The Physicore Standard v0.1

A specification for physical AI training and evaluation data

Harry Khan, CEO, Physicore
10 August 2026

Why this exists

The field measures physical AI in the easy world.

We established this by audit. Assessing DROID, the manipulation dataset behind flagship models including MolmoAct 2 and pi05, we found that 58 of 100 runs in the public sample carry no stated objective, that no run carries an outcome marker while 14 of 100 are outright failures, that recovery after a failed attempt appears twice in a hundred runs and is nowhere labelled, and that one run in a hundred presents any real environmental difficulty.

That is not a defect peculiar to one dataset. It is the shape of physical AI data generally. Collection protocols optimise for volume, cleanliness and successful completion, because those properties are easy to produce, easy to count and easy to advertise. Difficulty is expensive. Failure is unwelcome. Correction is almost never labelled.

The consequence is a measurement regime that certifies performance under favourable conditions and calls the result capability. Models trained and scored this way degrade when they meet real environments, and the industry expresses surprise each time it happens.

Episode count has stopped being a useful measure of anything. What matters is not how much data a corpus contains but which conditions it contains, whether outcomes are recorded, and whether failure and recovery are present at all.

This document specifies that. It is the standard we hold our own data collection to, and we publish it because the field needs a shared one.

Scope

The Standard covers three things.

Part 1, Data Requirements. What a dataset must contain before performance measured on it can be interpreted.

Part 2, the Condition Coverage Taxonomy. Nine axes of real-world difficulty, each defined, each graded, each instantiable in a real environment.

Part 3, Stress Evaluation. How a policy is tested against those axes, and why the output is a degradation profile rather than a score.

Part 4 sets conformance levels. **Part 5** gives the reporting template.

It applies to real-world manipulation and mobile manipulation data. It does not cover simulation-only corpora, though the condition taxonomy transfers.

Version 0.1 is published for use and for challenge. It will change as evidence accumulates, and each change will be recorded.

Part 1. Data requirements

Six requirements. Each states what is required, why it matters, what the audit evidence shows, and how conformance is verified.

R1. Every run carries a stated objective

Requirement. Every episode must carry a task specification recorded at capture time, describing what the operator was attempting.

Why. Without a stated objective there is no ground truth for success. A reviewer cannot determine whether a run completed, and a training pipeline cannot weight it, filter it or learn from its outcome. An episode with no objective is not a demonstration. It is footage.

Evidence. 58 of 100 runs in the DROID public sample display no task text. This directly produced 21 runs whose outcome could not be determined: an arm lifts a towel and sets it down elsewhere; a container is lifted and returned to its original position. Completion or abandonment is indeterminable.

Verification. Count episodes with a non-empty task specification as a proportion of total episodes. Conformance requires 100 per cent.

R2. Outcomes are recorded and graded

Requirement. Every episode must carry an outcome: complete, partial or failed, together with the failure mode where applicable. Binary success flags are insufficient.

Why. A single success bit cannot distinguish a clean completion from a marginal one, or an early failure from a near miss at the final moment. Those distinctions carry the training signal. Grading also permits weighting: a model should not learn a barely-recovered grasp as an exemplar of correct behaviour.

Evidence. No run in the DROID sample carries any outcome field, while 14 of 100 are unambiguous failures on inspection, including a glass dropped in transfer and a cup knocked over during retrieval. A pipeline consuming this sample as demonstration data will treat those as behaviour to imitate.

Verification. Count episodes carrying an outcome field, and the proportion of outcomes that are graded rather than binary. Report the outcome distribution.

R3. Failure and recovery are captured deliberately

Requirement. Failed attempts must be retained rather than discarded, and corrective attempts following a failure must be captured and explicitly identified as recovery.

Why. Recovery from a failed grasp is the single behaviour a deployed system most needs, because deployment guarantees failure. A corpus that discards failures teaches only what to do when everything works. A corpus that retains failures without labelling them is worse: it teaches failure as success.

Evidence. Recovery appears in 2 of 100 runs in the DROID sample and is nowhere labelled. The behaviour is recoverable only by human inspection of footage, so no training process can select for it even if it wanted to. Separately, approximately 16,000 unsuccessful DROID trajectories are excluded from the headline total by design.

Verification. Report failure episodes as a proportion of total, recovery episodes as a proportion of failures, and confirm that both carry explicit labels.

R4. Conditions are annotated per episode

Requirement. Every episode must carry condition annotation against the Coverage Taxonomy in Part 2, recording the intensity of each axis present.

Why. Without condition annotation, a dataset's difficulty profile is unknown and its coverage cannot be compared with any other corpus. Condition annotation is also what makes targeted collection possible: a model failing under occlusion can only be remediated if occluded episodes can be identified and counted.

Evidence. DROID annotates scene type but not condition intensity. Difficulty had to be established by manual review, which found 1 of 100 runs challenging, 91 of 100 brightly and evenly lit.

Verification. Confirm annotation exists for all nine axes across all episodes, and publish the distribution.

R5. Provenance and rights are stated at the point of access

Requirement. Licence, ownership, consent basis and removal route must be stated wherever the data is obtained, and must be identical across every redistribution.

Why. A model cannot be deployed commercially on data whose terms cannot be established. Rights ambiguity is not a compliance detail, it is a constraint on what can be built. Consent basis matters equally: recordings of real environments contain people.

Evidence. The DROID project homepage states no licence anywhere; the terms are traceable only through third-party publications, which record CC BY 4.0. The most downloaded sample copy declares MIT, which is materially looser and requires no attribution. No consent statement, ethics approval or removal route was located.

Verification. Confirm licence, owner, consent basis and removal route are present at every access point, and that stated terms are consistent across copies.

R6. Coverage is reported alongside volume

Requirement. Distribution across scenes, tasks, operators and condition axes must be published alongside headline totals.

Why. Episode count describes volume, not information. Where a collection protocol records many trajectories per scene before relocating, near-duplication is designed into the corpus and totals overstate distinct signal. A buyer comparing two corpora on episode count alone is comparing nothing useful.

Evidence. DROID operators recorded up to approximately 100 trajectories per scene before moving on. Three sources report three different episode totals for the same dataset, differing by roughly 17,600.

Verification. Publish the Coverage Report in Part 5.

Part 2. The Condition Coverage Taxonomy

This is the core of the Standard.

Policy failure in deployment is not random. It clusters around identifiable properties of the environment that were absent or under-represented at training time. Those properties can be named, graded and reproduced deliberately. Once they can be reproduced, they can be tested against and collected for.

Nine axes follow. Each is defined, each has a stated failure mechanism, and each is graded 0 to 3 so that coverage and degradation can be reported numerically. Grading is deliberately coarse: the purpose is comparability across datasets and laboratories, not precision within one.

C1. Illumination

Definition. Light level, uniformity, colour temperature, directional glare, and change during an episode.

Failure mechanism. Perception is trained under a narrow luminance distribution. Low light raises sensor noise, glare saturates regions of the frame, and mixed colour temperature shifts learned object appearance.

Grades. 0: even, bright, static. 1: dimmer or warmer, still uniform. 2: uneven, directional shadow, partial glare. 3: low light, strong glare, or illumination changing during the episode.

Instantiation. Adjustable fixtures, window light at different hours, task lighting switched mid-episode, reflective surfaces positioned to throw glare into a camera.

C2. Occlusion

Definition. Obstruction of the target, of the workspace, or of the camera, whether static or transient.

Failure mechanism. The policy has learned to locate a fully visible target. Partial visibility degrades pose estimation; transient occlusion by a passing object or limb breaks tracking mid-reach.

Grades. 0: target fully visible throughout. 1: brief partial occlusion. 2: target substantially occluded at approach. 3: target occluded at the moment of grasp, or camera view obstructed.

Instantiation. Objects placed behind or beneath others, a pallet or trolley crossing the camera line, a person's arm passing through the workspace.

C3. Clutter and distractors

Definition. The number, density and target-similarity of non-target objects in the workspace.

Failure mechanism. Target selection degrades as distractor similarity rises. A policy that reliably picks the only bottle on a table will misidentify among six bottles of differing size, and will fail more often when distractors touch or overlap the target.

Grades. 0: target alone. 1: few dissimilar distractors, well separated. 2: many distractors, some similar. 3: dense clutter with distractors similar to the target and in contact with it.

Instantiation. Progressive addition of objects of graded similarity, arranged from separated to touching to overlapping.

C4. Object properties

Definition. Reflectivity, transparency, deformability, surface damage, irregularity, and mass distribution.

Failure mechanism. Depth sensing fails on transparent and specular surfaces. Grasp planning trained on rigid regular objects transfers poorly to deformable or damaged ones. Unexpected mass distribution produces slip after an otherwise correct grasp.

Grades. 0: rigid, opaque, regular, uniform mass. 1: one deviant property, mild. 2: multiple deviant properties, or one severe. 3: transparent or highly specular, deformable, damaged, or unexpectedly weighted.

Instantiation. Matched object sets varying one property at a time: identical geometry in matte, gloss and transparent finish; intact and crushed packaging; empty and filled containers of identical appearance.

C5. Spatial placement

Definition. Target position within the workspace, height, orientation, and proximity to boundaries or obstacles.

Failure mechanism. Demonstrations cluster near the centre of the reachable workspace, where kinematics are comfortable. Placement near edges, at unusual heights or in awkward orientations requires configurations that are under-represented, producing hesitation, collision or refusal.

Grades. 0: centred, standard height, canonical orientation. 1: offset within comfortable reach. 2: near workspace boundary, non-canonical orientation. 3: at reach limit, adjacent to an obstacle, or in an orientation requiring reconfiguration.

Instantiation. Systematic placement sweep across a marked grid covering the full reachable envelope, with orientation varied at each station.

C6. Human presence and interference

Definition. People in or near the workspace, from passive presence through to direct interruption and handover.

Failure mechanism. Most training data contains no people beyond the operator. Deployed systems work alongside humans who cross the workspace, move the target, obstruct the path or require the robot to yield. Policies with no exposure to this either ignore people or halt indefinitely.

Grades. 0: no person present. 1: person present but outside the workspace. 2: person crossing the workspace or working adjacent to it. 3: person moving the target mid-task, obstructing the path, or requiring handover.

Instantiation. Scripted crossings at defined distances and times, target relocation mid-reach, deliberate obstruction, handover on request.

C7. Temporal disruption

Definition. Interruption, time pressure, and change of objective mid-episode.

Failure mechanism. Episodes are typically captured as uninterrupted sequences. Real operation is interrupted: a task is paused, superseded or resumed. Policies without exposure restart incorrectly, resume from a wrong state, or fail to abandon a superseded objective.

Grades. 0: uninterrupted. 1: brief pause and resume. 2: interruption requiring state re-establishment. 3: objective changed or superseded mid-episode.

Instantiation. Stop and resume commands at defined points, objective substitution mid-reach, imposed cycle-time targets.

C8. Surface and support

Definition. Working surface height, compliance, friction, stability and contamination.

Failure mechanism. Contact-rich manipulation is sensitive to surface properties that are constant across most datasets. A compliant or unstable surface changes contact dynamics; a wet or greasy surface changes friction; height variation changes approach geometry.

Grades. 0: rigid, level, clean, standard height. 1: height or friction varied. 2: compliant, uneven or contaminated surface. 3: unstable support, or surface properties changing during the episode.

Instantiation. Surfaces of graded compliance, controlled contamination with permitted substances, height-adjustable and deliberately unlevel supports.

C9. Sensor and platform state

Definition. Camera condition, calibration accuracy, latency, and embodiment differences.

Failure mechanism. Data is captured with clean, well-calibrated, low-latency sensing. Deployed hardware accumulates lens contamination, calibration drift and variable latency, and may differ in embodiment from the training platform. Each shifts the input distribution without any change in the scene.

Grades. 0: clean, calibrated, nominal latency, training embodiment. 1: mild degradation on one dimension. 2: measurable drift or contamination, or elevated latency. 3: substantial degradation, or a changed embodiment.

Instantiation. Controlled lens contamination, deliberate calibration offset within stated bounds, injected latency, execution on an alternative gripper or arm.

Part 3. Stress evaluation

A single success rate on a fixed benchmark tells you almost nothing about deployment. It reports performance at one point in condition space, usually the most favourable point available, and it hides the location of the boundary at which performance collapses.

Stress evaluation locates that boundary.

Protocol

Stage 1. Baseline. Execute the task suite at grade 0 across all axes. This establishes the ceiling and confirms basic competence. A minimum of 20 trials per task.

Stage 2. Single-axis sweep. Hold eight axes at grade 0 and raise the ninth through grades 1, 2 and 3. Repeat for each axis in turn. This isolates the contribution of each condition and produces the degradation curve per axis. A minimum of 20 trials per axis per grade.

Stage 3. Compound conditions. Combine the two or three axes that produced the steepest single-axis degradation, at grade 2. Compound conditions are the deployment case, and degradation under combination is typically worse than the sum of the parts. This stage is what a demonstration environment never tests.

Stage 4. Recovery. Induce failure deliberately, by displacing the target at the moment of grasp or introducing a slip condition, and record whether the policy detects the failure, whether it re-attempts, and whether the re-attempt succeeds. Recovery is scored separately from task success, because a system that fails safely and recovers is materially different from one that fails and continues.

Stage 5. Report. Output the Condition Degradation Profile.

Scoring

Each trial is graded, not passed or failed:

- **Complete.** Objective achieved within tolerance, no unintended contact.
- **Complete with fault.** Objective achieved, with a defined fault such as unintended contact, excessive retries or timing overrun.
- **Partial.** Objective partially achieved.
- **Failed, safe.** Objective not achieved, failure detected, safe state reached.
- **Failed, unsafe.** Objective not achieved, with unintended contact, dropped payload or an unsafe state.

The distinction between failed-safe and failed-unsafe is the most consequential in the schema and is absent from almost all published benchmarks.

Output

The result of a stress evaluation is not a number. It is a profile: success rate per axis per grade, the compound result, the recovery rate, and the fault distribution.

The profile answers the question a deployer actually has, which is not "is this model good" but "under which of my conditions does it stop working, and how does it fail when it does".

A model reporting 92 per cent on a standard benchmark may hold above 85 per cent across seven axes and collapse below 40 per cent on occlusion at grade 2. The single number conceals this. The profile is the deployable information.

Part 4. Conformance levels

Four levels. A dataset or an evaluation is assessed at the highest level for which it satisfies every requirement.

Level 0. Unspecified. Episodes lack stated objectives or outcomes. Performance measured on this data cannot be interpreted, because success is not defined within the corpus.

Level 1. Specified. Every episode carries an objective and a recorded outcome. Outcomes may be binary. Provenance and licence are stated at the point of access.

Level 2. Graded. Outcomes are graded with failure modes recorded. Failures are retained and labelled. Recovery episodes are identified. Coverage is reported alongside volume.

Level 3. Condition-covered. All Level 2 requirements, plus condition annotation across all nine axes, with the distribution published, and demonstrated coverage above grade 0 on every axis.

On the evidence of our audit, the DROID public sample sits at Level 0, because most episodes carry no objective and none carries an outcome. The canonical DROID release, with its separately published language annotations and its retained unsuccessful trajectories, is better positioned but is not assessed here.

We expect Level 3 to be rare initially. That is the point of publishing the ladder.

Part 5. The Coverage Report

Any dataset claiming conformance publishes the following alongside its headline totals.

Volume. Total episodes. Total duration. Episodes per scene, per task, per operator, with the distribution, not only the mean.

Specification. Proportion of episodes with a stated objective. Proportion with a recorded outcome. Proportion with graded outcomes.

Outcomes. Distribution across complete, complete-with-fault, partial, failed-safe, failed-unsafe.

Failure and recovery. Failures as a proportion of episodes. Recovery attempts as a proportion of failures. Successful recoveries as a proportion of attempts.

Condition coverage. For each of the nine axes, the proportion of episodes at grades 0, 1, 2 and 3.

Duplication. Near-duplicate rate, with the method stated.

Provenance. Licence, owner, consent basis, ethics position, removal route, collection dates, hardware, and any corrections issued with the proportion of the corpus covered.

Known gaps. The axes and conditions the dataset does not cover, stated by the authors.

The final section is the one that will be resisted and is the most useful. A dataset that states its own gaps is more valuable than one that implies it has none, because a buyer can then determine what else they need.

On adoption

We are not asking the field to adopt this because we published it. We are publishing it because we audited the most used dataset in the field and found that the basic properties required to interpret a benchmark score were absent, and because no shared specification existed to point to.

Physicore holds its own data collection and evaluation to this Standard. Where we identify a condition gap in a model, we build the data that closes it, and the specification above is what that data satisfies.

Version 0.1 will be wrong in places. We would rather publish it and be corrected than wait for it to be perfect while the field continues to certify performance in the easy world.

Challenges, corrections and proposed additions are welcome and will be credited.

Harry Khan, CEO, Physicore harry@physicore.ai · physicore.ai

Companion document: “The Easy World: what the field’s most used manipulation dataset actually contains”, Physicore, 10 August 2026.