

State Synchronisation Strategies for Operational Digital Twins

C. Chen and D. Devi

2023 (synthetic sample)

Synthetic sample paper. Written solely as a sample source for the chi-tragupta documentation's worked examples; all measurements are invented for illustration. Public domain (CC0).

The synchronisation problem

A digital twin is only as trustworthy as its last synchronisation. The engineering question underneath every twin deployment is how state moves from the asset to its representation: how often, at what granularity, and with what guarantee when the link fails.

Push, pull, and event-driven flows

We compare three strategies. Under **periodic pull**, the twin polls the asset on a fixed interval; the implementation is simple and the staleness bound is exactly the interval, but quiet assets are polled as often as busy ones. Under **threshold push**, the asset transmits when a sensed value moves more than a configured delta; bandwidth follows activity, but a mis-set delta hides slow drift entirely. Under **event-driven flows**, domain events – a job start, a valve close – trigger synchronisation; the twin's state history then reads as a process log, which operators in our trials strongly preferred.

In our synthetic benchmark across nine simulated production cells, event-driven flows reduced transmitted volume by 71 percent against one-second periodic pull while keeping worst-case staleness under two seconds during active periods. Threshold push performed comparably on volume but produced the two worst staleness excursions of the study, both during slow thermal drift that never crossed its delta.

Consistency after disconnection

Links fail, and a twin's behaviour during reconnection defines its trustworthiness. We argue for explicit staleness marking: a twin that knows it is stale must say so on every read, rather than serving the last value as if it were current. In our trials, marking

staleness cut operator mistrust incidents roughly in half, because the twin stopped being confidently wrong.

Replaying a backlog after reconnection is a separate decision from marking. Full replay restores the audit trail; last-value-wins restores currency fastest. The hybrid we recommend replays events into history while promoting only the newest value to the live state.

Conclusion

Synchronisation strategy is a first-class design decision, not plumbing. Event-driven flows with explicit staleness marking and hybrid reconnection gave the best operator-trust outcomes in our synthetic trials, and the pattern costs little more than the periodic pull it replaces.