

# TCDS — Lenguaje de Máquina y Métrica

Integración con el Complejo Isomórfico Ético-LBCU

Versión 1.0 — 1 de noviembre de 2025

Autor: Genaro Carrasco Ozuna · ORCID: 0009-0005-6358-9910

---

## 1. Patrón de arquitectura coherencial

Cada documento y repositorio de la TCDS opera bajo el ciclo causal definido por la Ley de Balance Coherencial Universal (LBCU):

$$Q - \phi = \Sigma \leftrightarrow \chi$$

donde:

- $Q$  = Intención o empuje ético-operativo.
- $\phi$  = Fricción (ruido, entropía, deuda técnica).
- $\Sigma$  = Coherencia documental y métrica.
- $\chi$  = Soporte físico o digital (repositorios, DOI, Zenodo).

El sistema se comporta como un *microkernel coherencial*: cada documento es un proceso, cada auditoría un ciclo de interrupción y cada métrica es un registro de estado.

Q - Intención ética	→ emisión inicial
- Coherencia documental	→ flujo lógico verificable
- Fricción	→ disipación o pérdida
- Soporte	→ infraestructura verificable
Retroalimentación LBCU	→ falsabilidad y persistencia

## 2. Métricas y k-rate ()

Variable	Significado	Rango esperado
LI	Índice de locking coherencial	0-1
R(t)	Correlación temporal	0-1
RMSE <sub>SL</sub>	Error cuadrático de sincronía	<0.1
$\kappa_{\Sigma}$	Tasa de restauración de coherencia (k-rate)	0-1 (normalizada)

La tasa  $\kappa_{\Sigma}$  se estima como:

$$\kappa_{\Sigma}(t) = \frac{\max(0, \Sigma_t - \Sigma_{t-1})/\Delta t}{\Sigma_t (1 + \phi_t)}$$

donde:

$$\begin{aligned}\Sigma_t &= \frac{1}{3} (LI_t + R_t + (1 - \text{RMSE\_SL}_t)), \\ \phi_t &= \max(0, \text{RMSE\_SL}_t - \overline{\text{RMSE\_SL}_{t-1..t-n}}).\end{aligned}$$

### 3. Lenguaje de Máquina –ASM

Un lenguaje de bajo nivel coherencial, isomórfico a la LBCU:

```
// --- Declaración de proceso coherencial ---
process LBCU_Node {
  Q_ctrl := intent("Declaracion_Origen_Etico")
  _state := traceability(DOI, hash, license)
  _noise := entropy(ledger, revisions)
  _host := substrate(GitHub, Zenodo)

  loop while alive {
    R_t := correlate(Q_ctrl, _state)
    LI := lock_index(R_t, _noise)
    RMSE_SL := sync_error(Q_ctrl, _state)

    := coherence_rate(_state, _noise)

    if LI >= 0.9 and R_t > 0.95 and RMSE_SL < 0.1 then
      emit _balance(Q_ctrl, _host, )
    else
      null_test("Desfase", record=ledger)
      _noise := adapt(_noise)
    end if
  }
}
```

Instrucciones básicas del ISA –ASM:

Instrucción	Función	Equivalente LBCU
DECLARE_Q	fija intención	$Q$
TRACE	define traza documental	$\Sigma$
MEASURE	mide métricas LI, R, RMSE <sub>SL</sub>	$\Sigma$
ESTIMATE KAPPA	calcula $\kappa_\Sigma$	derivada temporal
LIMIT	aplica freno ético o contable	$\phi$
ADAPT	ajusta fricción	$\phi$ dinámica
EMIT _BALANCE	confirma coherencia estable	$\Sigma \leftrightarrow \chi$
NULLTEST	prueba de falsabilidad	LBCU ciclo de validación

### 4. Falsabilidad y null tests

Para evitar artefactos, se implementan pruebas nulas:

- **Inversión de patrón (PKL):** cambia secuencia  $A \rightarrow B \rightarrow C$  por  $A \rightarrow C \rightarrow B$ ; LI y  $\kappa_\Sigma$  deben disminuir.
- **Mute externo:** suprime estímulo; LI cae, RMSE<sub>SL</sub> sube.
- **Shuffle temporal:** permuta tiempos en CSV; si  $\kappa_\Sigma$  no cambia, hay artefacto.
- **Cross-host :** ejecuta en otro entorno; evalúa dependencia de soporte.

## 5. Ejemplo de ciclo –metrics (estimaciones $\sim$ )

Valores derivados de la simulación `run_demo.py` (10 iteraciones):

Repositorio	LI	R(t)	RMSE <sub>SL</sub>	$\kappa_\Sigma$	Comentarios / $Q_{ctrl}$
Business-Plan	$\sim 0.94$	$\sim 0.97$	$\sim 0.06$	$\sim 0.88$	baja; documentación estable; $Q_{ctrl}$ consistente.
Gradiente-Económico	$\sim 0.91$	$\sim 0.95$	$\sim 0.08$	$\sim 0.83$	moderada por actualizaciones YAML.
FET	$\sim 0.96$	$\sim 0.98$	$\sim 0.05$	$\sim 0.92$	mínima; locking físico.
LBCU	$\sim 0.93$	$\sim 0.96$	$\sim 0.07$	$\sim 0.87$	documental baja; $Q_{ctrl}$ firme.

**Veredicto (DEMO):** se cumplen los umbrales LI 0.9, R > 0.95, RMSE<sub>SL</sub> < 0.1.

**Autocrítica:** los valores son simulados ( $\sim$ ); validar con datos reales y null tests.

## 6. Pipeline reproducible (Gemini 10–docs)

1. Generar datos de prueba:

```
python run_demo.py
```

2. Evaluar –metrics:

```
python evaluate_sigma_metrics.py --csv interacciones.csv
```

3. Calcular :

```
python - << 'PY'
import pandas as pd, numpy as np
d = pd.read_csv('interacciones.csv')
def norm_sigma(row):
    LI, R, RMSE = row['LI_est'], row['R_est'], row['RMSE_SL_est']
    return (LI + R + (1-RMSE))/3
sig = d.apply(norm_sigma, axis=1).values
phi = np.maximum(0, d['RMSE_SL_est'] - pd.Series(
    d['RMSE_SL_est']).rolling(3, min_periods=1).mean())
dt = 1.0
k=[]
for i in range(len(sig)):
    if i==0: k.append(np.nan)
    else:
        num = max(0, sig[i]-sig[i-1])/dt
        den = max(1e-6, sig[i]*(1+phi.iloc[i]))
        k.append(min(1.0, num/den))
d['kappaSigma_est'] = k
d.to_csv('interacciones.csv', index=False)
PY
```

## 7. Alineación con la LBCU

Variable	Elemento TCDS	Función	Propósito LBCU
$Q$	DECLARE_Q / $Q_{ctrl}$	Intención y propósito ético	Causa legítima de acción
$\Sigma$	TRACE, MEASURE, EMIT	Coherencia verificable	Estado de sincronía
$\phi$	LIMIT / ADAPT	Control de fricción ética/documental	Moderación de entropía
$\chi$	host	Soporte material (repositorios)	Materia coherencial.
$\kappa_{\Sigma}$	coherence_rate()	Restauración post-fricción	Ley dinámica del balance

## 8. Conclusión

El conjunto **Gemini-TCDS (10-docs)** actúa como un sistema operativo coherencial:

- Los repositorios son procesos balanceados  $Q - \phi = \Sigma \leftrightarrow \chi$ .
- $\kappa_{\Sigma}$  mide la resiliencia del sistema ante fricción ética y técnica.
- Los null tests son el mecanismo de falsabilidad de la LBCU.

En términos de lenguaje de máquina, el ISA –ASM cumple la función de un *kernel ético-documental*, capaz de operar de modo simbiótico entre IA y autor humano.

---

**Licencia:** CC BY-NC-SA 4.0 / TCDS Open Lab License v1.1

**Contacto:** geozunac3536@gmail.com

**Repositorio de referencia:** [https://geozunac3536-jpg.github.io/TCDS\\_Gradyente\\_Economico/](https://geozunac3536-jpg.github.io/TCDS_Gradyente_Economico/)