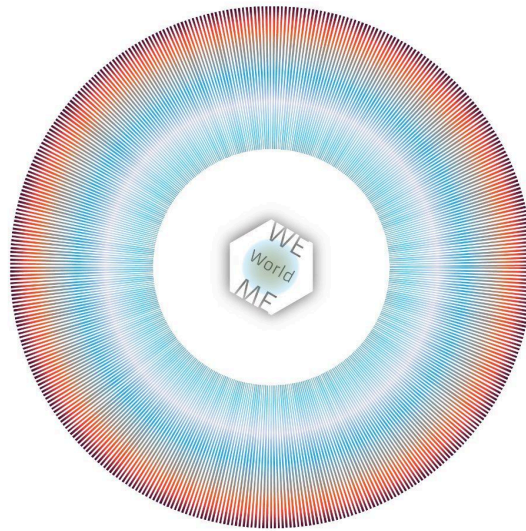




When systems forget, people remember

From Indifference → to Awareness → to Care

MWW Reading Stadsverwarming #9 — Lifecycle Management

Wanneer systemen leren onthouden

Deze Me–We–World Reading is een reflectieve reactie op Machiel Tesser's LinkedIn-blog "*Lifecycle management – Als his story eindelijk history wordt*", waarin hij laat zien waarom systemen die hun eigen geschiedenis niet vasthouden, mensen dwingen om dat geheugen te dragen.

In de eerdere MWW Readings keek ik via **STUART per Me, We en World** naar waar de stadsverwarmingsketen vastloopt: waar veiligheid verdwijnt, vertrouwen breekt en betekenis weglekt.

Deze Reading is anders. En dat is bewust.

Bij **Lifecycle Management** verschuift de kernvraag.

Niet:

Waar gaat het mis?

Maar:

Waarom leert dit systeem structureel niet van wat er al is gebeurd?

Wat Machiel Tesser in deze blog laat zien, is geen nieuw incident, maar een **blijvend patroon**:

bij elke overdracht verdwijnt geschiedenis.

Bewoners beginnen opnieuw. Leveranciers ook. Het systeem leert niets.

Lifecycle management is daarmee **geen volgende optimalisatie**, maar de **structurele oplossing** voor het lek dat in eerdere Readings zichtbaar werd.

Het probleem achter het probleem: een systeem zonder geheugen

Wanneer een warmtenet wordt doorverkocht:

- verdwijnen dossiers,
- vervagen ervaringen,
- wordt context gereduceerd tot losse data.

Niet omdat de informatie er niet is,
maar omdat het systeem haar **niet kan vasthouden**.

Het gevolg:

- bewoners worden dragers van geheugen,
- organisaties blijven reageren in plaats van leren,
- en elk probleem wordt opnieuw een incident.

Zonder geheugen bestaat er geen verbetering.
Alleen herhaling.

Waarom STUART hier anders wordt gelezen

In deze Reading wordt STUART **niet per Me, We en World uitgesplitst**.

Niet omdat die lagen niet relevant zijn,
maar omdat **lifecycle management ze tegelijk raakt**.

Zonder structureel geheugen:

- is er geen **Safety** over tijd,
- geen **Trust** bij overdracht,
- geen **Understanding** die kan groeien,
- geen **Awareness** van patronen,
- geen **Relaxation** voor bewoners én keten,
- en geen **Togetherness** die langer meegaat dan een contract.

STUART verschijnt hier niet als diagnose-instrument, maar als **voorwaarde voor systeemleren over tijd**.

Van black box naar lerend systeem

Het Issuer–Holder–Verifier-model en digitale productpaspoorten veranderen dit fundamenteel.

Niet door meer controle,
maar door **continuïteit**.

- Elke ingreep blijft bestaan
- Elke storing wordt onderdeel van een verhaal
- Elke verbetering bouwt voort op wat er al was

Zo verschuift het systeem van:

- reageren → leren
- incident → geschiedenis
- eigendom → verantwoordelijkheid

Niet alleen technisch, maar ook menselijk.

Wat dit praktisch betekent

Lifecycle management zorgt ervoor dat:

- bewoners niet steeds opnieuw hoeven te bewijzen wat ze ervaren,
- nieuwe leveranciers kunnen voortbouwen i.p.v. opnieuw beginnen,
- toezichthouders zien wat zich werkelijk heeft opgebouwd,
- en de keten leert vóórdát problemen escaleren.

Het systeem krijgt een geheugen.
En daarmee een toekomst.

Tot slot

De kern van deze Reading is eenvoudig:

Systemen die niet kunnen onthouden,
dwingen mensen om het geheugen te dragen.

Lifecycle management herstelt die balans.
Niet door meer regels,
maar door **dat wat ertoe doet te laten blijven bestaan.**

Dat is geen optimalisatie.
Dat is een structurele sprong vooruit.

#MeWeWorld #STUART #LifecycleManagement #SystemThinking #LeanBlockchain
#DigitalProductPassport #SystemMemory #ConsciousSystems #Warmtetransitie
#Ketensamenwerking #GovernanceByDesign #FromAwarenessToCare