



An inline string ('user.updated') is a landmine. Agents will change it, tests in unread files will pass, and distant systems will quietly break.

Implicit vs. Explicit Boundaries

Tribal Knowledge (Implicit)

Inline strings
(`'user.updated'`)

Assumed JSON shapes

Silent mutations

Repository Contract (Explicit)

Importable constants
(`EVENTS.USER_UPDATED`)

Validated runtime schemas

Loud compile/runtime
failures

The Strongest Contracts Aren't Code

Push **invariants** to the database. Application code can be confidently **deleted** or **bypassed** by an amnesic agent. A **database constraint** cannot be routed around.

Unbypassable Invariants

- NOT NULL constraints
- Foreign key relationships
- Check constraints
- Unique partial indexes

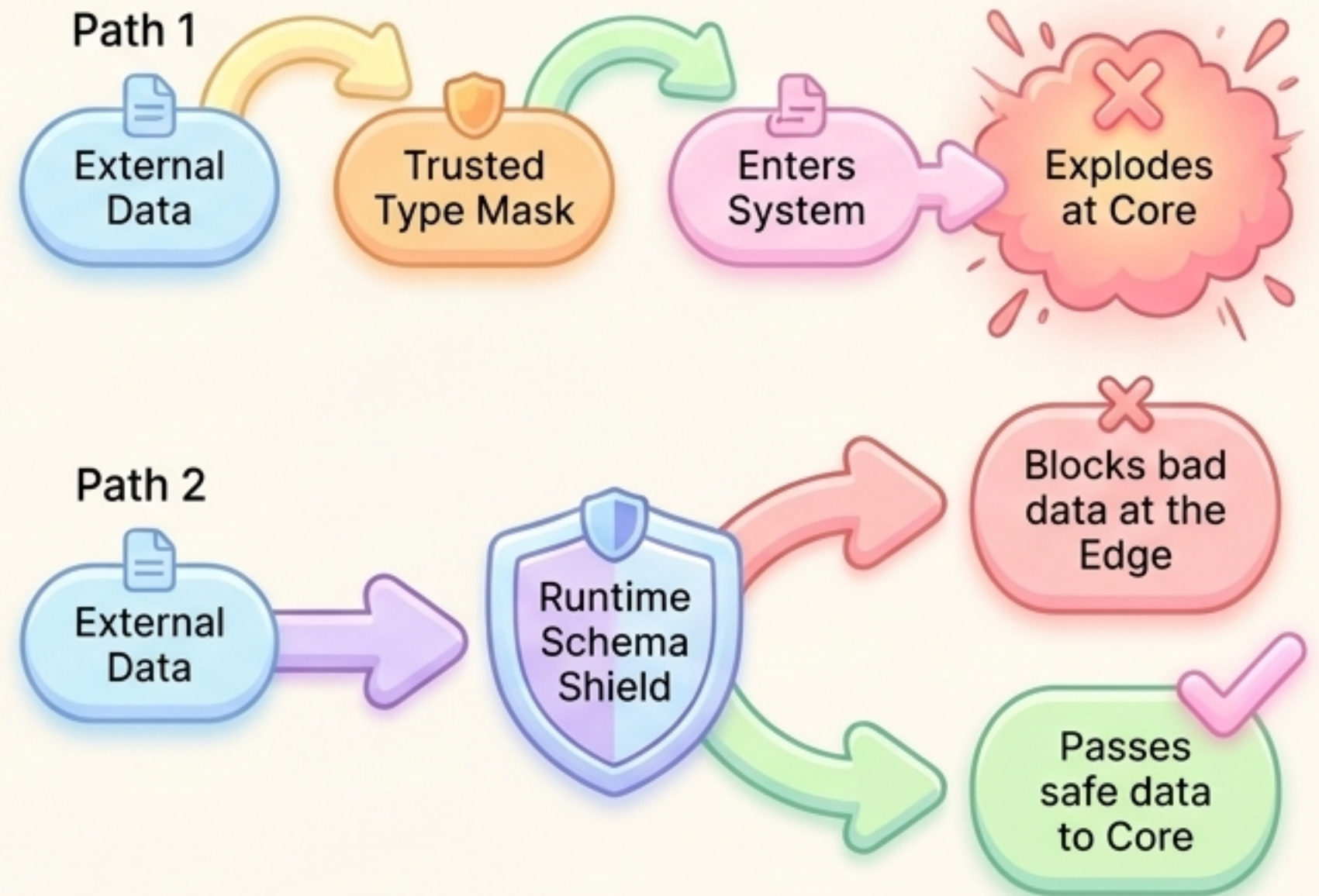


Parse, Don't Cast: The Danger of Fake Promises



Type casting (e.g., ``as Invoice``) is a fake promise that erases runtime validation. An agent takes the path of least resistance, satisfying the type-checker while letting bad data flow inward to explode layers later.

The Boundary Rule





The End of Tribal Knowledge

“Human memory subsidized tribal knowledge. We tolerated implicit rules because human developers remembered to check them.”

The Memory Subsidy is Gone

Agents have amnesia between sessions. They cannot remember to “be careful with events.” If another module, process, or agent depends on a shape, route, or key—it is a contract. Write it down, type it, and test it.

Bounding the Blast Radius



Contracts don't make agents smarter; they make the consequences of their blind spots visible.

Fail Fast: Schemas reject bad data instantly.

Fail Loud: Missing constants won't compile.

Contain Damage: Errors are caught in the diff, before they ship.