так и не встретились в одном месте.

Когда тревогам перестают верить

Начальник смены снова принёс журнал остановок. В нём были случаи, когда экран требовал вмешательства, а после осмотра явной неисправности не находили. Операторы всё равно выполняли положенные действия, но короткие сообщения вроде «Нарушение цикла» постепенно перестали помогать им различать события.

- Когда одно и то же сообщение означает разные ситуации, начинаешь смотреть не на него, а на линию, - сказал оператор.

Это не было признаком невнимательности. Экран действительно давал слишком мало сведений. Однако привычка полагаться только на видимое поведение участка могла скрыть отклонение, которое ещё не дошло до пропущенной коробки.

Редкие тревоги создавали другую трудность. Если необычный случай возникает впервые за долгое время, человеку приходится восстанавливать порядок действий под давлением остановки. Вчера оператор быстро заметил коробку и остановил участок, но никто не мог сразу ответить, что произошло в незавершённом цикле. Такой ответ нельзя требовать от памяти одного человека, если сведения распределены между экраном, журналом и состоянием оборудования.

Сведения между сменами

Инженер попросил обоих операторов описать запись, которая помогла бы следующей смене. Получился короткий, но более полезный текст: в какое время коробки поворачивались у направляющей, сколько таких случаев заметили, чем они закончились и проверял ли кто-нибудь узел после этого.

Наладчик добавил дату последнего обслуживания направляющей и результат проверки контрольной коробкой. Он отдельно отметил то, чего не измерял. Теперь запись не выдавала предположение «направляющая сместилась» за установленный факт, но позволяла следующему человеку понять, почему на неё обратили внимание.

Разработчик предложил сохранять вместе с такими наблюдениями данные о времени прохождения изделия. Если оператор отмечает неровное движение, а система записывает задержку в том же цикле, появляется основание для предметной проверки. Без привязки ко времени два сообщения могут относиться к разным коробкам.

Начальник смены посмотрел на новый образец записи.

- Не хочу, чтобы каждый небольшой перекося превращался в длинный отчёт.

- И не нужно, - ответил инженер. - Достаточно оставить следующей смене проверяемое наблюдение, а не только впечатление.

Человек не последний датчик

На обсуждении разработчик сказал, что теперь оператор сможет раньше замечать необычное движение коробок. Инженер остановил его.

- Не будем строить защиту на обещании, что человек всегда успеет это увидеть.

Оператор следил одновременно за несколькими участками. Коробка могла повернуться, пока он занимался другой задачей. Экран мог показать непонятную тревогу, а журнал - сохранить важную задержку без какого-либо сообщения. Надёжная работа требовала согласованности этих источников, а не надежды на безошибочную внимательность человека.

При этом наблюдения оператора оставались незаменимыми для расследования. Он видел, как коробка скользила вдоль направляющей, и мог заметить перемену, которую программа пока не выделяла из записей. Но такое наблюдение нужно было сделать доступным остальным: точно описать, передать между сменами и сопоставить с данными системы.

К концу встречи герои изменили форму сменной записи и договорились проверить, какие тревоги чаще всего заканчиваются без установленной причины. Если сообщение бесполезно, его недостаточно просто повторять громче. Нужно выяснить, что оно обозначает и какое решение помогает принять.

Начальник смены закрыл журнал. Теперь, когда происхождение вчерашнего сбоя было яснее, он вернулся к вопросу, с которого началось расследование:

- Что именно мы должны проверить перед повторным пуском?

Посмотрите как инженер. Какая подробность превратила бы прежнюю запись «коробки иногда идут неровно» в свидетельство, которое можно сопоставить с данными системы?

Что осталось за кадром. Новая форма записи не гарантирует, что каждое отклонение заметят и передадут. Она помогает сохранить наблюдения людей и проверить их, но не заменяет технического контроля там, где решение требуется быстрее, чем человек может отреагировать.

Глава 15. Почему повторный пуск - отдельная операция

Коробка, из-за которой остановилась линия, стояла у стены. Направляющую осмотрели. Робот находился в неподвижном положении, а на пульте больше не было новых сообщений.

- Если причина в перекосе, мы её убрали, - сказал начальник смены. - Что ещё мешает запуску?

Инженер по надёжности попросил подождать и показал на конвейер.

- Эта коробка была единственной, которую система не обработала? Где остановилась следующая? Что программа считает завершённым циклом?

На первый вопрос можно было ответить осмотром, но не взглядом с одного места. Второй требовал проверить участок подачи. Ответ на третий

находился в журнале контроллера, и герои ещё не сопоставили его с тем, что осталось на линии.

Удалённое изделие решало видимую часть проблемы. Оно не возвращало автоматически машину и её записи в согласованное состояние.

Что осталось после остановки

Разработчик открыл состояние программы на момент остановки. Система успела перейти от ожидания подтверждения к следующему шагу. Захват для проблемной коробки не выполнялся. Для следующего изделия цикл был начат, но не завершён.

Оператор осмотрел участок в порядке, предусмотренном для остановленного оборудования. Следующая коробка действительно находилась на ленте. Её положение не совпадало с тем, которое программа обычно ожидала в начале нового цикла.

- Если просто нажать «пуск», она пойдёт как новая? - спросил начальник смены.

- Сейчас нельзя на это рассчитывать, - ответил разработчик. - Программа и реальное положение изделия расходятся.

Наладчик добавил, что после остановки следовало проверить и состояние механизмов. Неподвижный робот не обязательно находится в исходной точке обычного цикла; конвейер мог остановиться между предусмотренными положениями. Дальнейшие команды должны опираться на установленное состояние, а не на предположение, что остановка «обнулила» участок.

Судьба незавершённого цикла

Герои отдельно разобрали изделия. Пропущенную коробку нельзя было считать обработанной. Следующую - тоже, пока не выяснено, до какого этапа она дошла. Для каждой требовалось определить дальнейший путь по правилам участка, чтобы ни одна не затерялась среди новых коробок.

- Это дольше, чем убрать одну застрявшую, - заметил начальник смены.
- Да, - сказал оператор. - Зато потом не придётся гадать, какая из них прошла мимо работа.

Инженер попросил разработчика показать, какие действия доступны при восстановлении. На пульте была команда сброса сообщения. Она очищала аварийное состояние, но сама по себе не проверяла положение изделий и механизмов. Слова «сброс» и «готовность к пуску» означали разные вещи, хотя прежде в разговоре их часто употребляли как одно.

Разработчик записал, какие подтверждения система должна получать перед возвратом к обычному циклу. Часть могла прийти от устройств. То, что нельзя было установить по их сигналам, требовало предусмотренной проверки человеком. Если необходимое состояние не подтверждалось, повторный пуск откладывался до выяснения причины.

Почему вчера ответили «пока неизвестно»

Начальник смены вспомнил первый вечер, когда инженер отказался сразу разрешить запуск. Тогда ответ звучал уклончиво. Теперь у неизвестного появились очертания: состояние незавершённого цикла, положение оставшихся коробок, готовность механизмов и судьба изделий, которые могли пройти без обработки.

Расследование не требовало ждать окончательного объяснения каждой потёртости на направляющей. Но оно требовало не возвращать линию к работе, пока существенные для пуска состояния не установлены и предусмотренная реакция на найденное расхождение не проверена. Причину вчерашней задержки продолжали искать; решение о работе участка опиралось на то, что можно было подтвердить сейчас.

Перед уходом инженер попросил показать оператору обновлённый экран восстановления. На нём не было одной большой кнопки, обещавшей «исправить ошибку». Система показывала, какое состояние уже подтверждено, что осталось проверить и почему обычный цикл пока недоступен.

Начальник смены прочитал экран без подсказки разработчика.

- Вот теперь понятно, что именно задерживает пуск.

Посмотрите как инженер. Почему удаление пропущенной коробки и сброс сообщения не доказывают, что линия готова продолжить обычный цикл?

Что осталось за кадром. Описанный разбор показывает, какие вопросы нужно закрыть перед повторным пуском. Конкретный порядок доступа, проверки и восстановления определяется устройством линии и установленными для неё процедурами; текст на экране не заменяет их.

Интерлюдия. Складской робот

На экране учебного примера складской робот стоял посреди прохода. Обычно он определял своё положение по нескольким признакам: данным о движении колёс, наблюдениям за окружающим пространством и карте

склада. Теперь эти сведения расходились. Колёса показывали, что робот проехал заданное расстояние, но видимые ориентиры не совпадали с ожидаемыми.

- Он потерялся? - спросил начальник смены.
- Он не может достаточно надёжно подтвердить, где находится, - ответил разработчик. - Это точнее.

Робот ещё способен выполнить команду на движение. Его приводы исправны. Но маршрут, рассчитанный для неверного положения, может привести его не туда, куда предполагалось. Точность исполнения команды не исправит ошибку в исходных сведениях.

Оператор вспомнил коробку на упаковочной линии.

- Там робот тоже мог двигаться точно, если бы мы дали ему неверное место захвата.

Инженер по надёжности кивнул. В обоих случаях прежде следующего движения требовалось ответить на вопрос о положении. Если подтверждения нет, возможные действия зависят от самого устройства и обстановки: получить новые наблюдения, ограничить движение предусмотренным образом или прекратить его и запросить помощь. Нельзя выбирать вариант только потому, что он быстрее вернёт машину к работе.

У складского робота оставался ещё один вопрос, которого не было на неподвижной схеме: что находится в проходе сейчас? Даже восстановленное представление о собственном положении не описывает автоматически людей и предметы вокруг него.

Начальник смены посмотрел на учебную карту, где значок робота стоял в одной точке.

- Если место не подтверждено, лучше не рисовать значок так уверенно.

Разработчик убрал точку. Вместо неё на карте появилась область, в пределах которой робот мог находиться по имеющимся сведениям. Она была менее удобной на вид, зато не выдавала расчёт за установленный факт.

Посмотрите как инженер. Какое сведение складскому роботу необходимо подтвердить прежде, чем продолжать маршрут, даже если его приводы работают без ошибок?

Что осталось за кадром. Пример поясняет, почему неопределённость положения влияет на выбор действия. Допустимые режимы конкретного мобильного робота зависят от его конструкции, окружающего пространства и предусмотренных мер защиты; их нельзя перенести с упаковочной линии или назначить по одной картинке.

Часть IV. Робот и его неполная картина мира

Глава 16. Точное движение к неверному месту

Наладчик положил контрольную коробку на неподвижный конвейер. Разработчик вывел на экран точку, к которой робот подошёл бы в обычном цикле. На коробке она находилась немного правее середины.

- Если поставить изделие ровно, захват получается здесь, - сказал он.

Наладчик повернул коробку так, как её могло развернуть у направляющей. Точка на экране осталась на прежнем месте. На самой коробке она оказалась ближе к краю.

Оператор посмотрел на работа.

- Он всё равно придёт точно в эту точку?
- Если получит такую команду и движение выполнится штатно, именно к ней и пойдёт, - ответил разработчик.

Точность работа не помогала определить, правильно ли выбрана цель. Она отвечала на другой вопрос: насколько точно механизм выполняет заданное движение.

Откуда берётся цель

Чтобы захватить коробку, программе нужно связать положение изделия с положением работа. Датчик сообщает о прохождении определённой точки. Камера даёт сведения о коробке на снимке. Данные конвейера помогают оценить, куда она переместится к моменту захвата. Из этих сведений система рассчитывает место действия.

Каждый шаг может внести расхождение. Если коробка повернулась после снимка, её положение уже изменилось. Если расчёт предполагает, что изделие движется вместе с лентой без смещения, контакт с направляющей может нарушить это предположение. Робот при этом не «сбивается с траектории». Он выполняет траекторию, построенную для неверно описанного изделия.

Разработчик показал запись штатного цикла. Рассчитанная точка находилась внутри области коробки, где захват был предусмотрен. Затем он вывел расчёт для контрольного перекося. Сама траектория работа не

изменилась, пока не изменились входные сведения. Разница возникала раньше команды на движение.

- Значит, проверка точности робота не поймает такую ошибку, - сказал начальник смены.

- Не поймает сама по себе, - ответил наладчик. - Робот может каждый раз приходить туда, куда ему велели.

Две разные проверки

Герои разделили работу на два вопроса. Первый: достиг ли робот заданного положения в пределах предусмотренных для него условий? Для ответа нужны сведения о движении самого механизма.

Второй: было ли заданное положение подходящим для этой коробки именно в этот момент? Здесь требовались сведения об изделии, времени их получения и том, что могло измениться между наблюдением и захватом.

Оператор предложил наблюдать коробку камерой непосредственно перед движением робота. Разработчик согласился проверить возможность, но не назвал её готовым решением. Новый снимок сократил бы промежуток между наблюдением и действием; он не сделал бы невидимые части коробки видимыми и не отменил бы необходимости проверить, как быстро изделие меняет положение.

Инженер по надёжности указал на вчерашний журнал.

- Мы знаем, что подтверждение пришло поздно. Мы не знаем точного положения коробки в момент, когда мог бы начаться захват. Поэтому не будем рисовать вчерашнюю точку как измеренную.

Разработчик убрал с реконструкции тонкий крестик, обозначавший «место коробки». Вместо него он оставил область возможных положений по имеющимся записям. Она пересекала обычное место захвата лишь частично. Реконструкция стала менее красивой, зато честнее.

Почему вчера робот не двинулся

Начальник смены напомнил, что в происшествии робот вообще не попытался взять коробку. Объяснение о точном движении к неверному месту могло показаться отступлением от расследования.

- Вчера правило не выдало команду на захват, - сказал разработчик. - Мы обсуждаем, нельзя ли исправить пропуск, просто разрешив роботу действовать при позднем сигнале.

Контрольный перекус показал опасность такого исправления. Если заменить осторожное ожидание командой на захват по расчётному положению, пропущенных коробок может стать меньше, но робот начнёт подходить к изделиям, положение которых не подтверждено. Один неверный результат можно было бы обменять на другой.

Герои решили проверять не только долю выполненных захватов. Им нужно было установить, при каких сведениях точку захвата можно считать обоснованной и что происходит, если коробка меняет положение после последнего наблюдения.

Наладчик убрал контрольное изделие с ленты.

- Допустим, точка выбрана верно и робот закрыл захват, - сказал он. - Откуда мы узнаем, что коробка поехала вместе с ним?

Разработчик открыл схему следующего шага. В ней закрытие захвата и успешный перенос были расположены совсем близко друг к другу.

Посмотрите как инженер. Какие разные проверки нужны, чтобы установить, что робот точно выполнил команду и что сама команда была рассчитана для действительного положения коробки?

Что осталось за кадром. Контрольный перекус показывает возможную ошибку выбора цели, но не воспроизводит вчерашнее движение изделия. Герои не утверждают, что робот тогда получил неверную точку захвата: по журналу команда на захват не выдавалась.

Глава 17. После прикосновения мир меняется

На проверочном участке робот подошёл к контрольной коробке и закрыл захват. На экране появилась отметка: «Захват выполнен».

Оператор посмотрел на изделие. Оно осталось на ленте.

Захват сомкнулся рядом с коробкой, не удержав её. Команда была выполнена, а нужного результата не было.

- Вот эту строку я бы переименовал, - сказал оператор.

Разработчик открыл журнал. Отметка появлялась после подтверждения положения механизма захвата. Программа знала, что он закрылся. Она не получала из этого сигнала прямого ответа, находится ли внутри коробка.

Команда ещё не результат

Герои повторили проверку с правильно расположенным изделием. Захват закрылся, робот поднял коробку и перенёс её к месту укладки. Теперь видимый результат совпал с ожидаемым.

В журнале оба случая начинались одинаково: команда на закрытие, подтверждение движения механизма. Различие появлялось позже, когда камера проверяла место, где должна была оказаться перенесённая коробка.

Инженер по надёжности нарисовал на доске короткую последовательность: «Закрыть захват → проверить удержание → перенести → проверить размещение». Между соседними шагами он оставил промежутки.

- Что может измениться здесь? - спросил он, указывая на первый.

Наладчик ответил: коробка может выскользнуть или деформироваться. При переносе она может сместиться в захвате. После отпускания - оказаться не там, где рассчитывали. Подтверждение одного шага не удостоверяло автоматически следующий.

Оператор вспомнил вчерашнее происшествие.

- Тогда робот не брал коробку. Но если бы взял, одного сигнала о закрытии всё равно было бы мало.
- Да, - сказал разработчик. - Мы должны различать действие механизма и состояние изделия после этого действия.

Чем подтвердить захват

Наладчик показал место установки захвата. В зависимости от его устройства удержание можно оценивать по дополнительным сигналам или

наблюдению за изделием. Но любой признак имеет пределы. Положение губок сообщает, насколько они сомкнулись, а не обязательно то, что коробка удерживается надёжно. Снимок может показать коробку в нужной области, но не гарантировать, что она не изменит положение при следующем движении.

Для испытания разработчик подготовил разные контрольные случаи: коробка стоит верно, смещена, имеет примятый край. Проверки проводили в предусмотренном режиме, чтобы наблюдать исход, не выдавая пробные движения за разрешение работать так с продукцией. Задача состояла не в том, чтобы найти один «волшебный» сигнал. Герои выясняли, какое подтверждение нужно перед переносом и какое - после него.

Начальник смены сравнил две отметки в журнале: «Захват выполнен» и «Коробка размещена».

- Если обе могут появиться без проверки самого изделия, оператор прочитает их как факт.

Разработчик согласился изменить названия. Для первой отметки подошло «Захват закрылся». Итоговая запись должна была появляться лишь после той проверки, которая действительно подтверждает размещение коробки. Если подтверждения нет, система обязана сообщить именно об этом, а не объявить операцию успешной по одному выполненному движению.

Когда проверка приходит поздно

Инженер вернулся к последовательности на доске. Проверка размещения в конце полезна, но она может обнаружить отсутствие коробки

уже после переноса. Если изделие не удерживалось с самого начала, часть движения прошла при неверном предположении.

- Поэтому некоторые результаты нужно подтверждать раньше следующего действия, - сказал он. - Не всё можно оставить до итогового кадра.

Это не означало проверять каждую мелочь отдельной остановкой. Означало понимать, какое предположение необходимо для следующего движения и чем оно подтверждено. Робот может точно подойти к изделию и точно закрыть захват. После прикосновения положение коробки всё равно способно измениться, а значит, расчёт до контакта уже не описывает происходящее полностью.

На последнем контрольном проходе коробка слегка повернулась при закрытии захвата, но осталась удержанной. Камера увидела её в непривычном положении. Программа выдала низкое качество распознавания, хотя оператор ясно различал изделие на изображении.

Разработчик сохранил кадр.

- Для нас это обычная коробка под необычным углом. Посмотрим, что о ней скажет новая версия распознавания.

Посмотрите как инженер. Какое состояние подтверждает сигнал о закрытии захвата и какой отдельный факт требуется установить перед переносом коробки?

Что осталось за кадром. Подходящие признаки удержания и размещения зависят от конструкции захвата, свойств изделия и доступных

средств контроля. Контрольные проходы показывают границы текущих сообщений, но не задают универсальную схему проверки для любого робота.

Глава 18. Умная камера тоже может ошибаться

Разработчик вывел сохранённый кадр. Коробка была видна, хотя после закрытия захвата немного повернулась. Оператор без труда указал её край. Новая программа распознавания отметила другую область - часть направляющей рядом с изделием.

- Она считает направляющую коробкой? - спросил наладчик.
- На этом кадре выбрала неверную область, - ответил разработчик. - Нужно проверить, почему.

Программу испытывали как способ лучше определять положение изделий. В обычных проходах она справлялась с коробками разного размера и небольшим поворотом. Именно поэтому ошибка на контрольном кадре заслуживала внимания: изображение не было пустым или полностью закрытым, а ответ выглядел достаточно определённым, чтобы его можно было принять за готовое измерение.

Что распознавание делает хорошо

Прежние правила камеры опирались на заранее заданные признаки изображения. Когда коробка меняла вид из-за поворота или неровного края, подобрать правило для всех случаев было трудно. Обучаемый алгоритм позволял находить изделие по примерам изображений, не описывая вручную каждую возможную форму его контура.

На привычных кадрах это давало хороший результат. Камера находила коробку там, где простое правило иногда теряло её край. Но результат распознавания оставался сведением от одного источника. Он зависел от того, что попало в кадр, как была освещена сцена и насколько похожи новые изображения на те, на которых алгоритм проверяли.

Разработчик открыл снимки из двух смен. На одной коробки освещались ровно. На другой на их боковой стороне лежала заметная тень: после обслуживания светильник направили немного иначе. Изделия не исчезли и не изменили своего назначения, но изображение, которое получала программа, стало другим.

Наладчик принёс коробку с обновлённой печатью на боковой стороне. Крупная тёмная полоса проходила близко к тому месту, где на старой упаковке был ровный светлый участок.

- Такие начали приходить недавно, - сказал он.

Теперь у героев были два возможных изменения условий съёмки. Ни одно не следовало объявлять причиной ошибки без испытания. Но оба нужно было включить в проверку, прежде чем доверять новой программе на работающей линии.

Уверенный ответ на незнакомом кадре

Разработчик запустил распознавание на подборке сохранённых изображений. На большинстве кадров коробка была найдена. На нескольких с новым рисунком упаковки граница определялась неверно. Один снимок с изменённым освещением дал особенно неприятный результат: программа

уверенно указала положение, которое не совпадало с видимым краем изделия.

Начальник смены вспомнил число из предыдущего разговора.

- Опять высокий показатель?

- Да. Поэтому я не хочу выводить его на экран как обещание правильного захвата.

Алгоритм выбрал ответ по признакам изображения. Высокий показатель означал, что по его внутренним правилам выбор получился определённым. Он не доказывал, что эти правила остаются верными после изменения света и упаковки.

Оператор предложил простой выход: при сомнительном изображении передавать решение человеку. Инженер по надёжности указал на ошибочный кадр.

- Этот ответ не выглядит сомнительным для самой программы. Если передавать человеку только то, в чём она признаётся неуверенной, такой случай пройдёт мимо.

Герои вернулись к сопоставлению источников. Камера могла ошибочно указать положение, но датчик и сведения о движении изделия давали независимые основания проверить, согласуется ли её ответ с происходящим. Они не обещали, что сравнение всегда обнаружит ошибку: несколько источников тоже могут разделять одно неверное предположение. Однако выдавать ответ камеры сразу за подтверждённое положение стало нельзя.

Испытывать нужно условия, а не название технологии

Разработчик составил набор контрольных кадров и проходов с разным положением коробки, новой упаковкой и доступными вариантами освещения. Он отдельно отметил изображения, на которых часть изделия закрыта. Для каждого случая требовалось смотреть не только на среднюю долю правильных ответов, но и на последствия конкретной ошибки: неверная область захвата важнее, чем небольшое смещение рамки на снимке, не влияющее на решение.

Начальник смены спросил, можно ли после этих испытаний поставить распознавание на линию.

- Сначала выясним, какие случаи в набор не попали, - ответил инженер.

- И как система поведёт себя, когда встретит такой случай в работе.

На экране всё ещё был кадр с повернувшейся коробкой. Для человека он выглядел необычным, но понятным. Для программы он оказался поводом указать не туда. Героям предстояло узнать, какие примеры сформировали её представление о «типичной коробке» и сколько привычек участка в этих примерах отсутствовало.

Посмотрите как инженер. Почему высокий показатель распознавания на кадре с новым рисунком упаковки не доказывает, что найденное положение коробки подходит для захвата?

Что осталось за кадром. Проверка на нескольких сохранённых кадрах выявляет возможную ошибку, но не даёт полной оценки работы камеры. Решение о применении распознавания потребует испытаний на условиях и

изделиях, с которыми система встретится на участке, а также проверки её реакции на неподтверждённый ответ.

Глава 19. Чему машину не научили данные

Разработчик открыл папку с изображениями, на которых настраивали распознавание коробок. Оператор быстро пролистал первые десятки кадров.

- Они все стоят ровно.
- Не все, - сказал разработчик и нашёл несколько снимков с небольшим поворотом.

Наладчик указал на изображение направляющей. Она была чистой. Коробки на снимках имели прежний рисунок упаковки, а свет падал почти одинаково. Программа хорошо отвечала на вопросы, которые ей задавали этими примерами. Теперь линия предъявляла ей несколько иные изображения.

Хорошая оценка на узком материале

При проверке разработчик отделял часть снимков от тех, на которых настраивал алгоритм. Результаты выглядели убедительно: коробки на проверочных кадрах находились правильно. Но обе группы изображений были собраны на той же линии в похожих условиях.

- Получается, мы проверили её на незнакомых фотографиях знакомой обстановки, - сказал инженер по надёжности.

Это была настоящая проверка, только её вывод имел границы. Она показывала, как алгоритм работает с изображениями, похожими на собранные ранее. Она не отвечала на вопрос, что произойдёт при другой

упаковке, тени от переставленного светильника или коробке, которую частично закрывает направляющая.

Разработчик не стал защищать общую оценку одним числом. Он разложил снимки по условиям. Для привычной упаковки и освещения результаты оставались хорошими. В группе с новым рисунком встречались неверно определённые края. На снимках с частично закрытой коробкой иногда было невозможно установить положение, необходимое для захвата, даже человеку без дополнительного наблюдения.

Последние случаи нельзя было «исправить данными» в простом смысле. Если важная часть изделия не видна, алгоритму неоткуда получить её действительное положение по одному снимку. Он может предположить его по знакомым примерам, но такое предположение не становится измерением.

Почему примеров может быть много, а нужных - мало

Начальник смены посмотрел на число файлов в папке.

- Снимков ведь собрали немало. Как мы пропустили новую упаковку?
- Когда собирали их, её ещё не было на линии, - ответил разработчик.

Количество изображений не гарантировало разнообразия условий. Сотни кадров одинаково освещённых коробок могли мало рассказать программе о тёмной полосе на новой упаковке. Небольшая группа перекошенных изделий помогала только в пределах тех положений, которые в неё попали.

Инженер попросил отметить, когда именно изменились освещение и упаковка. Герои сопоставили эти даты с записями ошибок распознавания. После изменений необычные ответы встречались чаще, но доступных случаев было мало для уверенного вывода о каждом условии по отдельности. Свет и рисунок упаковки менялись близко по времени; их влияние ещё предстояло разделить контролируемой проверкой.

- Значит, нельзя написать «виноват свет»? - спросил оператор.
- Пока нельзя, - ответил инженер. - Зато можно назвать условия, в которых старой оценке качества уже недостаточно доверять.

Что делать с незнакомым случаем

Разработчик предложил дополнить материал новыми снимками и снова проверить алгоритм. Наладчик поддержал испытания на коробках с разным перекосом. Оператор добавил просьбу: в проверку должны попасть не только снимки, где изделие нашлось, но и случаи, когда система обязана воздержаться от команды на захват.

Это меняло цель испытания. Недостаточно добиться высокой средней точности распознавания. Нужно было узнать, какие ошибки приводят к неверной точке захвата, замечает ли система противоречие с другими сведениями и что делает, если положение определить нельзя. Улучшение ответа на новой упаковке не имело бы смысла, если при следующем изменении условий камера снова выдавала уверенный, но неподтверждённый вывод.

Инженер попросил сохранить версии набора изображений и программы, с которой получены результаты. Иначе спустя несколько

месяцев никто не смог бы точно ответить, какие коробки система уже «видела» при проверке, а какие появились позже.

Перед закрытием папки разработчик вывел два снимка рядом. На одном коробка старого образца стояла ровно. На другом новая упаковка почти сливалась с тенью направляющей. Оба файла назывались одинаково просто: «коробка». Для человека это было название изделия. Для алгоритма между снимками лежало изменение условий, о котором ему никто не сообщил.

Героям оставалось испытать не только обновлённое распознавание, но и всю цепочку после его ответа. Если камера ошибётся снова, система должна заметить, что её сведения не согласуются с другими наблюдениями, и не превращать неуверенное положение в точную команду роботу.

Посмотрите как инженер. Почему высокая доля правильных ответов на отложенных снимках из той же смены ещё не показывает, как распознавание справится с новой упаковкой?

Что осталось за кадром. Найденные ошибки не доказывают, что обучаемое распознавание непригодно для линии. Они показывают предел проведённой проверки. Работу после изменений нужно оценивать на подходящих изделиях и условиях, отдельно наблюдая случаи, в которых неверный ответ влияет на действие робота.

Интерлюдия. Бытовая техника

После разбора снимков оператор задержался у экрана.

- Забавно. Дома стиральная машина тоже однажды сообщила, что дверца не закрыта. Я её закрыл ещё раз, а сообщение осталось.

Разработчик нарисовал две короткие строки: «Дверца закрыта» и «Система получила подтверждение закрытия».

Они часто совпадают, поэтому в обычном разговоре звучат как одно утверждение. Но для машины это разные вещи. Дверца может выглядеть закрытой, а нужное подтверждение не поступить. Возможен и другой вопрос: достаточно ли одного сигнала, чтобы сделать вывод о состоянии механизма при любых условиях?

Инженер по надёжности вернулся к упаковочной линии. Оператор видел коробку у направляющей; программа получила сигнал датчика. Дальше надо было выяснять, что именно этот сигнал подтверждал и позволял ли он системе считать положение изделия известным. Ошибка могла возникнуть не потому, что устройство «ничего не видело», а потому, что ограниченному сообщению придали слишком широкий смысл.

- Но стиральную машину я просто не смог запустить, - заметил оператор. - На линии последствия могут быть другими.

- Поэтому пример помогает задать вопрос, а не выбрать реакцию, - ответил инженер.

У бытового прибора и промышленного робота разные задачи, устройство и условия работы. Общим остаётся расстояние между состоянием предмета, сигналом о нём и выводом программы. Как поступать при расхождении, каждый раз приходится определять для конкретной машины.

Посмотрите как инженер. Если система сообщает, что дверца не закрыта, почему из одного сообщения нельзя сразу заключить, что человек оставил её открытой?

Что осталось за кадром. Этот пример не описывает диагностику конкретной стиральной машины и не предлагает способ обхода её блокировок. Он только показывает, почему сообщение системы следует отличать от установленной причины.

Часть V. От расследования к доверию

Глава 20. Испытание, в котором всё пошло не по плану

После изменений линия несколько раз обработала обычные коробки без остановок. Робот подходил к изделиям, захват закрывался, камера подтверждала результат. Начальник смены посмотрел на записи.

- Нормальный цикл работает. Когда возвращаем участок в работу?

Инженер по надёжности показал подготовленную программу испытаний.

- После того как проверим не только нормальный цикл.

Вчерашняя линия тоже долго работала нормально. Проблема обнаружилась, когда сигналы перестали складываться в привычную последовательность. Поэтому теперь герои хотели увидеть, что система делает при отсутствии коробки, задержке подтверждения, противоречии между источниками и временной потере связи. Проверки проводили в подготовленных условиях, с согласованным состоянием участка и без людей в зоне движения оборудования.

Когда коробка не пришла

Первым испытали ожидание изделия. На входе появился сигнал начала цикла, но следующего подтверждения не было. Программа дождалась установленного предела и не дала роботу команду на захват.

На экране появилось сообщение: «Положение изделия не подтверждено. Захват не выполнялся». Ниже указывалось, какие движения участка были прекращены и что требовало проверки.

Оператор прочитал текст и спросил:

- Откуда система знает, что коробки нет? Может, она остановилась до датчика.

Разработчик поправил запись в протоколе испытания. Система знала только то, что ожидаемое подтверждение не поступило. Сообщение на экране было точнее первоначального названия теста «Нет коробки». Даже в искусственно созданном случае, где испытатели знали причину, программе нельзя было приписывать знание, которого ей не дали сигналы.

Когда подтверждение опоздало

Для следующей проверки герои создали предусмотренным испытательным способом задержку подтверждения. Сигнал появился уже после того, как время ожидания истекло.

Робот остался неподвижным. Система сохранила поздний сигнал в журнале, но не вернулась молча к прежнему шагу и не превратила его в разрешение на захват. На экране было видно, что подтверждение пришло

после принятого решения и что состояние изделия перед продолжением работы требуется установить.

Начальник смены попросил повторить испытание с сигналом, который приходит незадолго до предела. Он хотел убедиться, что изменение настройки не вызовет постоянных остановок на обычных коробках. Несколько проходов закончились штатно. Один потребовал дополнительной проверки, хотя коробка на глаз выглядела правильно.

- Это ложная тревога? - спросил начальник смены.

Инженер не согласился с таким названием без уточнения. Для человека коробка в конце прохода стояла нормально. Но в момент решения система не получила нужного подтверждения. Требовалось выяснить, насколько часто такая ситуация возникает в работе и можно ли получать сведения вовремя, не ослабляя необходимую проверку.

Когда источники не согласны

Затем испытали противоречие между камерой и расчётным положением изделия. Новая логика не объявила одно устройство неисправным. Она зафиксировала, что положение, нужное для захвата, не подтверждено согласованными сведениями, и не разрешила движение робота к спорной точке.

Оператор заметил, что сообщение стало длиннее.

- Здесь мне достаточно знать, что захват не выполнялся и где проверять коробку. Спор камеры с расчётом пусть остаётся в подробностях.

Разработчик разделил экранную подсказку и диагностическую запись. Для разбора важно было сохранить, какие именно сведения не совпали. В момент остановки оператору прежде всего требовалась ясная картина выполненных действий и текущего состояния участка.

Последним проверили временное отсутствие одного из сообщений, которое система обычно получает по связи. Герои не пытались угадать, что произошло с устройством на другом конце. Они проверяли, не примет ли программа старое сообщение за новое подтверждение. В испытанном случае этого не произошло: действие, требовавшее свежих сведений, осталось недоступным.

Что испытание действительно доказывает

К концу дня у героев был протокол с удачными и неудачными результатами. В одном из ранних проходов сообщение оператору неверно называло коробку отсутствующей; текст исправили. В остальных проверенных случаях робот не начинал захват без необходимых подтверждений. Но инженер не позволил записать итог как «система теперь безопасна при любых сбоях».

Они испытали определённые отклонения в подготовленных условиях. Это давало основание судить о реакции именно на них. Другие сочетания неисправностей, изделий и условий работы оставались отдельной задачей для проверки.

Начальник смены подписал протокол испытаний и задержался у последней строки.

- Вчера мы спрашивали, работает ли линия. Сегодня спрашиваем, что она делает, когда не может работать как обычно.

Это был более трудный вопрос. Зато теперь на него отвечали наблюдениями за поведением системы, а не обещанием, что после настройки ошибочных коробок больше не будет.

Посмотрите как инженер. Почему успешная обработка нескольких обычных коробок не проверяет реакцию системы на запоздавшее или противоречивое подтверждение?

Что осталось за кадром. Описанные испытания проверяют выбранные сценарии и сообщения системы. Они не охватывают все возможные сочетания отказов и не заменяют предусмотренную для конкретного оборудования процедуру подтверждения его готовности к эксплуатации.

Глава 21. Почему не стоит искать одного виноватого

После испытаний инженер по надёжности собрал героев у доски. На ней ещё оставалась первая запись оператора: «Робот пропустил коробку». Рядом он прикрепил снимок направляющей, выдержку из журнала сигналов и старый текст сообщения о нарушении цикла.

- Если писать причину для отчёта одним предложением, что поставим?

- спросил начальник смены.

- Поздний сигнал, - ответил разработчик.

- Коробку повело у направляющей, - сказал наладчик.

Оператор указал на экранный снимок:

- А я узнал о проблеме, когда она уже прошла мимо робота.

Все три ответа относились к происшествию. Ни один не описывал его целиком.

Что установлено, а что остаётся версией

Инженер начал с того, что записи подтверждали достаточно надёжно. Сигнал у направляющей в проблемном цикле пришёл с меньшим запасом времени, чем обычно. К моменту перехода программы нужное подтверждение не было доступно для решения о захвате. Робот не получил команду брать эту коробку. Позже система сообщила о нарушении цикла, а оператор обнаружил изделие за местом обработки и остановил участок.

Дальше начинались менее определённые сведения. Контрольные проходы показали, что перекося может изменить движение коробки и момент появления сигнала. След на направляющей и наблюдение оператора согласовывались с такой версией. Но записи не позволяли восстановить вчерашнее положение изделия непрерывно, поэтому назвать перекося доказанной единственной причиной задержки было нельзя.

Увеличивавшееся время движения привода тоже требовало проверки. Оно могло уменьшать запас времени в части циклов, но прямую связь с пропущенной коробкой герои ещё не установили. Инженер разделил на доске подтверждённые события, вероятные объяснения и вопросы, оставшиеся открытыми.

- В отчёте так и напишем? - спросил начальник смены.
- Иначе мы превратим удобную версию в факт, - ответил инженер.

Почему одного исправления было мало

Даже если коробка действительно повернулась у направляющей, остановка возникла не от одного поворота. На линии и раньше проходили изделия с небольшим перекосом. Важным стало то, что подтверждение запоздало, правило перехода допускало дальнейшую работу без установленного положения этой коробки, а сообщение оператору не объясняло, какое состояние участка осталось неопределённым.

Разработчик показал изменения программы. Теперь поздний сигнал не разрешал захват задним числом, а противоречие между источниками не превращалось в точную координату изделия. Наладчик оформил проверку направляющей и привода. Оператор участвовал в проверке нового сообщения на пульте.

Это были меры для разных звеньев одной цепочки. Исправить положение направляющей было полезно, но оно не делало прежнее правило перехода хорошим. Изменить программу было необходимо, но оно не устраняло возможное изменение механики. Переписать сообщение помогало человеку после тревоги, однако не заменяло своевременного обнаружения расхождения.

Начальник смены посмотрел на перечень выполненных работ.

- Значит, в отчёте не будет фамилии в графе «кто допустил»?
- Мы можем указать, кто отвечает за каждую проверку и изменение, - сказал инженер. - Это не то же самое, что назначить одного человека причиной всего события.

Ошибка без удобного злодея

Разработчик не предусмотрел вчерашнее сочетание задержки и положения изделия. Наладчик не имел измерений, которые позволили бы заранее связать редкие перекосы с изменением времени прохода. Оператор увидел пропущенную коробку и остановил участок, но не мог по виду линии узнать, какой переход выполнила программа. Начальник смены должен был решать вопрос о простое ещё до окончания разбора.

У каждого была часть сведений и своя обязанность. Поиск единственного виновного закрыл бы обсуждение раньше, чем они проверили бы слабые места участка. Отказ от такого поиска не означал отказа от ответственности: каждое принятое изменение теперь имело исполнителя, срок проверки и наблюдаемый результат.

Инженер вернулся к первой записи на доске. «Робот пропустил коробку» оставалось верным описанием того, что увидел оператор. Оно больше не казалось объяснением.

Когда совещание закончилось, наладчик задержался у журнала обслуживания.

- Я ведь отмечал, что коробки иногда идут боком. И оператор писал об этом. Почему мы начали сопоставлять записи только после остановки?

Инженер не стал указывать на одну пропущенную строку. Этот вопрос касался не отдельного датчика и не одного человека, а того, как участок обращался с предупреждениями до происшествия.

Посмотрите как инженер. Почему устранение возможного перекоса коробки не делает ненужной проверку правила, по которому система продолжала цикл без подтверждённого положения изделия?

Что осталось за кадром. Расследование установило последовательность записанных событий и выявило несколько условий, усиливших последствия отклонения. Точная причина запаздывания вчерашней коробки остаётся не полностью доказанной; отчёт должен сохранить эту границу вывода.

Глава 22. Организация, которая не замечает предупреждений

Наладчик принёс журнал обслуживания, оператор - сменные записи, начальник смены - перечень остановок. На столе лежали три истории одной линии. В каждой были свои даты и свои формулировки, но ни одна не связывала сведения из остальных.

В журнале обслуживания говорилось о проверке направляющей после снятия узла. В сменной записи - о коробках, которые иногда шли боком. В перечне остановок - о нескольких тревогах без найденной неисправности. Отдельно каждая запись выглядела рядовой. Вместе они указывали, что поведение участка стоило проверить внимательнее.

- Почему никто не собрал это раньше? - спросил начальник смены.

Вопрос не требовал ответа «потому что один человек не прочитал журнал». Наладчик читал записи об обслуживании. Операторы передавали замечания следующей смене. Разработчик видел ошибки в программе. Но сведения хранились так, что сопоставить их можно было лишь после специального запроса.

Когда запись есть, а наблюдения нет

Инженер по надёжности предложил найти все случаи, когда время прохождения коробки приближалось к пределу ожидания. Записи контроллера существовали, но обычный отчёт показывал только остановки. Успешные циклы с уменьшающимся запасом времени в него не попадали.

- Мы сохраняли данные и всё равно не видели изменения, - заметил разработчик.

- Сохранять мало, - ответил инженер. - Нужно заранее понимать, при каком изменении их проверять и кто этим занимается.

Начальник смены напомнил о другой стороне вопроса. Если выводить оператору каждое колебание времени, экран быстро заполнится предупреждениями. Герои уже видели, что лишние тревоги не делают решение лучше. Поэтому повторяющиеся задержки предложили рассматривать при плановой проверке участка, а отсутствие обязательного подтверждения перед движением - показывать сразу. Это были разные задачи для разных моментов работы.

Наладчик попросил добавить к проверке данные об обслуживании. Если после регулировки меняется время прохождения изделий, сравнение с прежними циклами помогает заметить результат, которого не видно на контрольной коробке. Разработчик указал, что сравнивать следует одинаковые режимы работы, иначе график может показать разницу условий, а не изменение узла.

«Обычная мелочь»

Оператор нашёл в сменном журнале фразу: «Снова краткая тревога, после сброса всё работает». Тогда она казалась понятной: участок продолжил выпуск, неисправность не повторилась. После вчерашнего происшествия слово «снова» потребовало расшифровки.

Какая тревога? На каком шаге? Что успела сделать система до сброса? Проверяли ли изделие, которое находилось в зоне? Без этих сведений повторение оставалось впечатлением, которое нельзя было сопоставить с журналом контроллера.

Начальник смены признал, что при плотном графике работы короткая остановка выглядела как задача, которую нужно быстрее закрыть. Но он возразил против идеи разбирать каждую из них полным составом.

- Если на каждую мелочь собирать совещание, времени на работу не останется.

Инженер предложил иной порядок. Повторяющиеся события должны сохранять достаточные сведения для последующего сравнения. Если их характер меняется или они приближаются к условию, при котором система не может подтвердить безопасный следующий шаг, разбор становится срочным. Так люди не обязаны расследовать каждое колебание на месте, но и не теряют ранние признаки среди одинаковых фраз «после сброса работает».

У предупреждения должен быть адресат

Герои распределили не вину, а работу со сведениями. Оператор мог коротко фиксировать видимое отклонение и его время. Наладчик - сопоставлять повторения с состоянием механики и обслуживанием.

Разработчик - проверять, что именно означали сигналы и переходы программы. Инженер по надёжности - смотреть, складываются ли отдельные записи в повторяющуюся цепочку событий. Начальнику смены требовалась ясная информация о том, какие проверки нужно выполнить до продолжения работы и какие можно запланировать без немедленной остановки.

Ни одна роль не давала полной картины сама по себе. Зато теперь у замеченной задержки или неясной тревоги появлялся путь от наблюдения к проверке. Если проверка ничего не подтверждала, её результат тоже сохраняли, чтобы при следующем случае не начинать всё заново.

В конце совещания начальник смены задержался у списка изменений. Новое правило регистрации выглядело скромнее модернизации камеры или программы. Но без него будущие предупреждения могли снова остаться в разных журналах, пока очередная коробка не прошла мимо робота.

Он поставил рядом с правилом срок первой проверки: после нескольких смен работы линии герои должны были посмотреть не только на число остановок, но и на задержанные успешные циклы, понятность сообщений и результаты осмотра механики.

Посмотрите как инженер. Какие две записи из разных журналов следовало сопоставить до происшествия, чтобы получить основание для проверки направляющей, не объявляя её неисправной заранее?

Что осталось за кадром. Новый порядок помогает не терять повторяющиеся признаки, но не доказывает, что любой будущий сбой будет

замечен заранее. Польза зависит от качества записей, регулярности проверки и того, принимаются ли решения по найденным расхождениям.

Глава 23. Кто отвечает за решение машины

Перед повторным вводом линии в работу начальник смены принёс лист согласования изменений. На нём были поля для подписей, но рядом с новым правилом остановки при неподтверждённом положении коробки оставалось пустое место.

- Кто решил, что теперь линия должна вести себя именно так? - спросил он.

Разработчик поднял руку:

- Правило написал я.
- Я спрашиваю не только кто написал программу.

На испытаниях система не давала роботу команду на захват, когда положение изделия не было подтверждено. Это выглядело разумно. Но правило затрагивало и другое: как прекращалась подача коробок, что показывал экран, кто проверял участок и на каких основаниях разрешался повторный пуск. Один человек не мог подтвердить всё это своей подписью под программным кодом.

Кто задаёт задачу

Разработчик открыл исходное описание цикла. Там было указано, как обнаружить коробку, выполнить захват и проверить результат. Случай позднего подтверждения описывался короче: «Перейти к следующему шагу после истечения времени ожидания».

- Я реализовал этот переход, - сказал он. - Но документ не отвечал, где в этот момент может находиться изделие.

Тот, кто задаёт системе задачу, определяет не только желаемый результат - обработать коробку. Ему приходится определить и границы: при каких сведениях действие допустимо, что считать завершённой операцией и как поступать, когда ожидаемых сведений нет. Программа может безошибочно выполнять плохо сформулированное правило.

Начальник смены посмотрел на старую строку. Она объясняла поведение системы, но не доказывала, что такое поведение было подходящим для реального участка.

Кто подтверждает условия

Наладчик отвечал за проверку механики после изменений. Ему нужно было установить, как коробки проходят у направляющей, как ведёт себя привод и сохраняются ли проверенные положения узлов. Он не мог по этим наблюдениям в одиночку заключить, что программа правильно обработает поздний сигнал.

Разработчик мог подтвердить, какой переход выполняет контроллер и какие действия блокирует новое правило. Он не мог по коду удостовериться, что оператор увидит на экране достаточно сведений для осмотра участка. Это проверяли вместе с людьми, которые работают у линии.

Инженер по надёжности сводил результаты испытаний и открытые вопросы. Он предложил оставить в листе согласования не общую фразу «система проверена», а указание на конкретные условия, в которых её проверяли: обычные коробки, позднее подтверждение, противоречие

источников и потерю сообщения. Тогда подписи относились бы к наблюдаемому поведению, а не к неопределённому обещанию надёжности.

- А если появится новая упаковка? - спросил оператор.
- Проверим, входит ли она в условия, для которых подтверждена работа камеры, - ответил инженер. - Если нет, прежняя запись об испытаниях сама по себе ничего о ней не доказывает.

Ответственность не заканчивается при запуске

Начальник смены провёл черту под результатами испытаний.

- Допустим, сегодня мы согласовали ввод в работу. Через месяц наладчик видит новую задержку. Чьё это дело?
- Того, кто заметил, - начал оператор и остановился. - Но ему нужно знать, куда передать наблюдение.

Герои вернулись к порядку, который разработали после происшествия. Внедрение не заканчивалось первым успешным пуском. Во время эксплуатации могли измениться коробки, положение направляющей, освещение и привычное время движения. Условия, на которых держалось принятое решение, следовало проверять и при необходимости пересматривать.

Разговор не свёлся к поиску одной должности, отвечающей «за машину». Разные люди принимали разные решения: что должна делать система, как это реализовать, в каких условиях испытать, когда вводить в работу и что проверять после изменений на участке. В иной отрасли или на другом оборудовании порядок согласования мог бы быть другим. Для этой

линии важнее было не угадать универсальное название ответственного, а сохранить основания каждого решения.

Начальник смены дополнил лист вопросом: «На основании каких проверок разрешено это поведение?» Рядом появились ссылки на результаты испытаний и перечень ещё не проверенных условий. Пустое место не исчезло полностью, но стало видно, какой ответ в нём требуется.

- Теперь я понимаю, что подписываю, - сказал он. - Не обещание, что линия никогда не ошибётся. Разрешение работать в условиях, которые мы проверили, с определённой реакцией на то, чего она не знает.

До повторного ввода оставался последний разговор: достаточно ли этих оснований, чтобы доверить линии работу?

Посмотрите как инженер. Почему подпись разработчика под новым правилом программы сама по себе не подтверждает, что весь участок готов работать с этим правилом?

Что осталось за кадром. Эта глава не устанавливает юридическую ответственность и не заменяет требования, действующие для конкретного оборудования и места его применения. Она показывает, какие решения и основания стоит сделать явными при проектировании, внедрении и эксплуатации системы.

Глава 24. Сколько доверия достаточно?

Линия была готова к проверочному пуску. На конвейере стояли контрольные коробки, у пульта собрались оператор, наладчик, разработчик, инженер по надёжности и начальник смены. На этот раз никто не спрашивал,

«исправили ли работа». Его движение с начала расследования не было главным подозреваемым.

Начальник смены положил на стол протокол испытаний.

- Можем доверить линии работу?

Разработчик посмотрел на экран, где были видны сигналы участка.

• С обычными коробками в проверенных условиях она выполняет цикл. При позднем подтверждении не начинается захват.

• Это ответ на часть вопроса, - сказал инженер. - Осталось назвать пределы.

Для какой задачи

Оператор первым уточнил задачу. Линия должна не просто перемещать коробки. Она должна обнаружить изделие, определить подходящее положение для захвата, перенести его и подтвердить результат. Если какой-то шаг не установлен, запись «цикл завершён» не должна скрывать пробел.

После испытаний сообщения на пульте различали выполненную команду и подтверждённый результат. Оператор мог увидеть, что захват закрылся, но удержание коробки ещё не подтверждено. Это не делало машину безошибочной. Зато она перестала выдавать одно событие за другое.

При каких условиях

Наладчик перечислил то, что проверили на участке: положение направляющей, движение подающего узла, проход контрольных коробок. Разработчик добавил условия работы камеры и упаковку, с которой распознавание испытывали. Инженер попросил оставить в протоколе и границу: новая форма изделия, заметное изменение освещения или механики потребуют проверки того, применимы ли прежние результаты.

- Если для каждой новой коробки надо собирать всех нас, линия не заработает, - заметил начальник смены.

- Не для каждой, - ответил наладчик. - Нужно знать, какие изменения существенны для того, как она обнаруживает и берёт изделие.

Этот ответ не превращался в список всех возможных коробок на годы вперёд. Он устанавливал, что доверие к линии относится к определённым условиям работы, а не переносится автоматически на любое изменение цеха.

С каким контролем

Инженер вернулся к карте расследования: сигнал, интерпретация, решение, действие, проверка результата. Теперь возле каждого звена были вопросы, которые герои умели задавать.

Пришёл ли нужный сигнал вовремя? Что именно он подтверждает? Не расходится ли рассчитанное положение с наблюдением камеры? Разрешено ли движение при доступных сведениях? Подтверждён ли результат после действия?

Карта не гарантировала, что на каждый вопрос всегда найдётся ответ. Её польза была в другом: неизвестное не пряталось внутри общей отметки

«исправно». Если подтверждения не хватало, система должна была обозначить, на каком шаге остановилось знание о коробке, и выполнить предусмотренную для этого реакцию.

Что происходит при неопределённости

Начальник смены попросил показать один из испытанных сценариев без движения линии. Разработчик открыл сохранённую запись с запоздавшим сигналом. В ней было видно, что робот не получил разрешения на захват, подача новых изделий прекратилась предусмотренным образом, а экран сообщил оператору о неподтверждённом положении коробки и выполненных действиях.

- Вот этому я могу доверять, - сказал оператор. - Не тому, что коробка всегда придёт вовремя, а тому, что я узнаю, когда она не пришла так, как ожидалось.

Инженер поправил его лишь в одном: система обнаруживала не всякое возможное отклонение, а те расхождения, которые они определили и проверили. Если условия изменятся за пределами испытанного, поведение придётся оценивать снова. Оператор кивнул. Ему не требовалось обещание всеведения; ему требовалось, чтобы пульт не изображал уверенность там, где подтверждений нет.

Начальник смены подписал решение о проверочном вводе участка в работу с предусмотренным наблюдением за первыми сменами. Наладчик должен был следить за временем движения узла, разработчик - за записями противоречий между источниками, оператор - сообщать о видимых изменениях прохода коробок. Это были не заменители испытаний, а способ заметить, если реальные условия начнут отличаться от проверенных.

Перед пуском инженер снова прочитал три вопроса, которые когда-то записал на полях первого журнала:

Что система знает?

Чего она не знает?

Что она делает, когда не знает?

Теперь для вчерашнего случая ответы существовали. Часть подтверждалась испытаниями. Часть оставалась обозначенным пробелом, за которым следовало наблюдать. Этого было достаточно не для заявления «машине можно верить всегда», а для более строгого решения: поручить ей конкретную работу в проверенных условиях и знать, как она должна поступить, когда оснований продолжать обычный цикл не хватает.

Оператор дал разрешение на проверочный проход. Первая коробка подошла к направляющей ровно. Робот снял её с ленты. Камера подтвердила размещение.

За ней шла следующая.

Посмотрите как инженер. Какие условия нужно назвать вместе с фразой «линия работает надёжно», чтобы она стала проверяемым утверждением, а не общей оценкой?

Что осталось за кадром. Испытания подтвердили поведение линии в выбранных условиях и сценариях. Они не доказывают отсутствие всех будущих ошибок. Доверие к системе придётся пересматривать, если изменятся изделия, оборудование, программа или условия наблюдения.

Эпилог. Линия работает - но иначе

Вторая коробка подошла к направляющей под углом. Её передний край коснулся борта, а задний ещё двигался вместе с лентой. Оператор узнал это движение: так могла начинаться история, которую они разбирали последние дни.

Робот не потянулся к изделию.

На экране появилось сообщение: «Положение коробки перед захватом не подтверждено. Захват не выполнялся. Подача новых изделий прекращена. Проверьте обозначенную зону после подготовки участка к осмотру».

Оператор посмотрел на линию. Коробка оставалась до места захвата. Следующее изделие не приближалось к ней. На экране не было ни уверенной координаты, которой система не измерила, ни диагноза «неисправен датчик». Было указано, чего она не смогла подтвердить и что уже сделала.

- Опять коробку повело, - сказал начальник смены.
- Возможно, - ответил наладчик. - Сначала посмотрим, что произошло.

Ему уже были доступны записи о времени прохода, сигналах и состоянии привода. Оператор знал, какой участок требуется проверить. Разработчик мог проследить решение программы, не восстанавливая его по одной строке «Нарушение цикла». Начальнику смены не пришлось выбирать между немедленным пуском и ожиданием объяснения, которого ещё не существовало.

После предусмотренной проверки коробку вывели из незавершённого цикла. Причину её перекоса продолжили разбирать отдельно. Одна

остановка не доказывала, что все изменения оказались удачными; за повторениями и лишними тревогами ещё предстояло следить.

Но на этот раз небольшое отклонение не стало непонятным происшествием. Система заметила, что наблюдаемое движение больше не даёт оснований для обычного захвата, и не подменила недостающее подтверждение уверенностью. Когда участок вернули к работе, оператор уже знал, что произошло с этой коробкой и почему робот её не взял.

Лента пошла дальше.

Сайт: <https://electricschool.info/>

Канал в ТГ: <https://t.me/electricschool>

*Автоматизация и робототехника:
<https://electricschool.info/e-textbooks.html>*