**PRD-20 — ECN/ECO Bridge & Change Impact**

**Intent**  
When engineering issues an **Engineering Change Notice (ECN/ECO)**, sync it into Product Store, compute compliance deltas (codes, governed attributes, kits), and drive re-classification + re-decisions with proofs.

**Preconditions**

* ECN/ECO connector (pull or webhook).
* Mapping table: ECN fields → Product Store **governed attributes** (PRD-12) and BOM components (PRD-05).
* Role: Engineering Owner, Compliance Reviewer.

**Flow (happy path)**

1. **Ingest ECN**: receive ECN payload; write **ECN Intake Receipt** (hash + source + time).
2. **Normalize & Link**: map ECN parts to products/variants/BOM components by SKU or crosswalk; show confidence.
3. **Draft Changes**: build a **Change Set** (attribute deltas, component swaps, effective dates) — no live writes yet.
4. **Compliance Preview**: simulate effects:
   * Recompute governed attributes → flag those affecting HS/ECCN/overlays.
   * For kits, cascade “most-restrictive” logic.
   * Show **Impact Deck**: products touched, codes likely to change, affected orders/licenses.
5. **Review & Approve**: Compliance Reviewer approves the Change Set; system creates new **ProductVersions** and queues **PRD-06 impact** jobs.
6. **Execute & Notify**: mint diffs; re-classify where required (new PCVs), re-mint **PCTs**, publish **ImpactDelta** events with WHY text.
7. **Evidence**: produce a **Change Proof Bundle** (ECN Intake Receipt + Change Set + approvals + new PCVs/PCTs + impacted entities list).

**Edge cases**

* ECN refers to unknown SKU → create **linking exception** with suggested crosswalks.
* Conflicting ECNs in window → queue by priority; merge or require manual sequencing.
* Effective date in the past → backfill decisions; clearly tag as “retroactive” and notify.

**Done when**

* Every ECN yields a signed Change Proof Bundle.
* All impacted products/lines processed; decisions reproducible with snapshot\_id + pcv\_id.
* No live changes occur without reviewer approval.