Why Disjointness Matters

Non-deterministic == Bugs

Why Disjointness Matters

Non-deterministic == Bugs

User Role	Resource Type	Action	Owner?	Result
Admin	*	*	*	✓ Allow
*	Public	Write	*	✗ Deny
User	*	Read	Yes	✓ Allow

- Disjointness = No input should match multiple rules
- **Problem Case: (User, Public, Write, true)**
  - Matches row 2: ✗ Deny
  - Matches row 3: ✓ Allow
- **Without disjointness:** Non-deterministic behaviour, hard to reason about

Why Complete and Disjoint Matters

Where do the “requirements” come from

- On the previous slides are examples specifications embodied as tables
- Typically from a product manager in the form of PRD or user stories
  - Often they are incomplete and/or inconsistent
  - Rarely (if ever) define the boundary conditions precisely
- PM may not be a domain expert, developers typically are not:
  - So who’s going to define what the “truth” is for these cases?
  - Can the domain expert understand your code or test cases?