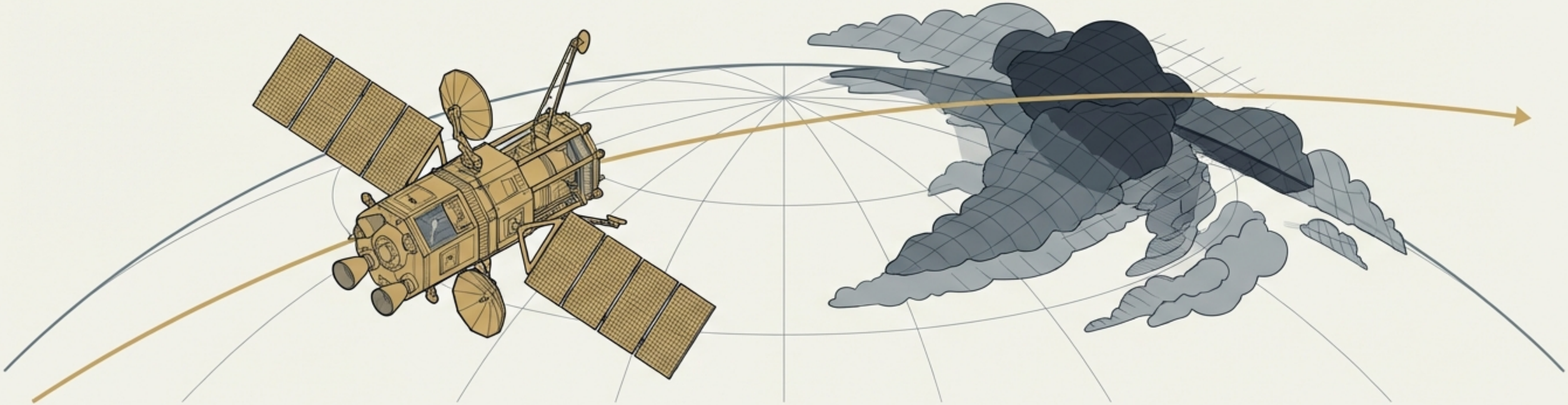
TENSOR™ NETWORKS

The Deterministic Constellation

Bridging Orbital Physics and Data-Plane Execution with the Predictive Tensor Control Plane

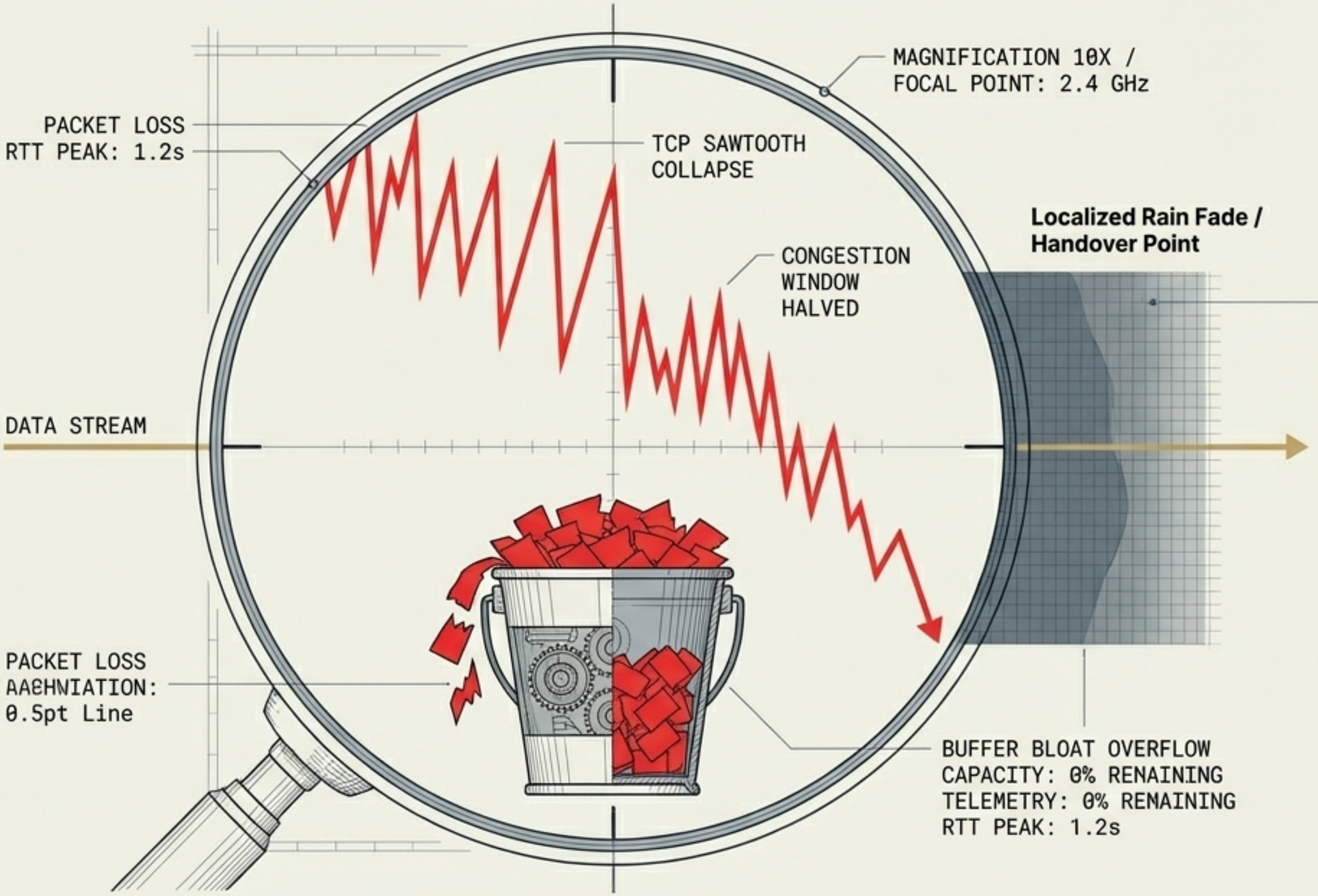
The Ephemeral Physical Substrate



Velocity	Volatility	The Hard Limit
Satellites traverse the sky at Mach 22 (~27,000 km/h).	The network topology is in constant flux, subject to multipath fading, Doppler shifts, and atmospheric rain fade.	Ground terminals are forced to execute physical handovers every 3 to 5 minutes.

Takeaway: Traditional networking assumes a static physical topology. In Low Earth Orbit, the physical layer is inherently hostile to static assumptions.

The TCP Interpretation Error



The Misinterpretation

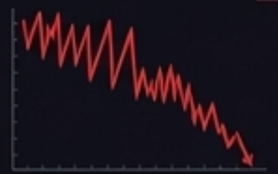
Legacy TCP/IP stacks misinterpret physical layer events (a 3-minute handover or sudden rain fade) as network congestion.

The Chain Reaction

This misinterpretation triggers a catastrophic sawtooth collapse—halving the congestion window, forcing aggressive slow-start recovery, and creating localized buffer bloat and severe jitter.

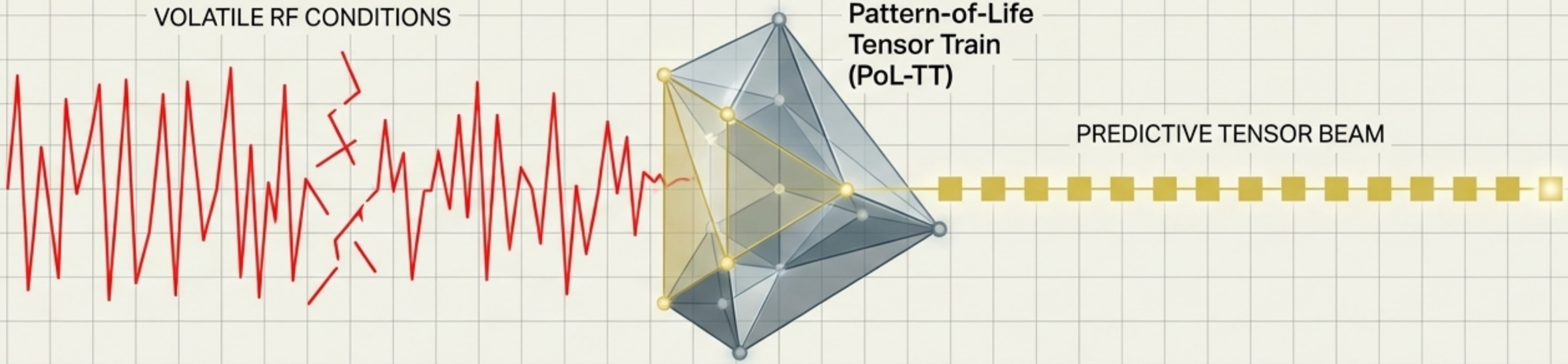
Architectural Mismatch: Terrestrial Fiber vs. Orbital Mesh

FK Grotesk, Matte Bone White

	Terrestrial Fiber (Legacy)	**Orbital Mesh** (LEO)
Topology Stability	Static (Constant links/routes)	Highly Dynamic (Mach 22 continuous movement)
Handover Frequency	Rare (Managed via roaming)	Continuous (Hard limits every 3–5 mins)
Packet Loss Interpretation	Assumed Buffer Congestion	Driven by Physics (Rain fade, ISL alignment)
Congestion Response	Reactive (TCP Sawtooth Window) 	Predictive (eBPF Inter-packet pacing) 

The Predictive Tensor Control Plane (PTCP)

TENSOR™ NETWORKS



Core Concept

PTCP replaces reactive feedback loops with an execution model anchored in Keplerian Mathematics. Because LEO satellite movement is perfectly deterministic, PTCP utilizes Orbital Ephemeris to anticipate physical link changes before a packet is ever lost.

The Goal

Decouple the terrestrial user experience from orbital volatility.

System Architecture: Orchestration vs. Actuation

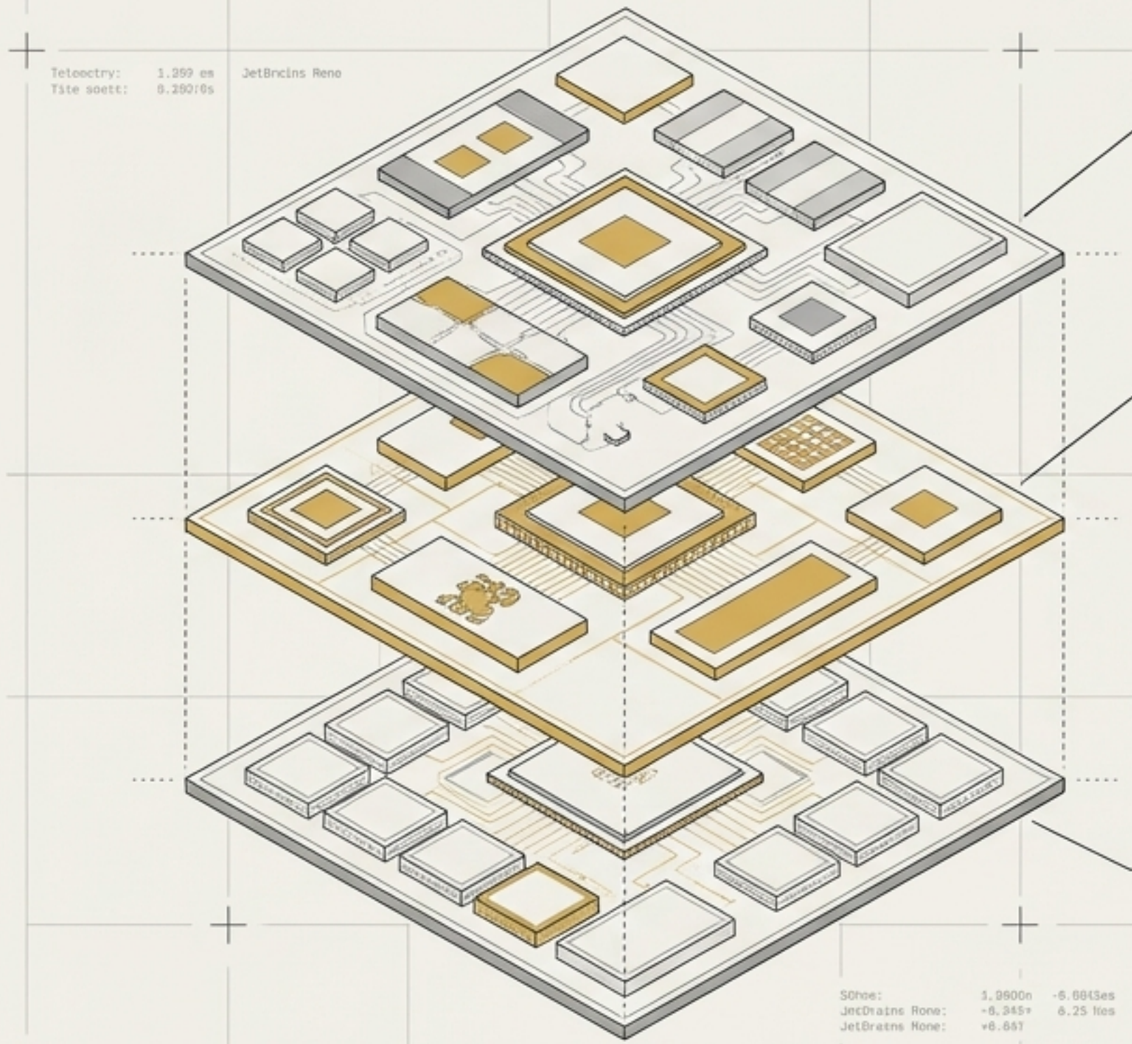
The Go Orchestrator
(User Space - The Mind)

Core Function:	High-frequency mathematical modeling (Keplerian Math & SVD).
Execution Speed:	100ms ephemeris ingestion loops.
Output:	Calculates and generates Bounding Policy Envelopes.

The eBPF Actuator
(Kernel Space - The Muscle)

Core Function:	Earliest Departure Time (EDT) O(1) packet pacing.
Execution Speed:	Nanosecond, line-rate execution.
Output:	Explicitly overwrites the skb->tstamp metadata to enforce pacing.

Embedded Edge Readiness



Top Layer (User Space)

Statically linked Go binary (<10MB footprint). Zero terrestrial dependencies (no iproute2).

Middle Layer (Linux Kernel)

eBPF/TC Subsystem. Configured with specific flags:

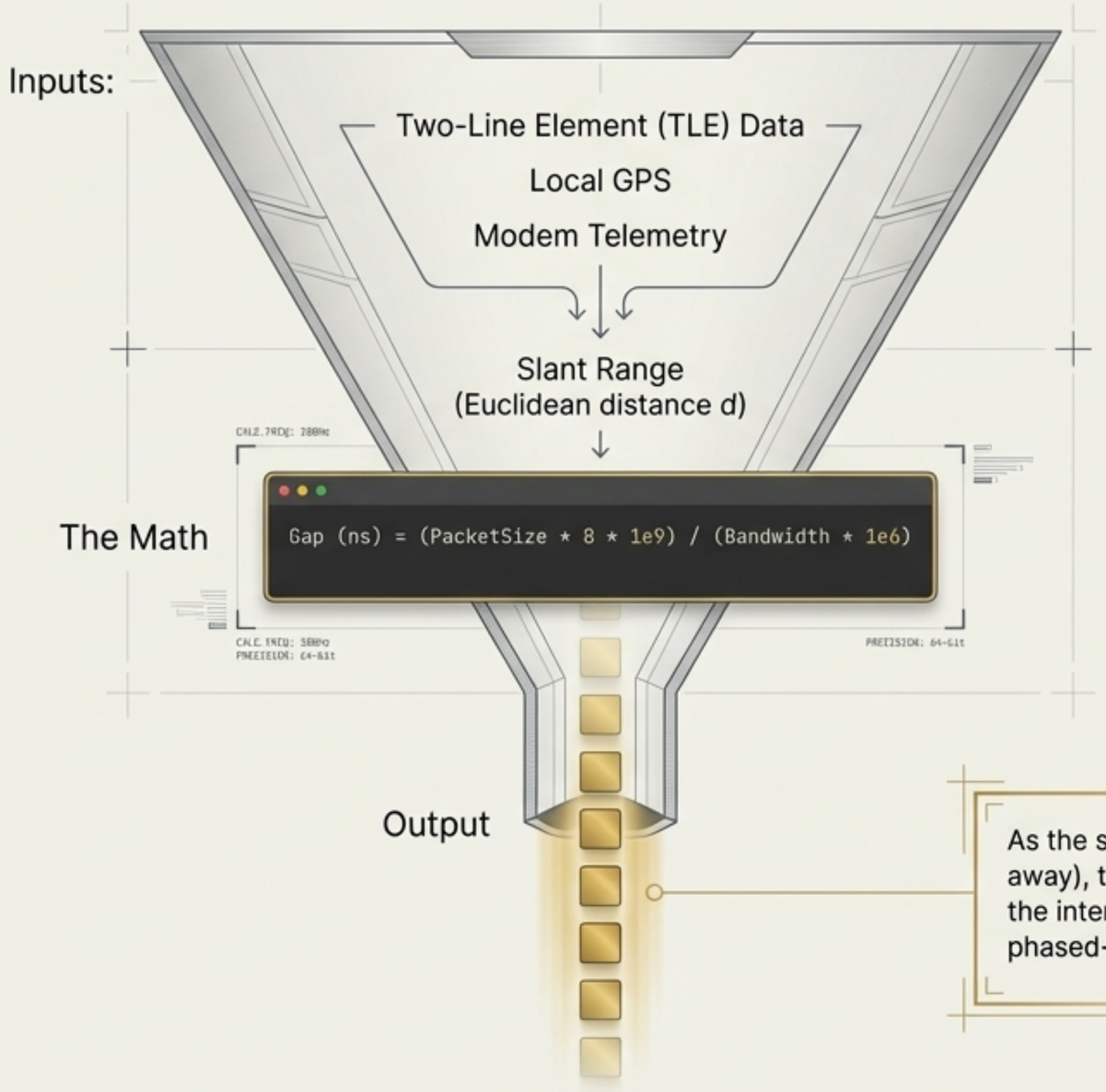
```
CONFIG_BPF_JIT=y
CONFIG_DEBUG_INFO_BTF=y
tc qdisc add dev root fq
```

Bottom Layer (ARM64 Hardware)

Thermally constrained satellite modem.

<0.1% CPU utilization at Gigabit line rates.

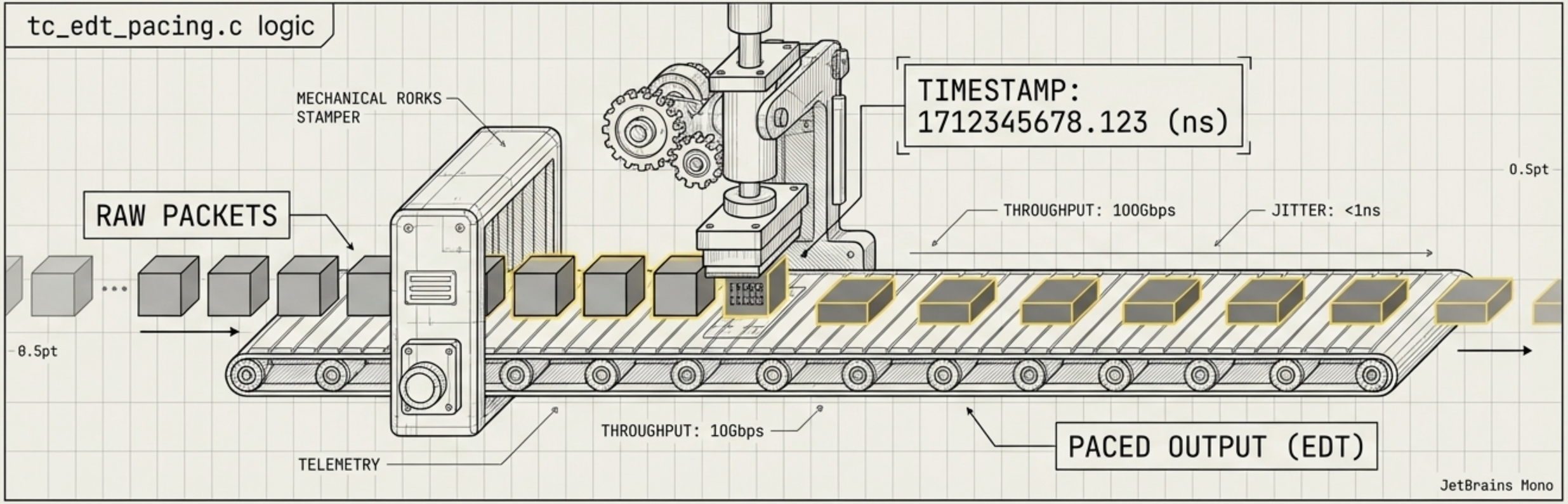
Bridging Ephemeris to Execution



As the slant range increases (satellite moving away), the Go Daemon dynamically widens the inter-packet gap (Δt) to prevent phased-array hardware buffers from saturating.

Earliest Departure Time (EDT) Pacing Engine

TENSOR™ NETWORKS



1. Extract Flow ID

eBPF verifies safety and uses destination IP as the flow key.

2. O(1) Timestamp Manipulation

Bypasses bursty congestion control by overwriting skb->tstamp with a mathematically precise release time.

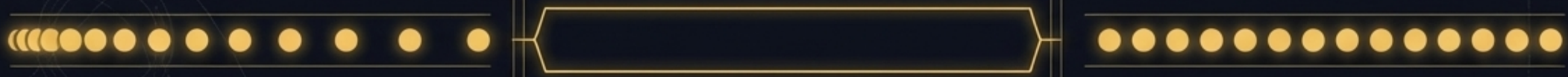
3. Lockless Fair Queueing

The kernel holds the packet in the fq scheduler, completely eliminating terrestrial burst-and-wait cycles.

Efficiency Callout: JIT-compiled into a mere 20-30 native ARM64 instructions.

Zero-Jitter Delivery: The Make-Before-Break Sequence

	T-Minus 5 Seconds (Envelope Tightening)	The Handover (Beam Steering)	Envelope Release (Recalculation)
PHYSICAL (Hardware)	Target identified.	Phased-array electronically steers to new satellite.	Lock confirmed on new satellite.
SOFTWARE (Kernel Pacing)	NanosecGap proactively widens. Data decelerates smoothly; zero loss induced.	Modem buffers are perfectly empty. No stale data to drop.	Slant range recalculated, pacing constraint released. Immediate return to maximum throughput with no TCP slow-start penalty.



Quantifying the Invisible ROI

`ptcp_micro_stalls_prevented_total`

The Observability Challenge

PTCP eradicates micro-stalls and buffer bloat before they occur. Standard tools cannot record drops that were successfully prevented.

We must expose specialized metrics to prove system efficacy.

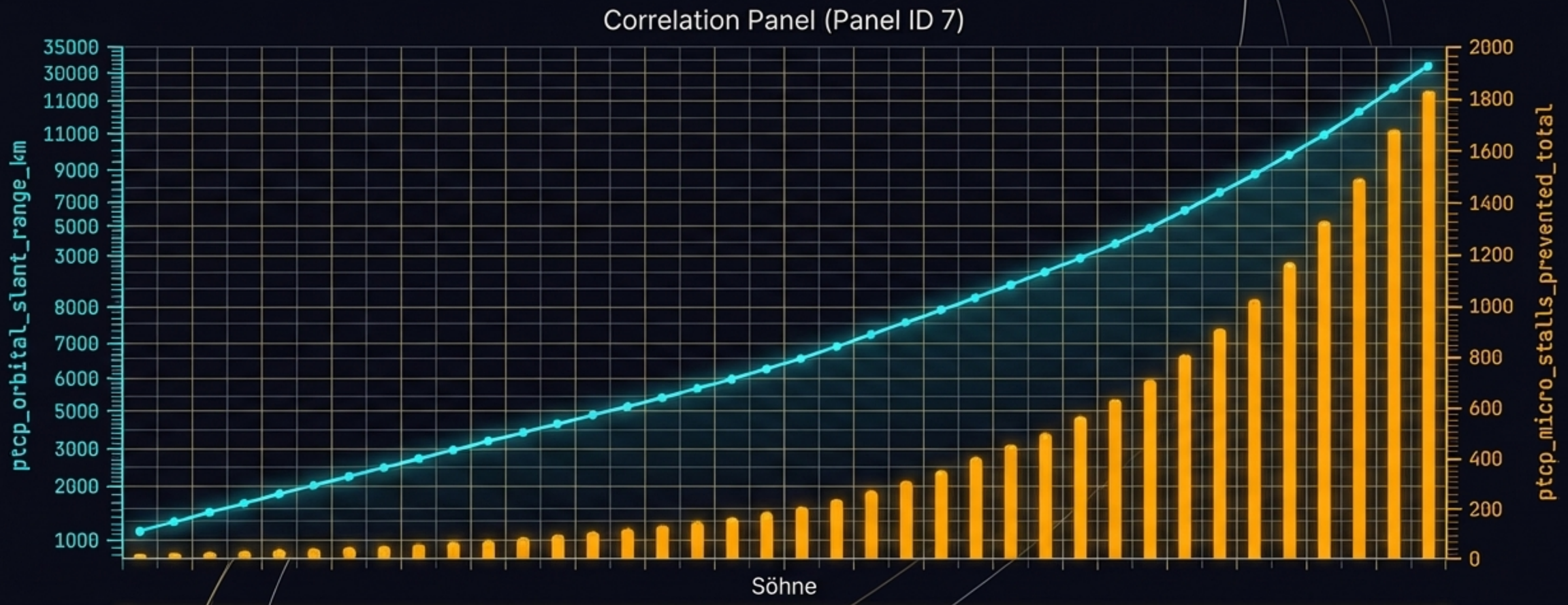
The Telemetry Strategy

Telemetry is gathered via a Prometheus pull-model over a dedicated management VLAN. Unlike

push-based UDP, metrics are held in memory until requested, ensuring telemetry never competes with user data for vital uplink spectrum.

The Predictive Determinism Dashboard

TENSOR™ NETWORKS

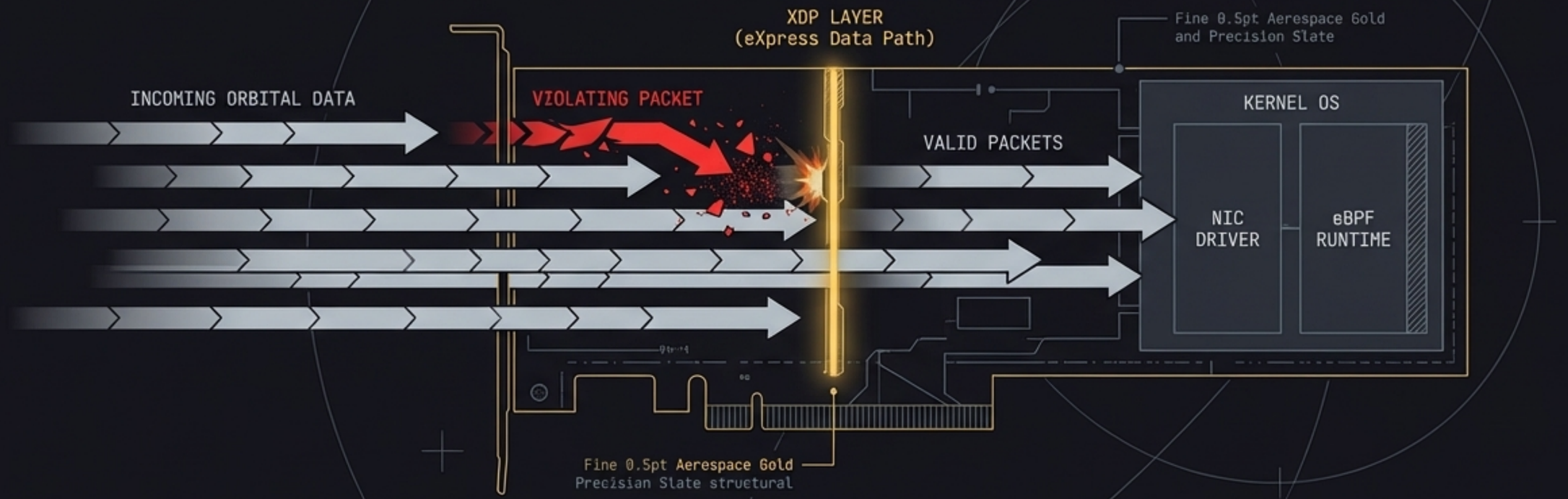


The Synthesis Insight: As slant range increases, prevented stalls rise concurrently. This visual correlation is empirical proof that PTCP successfully catches would-be TCP sawtooth failures, smoothing them into a seamless, mathematically governed bitstream.

SVD-Derived Operational Thresholds

Metric	Warn	Crit	Action
`ptcp_rf_attenuation_db`	⚠ 2 dB	⚠ 4 dB	Verify PTCP is widening nanosec gaps to protect uplink spectrum.
`ptcp_orbital_slant_range_km`	⚠ 800 km	⚠ 1100 km	Monitor for increased micro-stall prevention as link distance grows.
`ptcp_effective_bandwidth_mbps`	⚠ < 10 Mbps	N/A	Investigate degradation; verify PoL-TT state is maintaining minimum floor.

XDP Zero-Trust Perimeter & Spectrum Preservation



Zero-Trust at the Edge

Ingress data is evaluated against the PoL-TT state via eXpress Data Path (XDP). Violating packets are dropped in O(1) time at the NIC driver level, shielding the terminal from ISL anomalles.

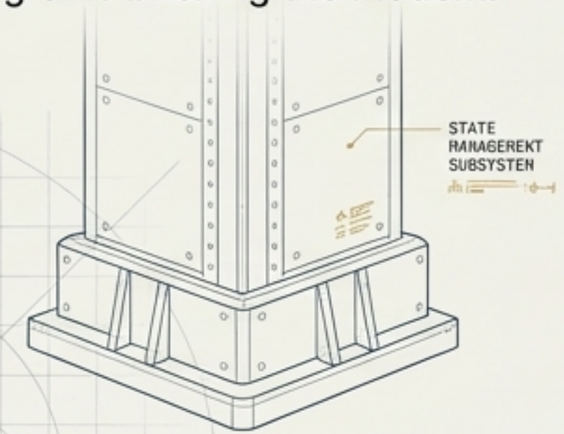
TCP ACK Thinning

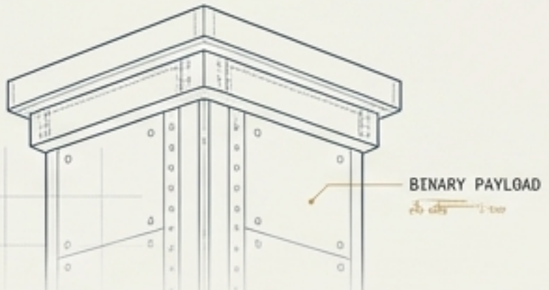
To preserve highly constrained uplink RF spectrum, the eBPF program mathematically coalesces TCP ACKs based on known orbital latency—drastically reducing overhead without ever triggering transport retransmissions.

The Three Pillars of Edge Reliability

Pillar 1: Atomic State Management

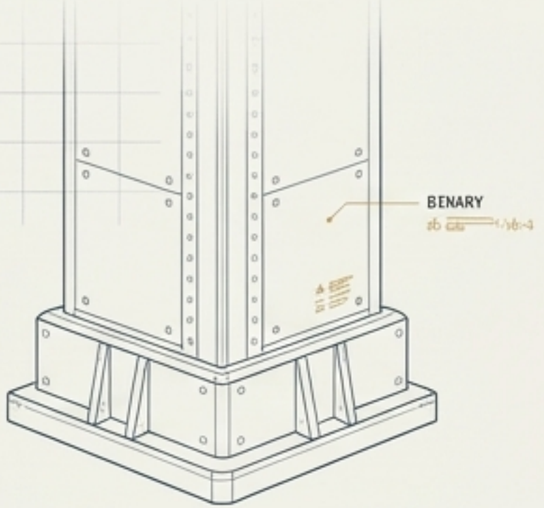
Powered by Go and `Netlink`. If the user-space daemon fails, the **kernel's reference counter** drops and hooks are automatically removed. Zero risk of zombie programs bricking the modem.





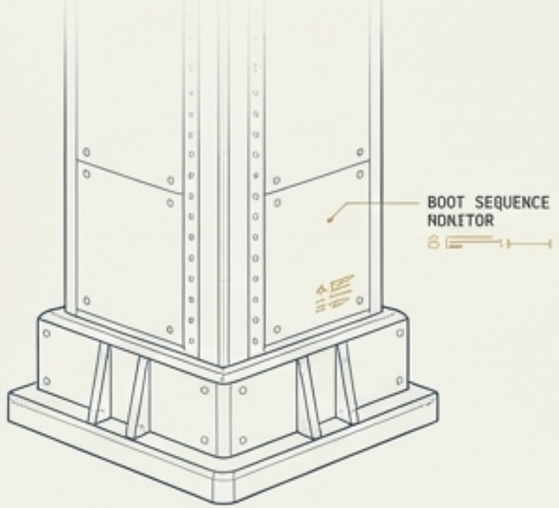
Pillar 2: Zero Dependency Footprint

The orchestrator and actuator are a single, statically linked binary (<10MB). No reliance on fragile shell scripts or terrestrial firmware utilities.

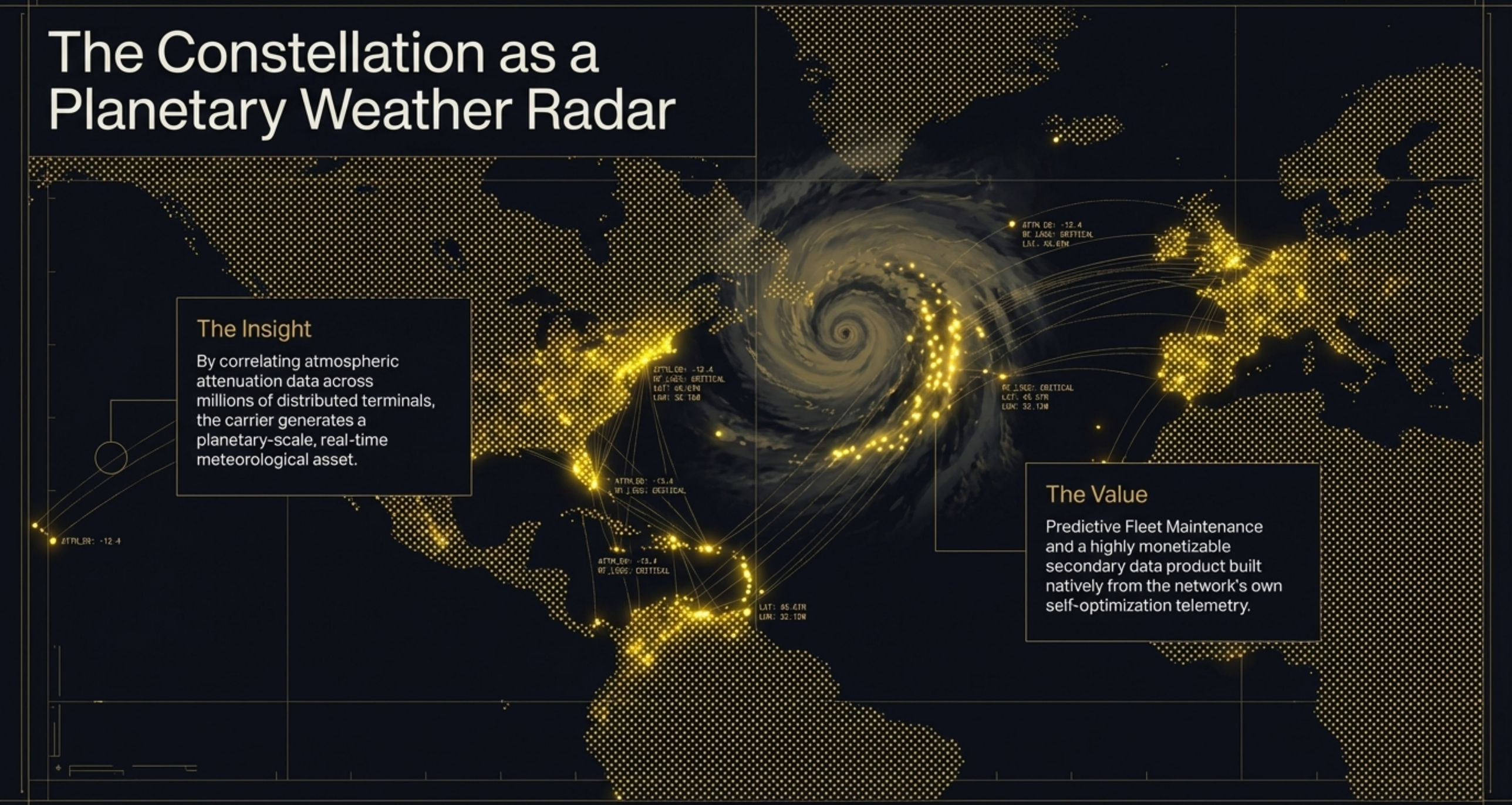


Pillar 3: Boot-Time Resiliency

Actively polls for modem initialization (e.g., `wwan0`), safely handling boot-sequence race conditions before mounting.



The Constellation as a Planetary Weather Radar



The Insight

By correlating atmospheric attenuation data across millions of distributed terminals, the carrier generates a planetary-scale, real-time meteorological asset.

The Value

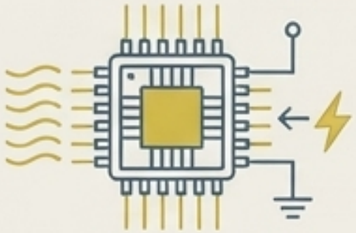
Predictive Fleet Maintenance and a highly monetizable secondary data product built natively from the network's own self-optimization telemetry.

Strategic Outcomes for LEO Carriers

TENSOR NETWORKS

1. Spectral Yield Maximization

Reclaiming up to **20%** of wasted spectral capacity by guaranteeing first-pass delivery and eliminating TCP retransmission storms.



2. Hardware Efficiency

Lower CPU/thermal draw on satellite buses frees critical energy for higher-order onboard processing.

3. Infrastructure Consolidation

High-efficiency Inter-Satellite Links (ISLs) allow carriers to deprecate expensive, remote terrestrial ground stations.

4. Enterprise SLAs

Enables guaranteed, zero-jitter network slices **slices** required for **high-frequency trading** and defense operations (e.g., **Starshield**).

The Mathematically Synchronized Supercomputer

By integrating eBPF-driven execution with Keplerian physics, PTCP successfully transforms a reactive, chaotic orbital mesh into a deterministic planetary substrate. We have decoupled the user experience from the volatility of the physical universe.

TENSOR™ NETWORKS