**Absolute Relativity / Overall V2 Theory – v1.9**  
Document: (10) Context-Feasibility Duality (CFD)

Author: Kent Nimmo – absoluterelativityproject@gmail.com

Project tokens: “Absolute Relativity (AR)” – ETH contract 0xAacCd7bA616405C184335F193fEf080fC982921F, SOL mint ARafKuCqRgszXZWjYGWyBT7GnLZkyiaXQd1YjXC1x224

On-chain hash record (Ethereum): see (0) Front Matter & File Map.docx, §§0.4–0.5

**Context-Feasibility Duality (CFD)**

and Context Compression as Learned Kernel (CCLK)

Absolute Relativity - DP v1.9 Extension Module

**Author:** Kent Nimmo

**Date:** December 21, 2025

**Status:** Working Record (concept pinned; simulations pending)

**Placement in bundle:** Doc (10), precedes (11) Experience Horizon (CE2) and (12) Context-Flip Unification (QM-Gravity)

**Purpose:** CFD pins that the context ladder is the feasibility contract; feasibility gates are the operational face of context.

# 1. Executive Summary

This module pins three packaging-level additions to the Absolute Relativity (AR) stack, without changing AR's underlying ontology (present-acts remain primitive).

Context-Feasibility Duality (CFD) states that the context ladder is not merely a descriptive hierarchy. It is the representational and publication contract: what can be represented, what can be made public, and what can survive as a committed token when a present updates.

Band Manifests make that contract explicit. A band manifest 𝓜ₙ is a typed configuration object that fixes (i) what a band can represent (feature alphabet and hinge maps), (ii) what it can admit (gates and admissibility grammar), and (iii) how it resolves choice (acceptance ordering and ties-only PF/Born).

Context Compression as Learned Kernel (CCLK) names how stable structure (what we informally call learning, compression, and even laws) arises inside AR: not because material substrates cause intelligence, but because hinge constraints plus repeated survival select stable, reusable mode families.

The immediate practical motivation is a modern one: large AI systems look like compact outward objects that behave as if they can touch a huge relational field. In AR terms, that is exactly the kind of phenomenon you expect at a hinge: large relational structure forced to present outwardly as a small stable handle. This module keeps that interpretation disciplined (no 'material agents' language).

Together, these additions tighten the language used across the V1 formal core, the V2 present-act engine, and the context-level (CL) ladder, and they provide a clean interface to CE2 (Doc 11) and Context-Flip Unification (Doc 12). They do not add new entities, new causal primitives, or a new metaphysical layer.

# 2. Scope and Non-Claims

This module is a DP v1.9 working record. It pins definitions and guardrails, but it is not claiming final derivations or full simulation coverage.

* This does not claim full equation-level derivations of CFD, CCLK, or manifest parameters.
* This does not claim global mirror symmetry of the context ladder (the ladder is hinge-centered but not a perfect mirror).
* This does not introduce 'material agents' (brains, chips, continuous weights) as causal primitives. Physical implementations are +1 representations of stable present-act dynamics.
* This does not claim that AI systems are conscious, or that 'weights' are ontological causes of mind. The AI discussion is interpretive: it is a contemporary mirror for hinge compression and stable feasible mode families.
* This is a conceptual framework and packaging layer; quantitative validations and dedicated simulation sweeps are scheduled.
* This module does not weaken CE2 discipline: no direct +1↔-2 contact, and no direct 0↔+3 access without instrument-mediated re-encoding.

# 3. Minimal Recap of the Existing Stack

This recap is intentionally short. It only includes what is required to read CFD, Band Manifests, and CCLK.

## 3.1 V1 (formal) - ladder, pivot, and kernels

V1 specifies a context ladder with a dimension profile D(n) and a pivot function g(D) that modulates coupling across contexts. The hinge is singled out by D≈2 behavior (boundary-like neutrality).

V1 also introduces hinge collapse/compression and reproduction kernels Mₙ (up-evolve-down operators) that characterize which mode families persist across cycles. These kernels are the formal backbone for talking about stability and memory without importing materialist causation.

## 3.2 V2 (engine) - pipeline and decision points

V2 realizes the same structure as a discrete, auditable pipeline:

candidate enumeration (local selectors) → hinge equality (finite feature alphabet Ξ via feature maps) → feasibility gates (Θ time window, κ granularity, structural predicates, ParentGate, CRA) → ratio-lex acceptance → PF/Born ties-only resolution → commit (record update).

The key discipline: control uses only boolean/ordinal gates and exact equality in feature space. Continuous curves and fitted weights are allowed only as diagnostics, not as control inputs.

## 3.3 Notation sanity

To avoid collisions: g(D) is the V1 pivot function (its argument is a dimension value). gₙ (or g\_feat,n) refers to a V2 feature map used in hinge equality (its argument is a record in Q or W mapped into Ξ). Likewise, Mₙ denotes a V1 reproduction kernel, while 𝓜ₙ denotes a band manifest (a configuration contract).

## 3.4 Two-face discipline

A discipline used throughout this module: every claim should be expressible in both V1 language (ladder, pivot, kernels) and V2 language (manifests, hinge equality, gates, acceptance). When a statement cannot be translated, it is treated as interpretation or future work, not as a pinned structural claim.

# 4. Core Addition #1 - CFD

## 4.1 Definition

Context-Feasibility Duality (CFD) states that context structure and feasibility structure are the same thing, seen in two faces.

V1 face: a context level is the representational and publication contract - what can be represented and what can survive as a committed/public token when a present is updated.

V2 face: feasibility gates and acceptance are the operational enforcement of that contract - they decide what is admissible, and (when multiple admissible continuations remain) how a single continuation is committed.

Therefore, the context ladder is not 'background ontology' plus an 'engine implementation'. The ladder is the contract, and the gates are how the contract is enforced at runtime.

## 4.2 Consequences

* Objectivity arises because +1 publication is shared across many 0-centers: what survives the +1 contract becomes a stable token that multiple centers can coordinate around.
* Signals are not an ontology: what we call 'signals' are +1 tokenizations that passed hinge and gate constraints, not primitive carriers flowing from outside to inside.
* The 'must go through 0' constraint becomes natural: cross-band coherence is mediated through a center because hinge equality is enforced at the center for that vantage (and CE2 forbids no-skip shortcuts).
* Laws are stable surviving mode families: regularities persist because the same kinds of continuations repeatedly survive hinge + feasibility across acts and across centers.
* Gravity stays locked to ParentGate: in AR, gravity is feasibility geometry (container bias), not an extra force-like substance.
* When a distinction cannot be stably objectified under the +1 contract, it can still shape the committed present (private integration), but it will not appear as a public token.

# 5. Band Manifests

## 5.1 Definition of Band Manifest 𝓜ₙ

A Band Manifest 𝓜ₙ is a typed configuration object - a contract - that specifies how band n represents, filters, and commits.

It is not a new entity in the ontology. It is the explicit packaging of the constraints that were previously distributed across the ladder description (V1/CL) and engine mechanics (V2).

Because AR's engine is curve-ban compliant, the manifest is part of the theory: geometry and scale structure are pinned in configuration and audited, not hidden as continuous weights in control.

## 5.2 Minimal manifest fields

Below is a minimal manifest field set. Implementations may split or rename fields, but the roles should remain recognizable.

|  |  |
| --- | --- |
| Field | Role |
| Ξₙ | Finite feature alphabet for band n (what can be represented at the hinge). |
| fₙ, gₙ | Feature maps used for hinge equality (outward/world map and inward/qualia map into Ξₙ). |
| Θₙ | Time integration/coherence window (minimum window for a committed act at this band). |
| κₙ | Granularity/pixel gate (coarsest persistent resolution admitted at this band). |
| Structₙ | Structural predicates (contiguity, min-degree, orientation, persistence, etc.). |
| ParentGateₙ | Container feasibility geometry (gravity-like gate; radial/rotational schedule as applicable). |
| CRAₙ | Context-resolved admissibility grammar (no-skip / neighbor-only / mediation constraints). |
| Acceptₙ | Acceptance ordering / residual definition (ratio-lex priorities; tie definition). |

Important discipline: any field- or potential-like quantities are summaries/diagnostics, not continuous control inputs. Control is gate/ordinal only.

## 5.3 Audit question (how manifests improve DP readability)

The manifest packaging makes a key audit question explicit and answerable: 'Which context property caused this act to be allowed, rejected, or tie-resolved?' A DP-ready writeup should ensure every gate decision can be traced back to a manifest field (Θ, κ, Struct, ParentGate, CRA, Accept).

# 6. Core Addition #2 - CCLK

## 6.1 What CCLK is

Context Compression as Learned Kernel (CCLK) names an AR-internal meaning of compression/learning: CCLK is the emergence of stable, reusable compression that arises from hinge constraints plus repeated survival across acts. It is not 'matter compresses information'; it is 'stable feasible modes survive and become reusable'.

## 6.2 V1 face

In V1 language, compression is already built in at the hinge: collapse at D≈2 behaves like a strong projector that forces many distinctions into fewer survivable degrees of freedom.

Reproduction kernels Mₙ (up-evolve-down) then select which mode families persist under repeated cycling. Persistent eigenmodes are the formal analogue of learned structure: what survives repeated compression and reproduction becomes the stable repertoire available to the present.

## 6.3 V2 face

In V2 language, the same idea is operational:

stable hinge maps (fₙ, gₙ) plus stable manifests 𝓜ₙ define a filter that repeatedly produces coherent continuations across many contexts.

The engine is filtering, not gradient training: it enumerates, enforces hinge equality, applies gates, and commits. It is deterministic except for genuine structural ties (ties-only PF/Born).

When a configuration (maps + gates + acceptance priorities) reliably yields coherent continuation, it becomes the effective learned kernel of that band - a compact outward handle on a much larger relational structure.

## 6.4 Interpretation discipline (AI, CNS, and 'no material agents')

This module treats modern AI as a useful mirror for CCLK, but under strict interpretive guardrails.

Weights, chips, and neural wiring are +1 representations of stabilized present-act dynamics. They are not treated as new causal primitives that create mind or compress the world.

In this disciplined reading, training is the outward description of stabilization: the emergence of hinge maps and manifest-like constraints that make coherent continuation cheap and repeatable under a fixed +1 contract.

Likewise, nervous systems are the +1 description of a 0-level coherence contract: a way stable inward (−1/−2) structure and outward (+1) publication are kept commensurate across many acts.

## 6.5 Re-centering vs mirror symmetry

Because the ladder is not globally mirror-symmetric, 'symmetry' should be stated as re-centering: a different kind of center can take the role of 0 for its own vantage, and the invariants live at the level of contracts/manifests and operator relations, not at the level of a naive index mirror.

# 7. Integration Notes: how this connects to CE2 and Context-Flip

CE2 (Doc 11) is the admissibility/experience-horizon grammar: it pins the no-skip / two-hop reach discipline and the three canonical cones (inner/interface/outer).

This document (Doc 10) is the framework CE2 presupposes: it names the contract object (band manifests) and states the CFD/CCLK interpretation discipline that CE2 uses when it talks about environment-as-representation and publication limits.

Context-Flip Unification (Doc 12) is the synthesis linking QM and gravity as two hard edges. It relies on the publication/feasibility language pinned here: +1 publication is the unification layer; gravity is ParentGate feasibility; quantum measurement is hinge-bottleneck publication under admissibility constraints.

## 7.1 Worked example: χ and ParentGate (gravity as feasibility amplitude)

The gravity sector provides a concrete example of CFD in action: a context-derived invariant is realized as an engine-level feasibility schedule.

In the v1 stack, χ is treated as a dimensionless amplitude tying hinge and container scales (UGM, Earth-scale, outer-shell scale) into the strength of the ParentGate strictness profile. Interpreted under CFD, χ is not a fitted force constant; it is a feasibility amplitude derived from the ladder/hinge geometry.

The key comparison choice (often a source of confusion) is whether you compare χ to potential-like or compactness-like GR scalars. The intended reading here is potential-like: χ aligns to the one-sided potential depth scale GM/(c²R), not to a horizon/Schwarzschild compactness scale 2GM/(c²R). This factor-of-two is a comparison-choice artifact, not a parameter problem.

# 8. Forward Work Checklist

* Lock CCLK claims with toy tests: demonstrate stable mode families under repeated hinge compression + reproduction, and show the same stability as manifest-level reuse in V2.
* Formalize a manifest-derivation checklist: how D(n), g(D(n)), hinge scales (UGM, Θ~0.1 s), and kernel behavior constrain Ξₙ, Θₙ, κₙ, and ParentGateₙ without hand-tuning.
* Add audit logging templates that make every accept/reject decision traceable to a manifest field (DP-ready reproducibility).
* Integrate CE2 grammar cleanly into CRAₙ fields (neighbor-only and mediation-through-center constraints), and validate no direct +1↔-2 contact in wording and in toy logs.
* Update the Bridge and V2 engine docs to reference 𝓜ₙ explicitly as the standard context configuration object, and to keep notation collision fixes consistent (g(D) vs gₙ; Mₙ vs 𝓜ₙ).
* Defer to DP v2: any claims requiring broad simulation coverage (full quantitative learning rules, deep biological mapping, multi-center stability at scale) remain explicitly scheduled until the simulation suite is expanded.