RED BEAR: ENTERPRISE DECISION INTELLIGENCE · DATA SHEET · F · REACH: PRESENCE, DELEGATION AND NOTIFICATION

The gate exists on paper. Is anybody behind it?

Routing decides who should decide. Reach decides whether they are there, and what happens when they are not.

KEY POINTS

- Presence is a contract the product consumes, with a freshness and a documented failure position: stale resolves to unknown, never silently to someone else
- Authority moves in three ways that are not the same thing: standing delegation, temporary delegation and automatic cover
- Notification is a ladder, not a channel; acknowledgement is a state, recorded only on an explicit action, and it is the one escalation watches
- Reach analysis leads with the class, never the person: a screen that answers "who is absent" ends the deployment

What it is

Routing decides *who* should decide; this cluster decides *whether they are there*, and what happens when they are not. Book 1 of *Empower, Trust, Verify* names the failure it removes: the unstaffed gate. The gate exists, nobody is behind it, and the decider acts outside authority or does nothing.

Reach establishes whether a decider is reachable now, from declared status, calendar, working hours, time zone, device and on-call rota; moves authority to a cover or a delegate in a recorded way when they are not; delivers the notification by a ladder matched to the class's urgency; records whether a human actually saw it; and escalates tier by tier against a time budget until the decision is made or the class's default applies. It cannot be deferred to a later phase: routing that does not know whether anyone is there is the bottleneck it was meant to remove.

The pain point

A referral routes to someone travelling, asleep, on leave or in surgery, and the queue does not know. Delegation is remembered, not recorded: "Marcus is covering me this week", agreed in a corridor, produces decisions of uncertain provenance. Notification is a firehose or a whisper: everything arrives by the same channel at the same volume, so people mute the channel that carries the crisis. Nobody knows whether it was seen: the platform records that a notification was sent, not that a human read it, so escalation fires on no decision, hours after it should have fired on no acknowledgement.

Escalation is a favour, not a mechanism: referring upward is chosen, usually late, usually apologetically, so the slowest decisions are the ones nobody felt entitled to escalate. And cover is authority without evidence: a deputy decides with the same latitude and no additional scrutiny, so the highest-risk decisions are taken by the person with the least context.

How it works

Presence and availability. The product consumes a presence contract, who, reachable or not, until when, by which channels, and the time it was established, and does not own it. Every assertion carries its timestamp; beyond the class's staleness limit it resolves to unknown, which routes to the holder *and* notifies the cover, so a stale signal cannot silently divert authority. Declaring absence must be easier than not declaring it, so the calendar is read by default.

Three ways authority moves. *Standing delegation* lives in the rights model as a permanent property of the two roles. *Temporary delegation* is declared by the holder for a bounded period, narrower than or equal to the holder's own authority and never wider. *Automatic cover* routes work to a nominated cover when the holder is unavailable, without the holder declaring anything, bounded by the unavailability; as a handling rule the cover's decisions are reviewed one trust band down for the duration, and no dial anywhere is written. Every decision taken under any of the three names both parties and the mechanism, and the cover sees that they are covering, and for whom, on every item.

Notification as a ladder. Delivery follows a channel ladder matched to the class's urgency: quiet channels first, interruptive ones at defined intervals with a maximum. Quiet hours are respected, except for classes declared crisis, which record the break. A notification moves through sent, delivered, acknowledged, decided, and acknowledgement is recorded only on an explicit action, one tap, never inferred from a page view. That state, not the decision, is what escalation watches.

The escalation ladder. Each class has tiers, typically holder, cover, owner, accountable director, each with a time budget. Expiry of a budget without acknowledgement escalates to the next tier and records it. The ladder ends in a decision or the class's timeout default, and an unbounded ladder with no default is refused at configuration. The three intervention verbs govern the clock as in

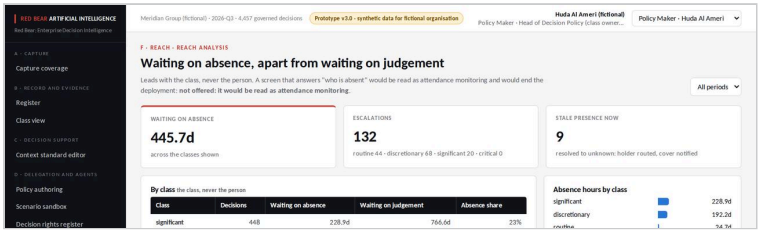
routing: ask pauses it, approve and reject close it, escalation neither pauses nor resets it. Escalation is shown as a ladder with a position, so referring upward is a mechanism rather than a favour.

Reach analysis. Waiting time is attributed to absence separately from judgement: how long a class waited because nobody was there, as distinct from how long it waited because someone was thinking. It leads with the class, never the person, because a view that answers "who is absent" will be read as attendance monitoring and will end the deployment. Patterns, never individuals.

What you get

Capability	What it does	Requirements
Presence and availability	Reachable now, from status, calendar, hours, time zone, device and rota; every assertion timestamped; stale resolves to unknown and notifies the cover	FR-RCH-01
Standing and temporary delegation	Recorded grants, temporary ones bounded in time and never wider than the holder's own authority; both parties named on every decision	FR-RCH-02
Automatic cover	Work routed to the nominated cover without a declaration, bounded by the unavailability, reviewed one band down for the duration	FR-RCH-03
Notification and acknowledgement	A channel ladder by urgency, quiet hours, crisis classes that may break them; sent, delivered, acknowledged, decided	FR-RCH-04
Escalation ladder and service level	Tiers with time budgets per class; escalation on no acknowledgement; ends in a decision or a default; reach analysis by class	FR-RCH-05
Routing it serves	Cover with the same authority and the record naming both; the timeout default it can end in	FR-RTE-01, FR-RTE-02

Screens



S-38 Reach analysis: Reach analysis: where decisions waited on absence rather than on judgement, by class first. Prototype render on synthetic data for fictional entities.

RETURN ON INVESTMENT

The return of this cluster is the return of routing made real. A ninety-day pilot on one decision class requires it, because a gate that nobody is behind produces exactly the failure governance was bought to remove: the decider acts outside authority, or does nothing, and either way the record shows a decision that waited. **Faster decisions at the edge** and **leadership time where it counts** are both measured against target times that only mean something once absence is modelled; reach analysis shows how much of the wait was absence and how much was judgement, which is the split a class owner needs before changing either the ladder or the dial. **Provable accountability** gains the mechanism by which authority moved on every decision taken under cover or delegation. No measure carries a target here; the split between absence and judgement is a baseline the pilot takes before anything changes.

Platform engines

Reach consumes the Presence engine for who is deciding what, now, across people, agents and applications, and the presence contract with its freshness (RBP FR-PR-02), and the Decision Intelligence engine for the delegated-authority table (RBP FR-DI-04). Platform behaviour is specified in the platform functional specification, not here.