

PolicyManager Contract Comparison Report

This document summarizes the key differences between two Solidity smart contracts for managing insurance policies: `PolicyManager_v1.sol` (legacy version) and `PolicyManager_v2.sol` (updated version).

✓ Overview

Feature	Version 2 (v2)	Version 1 (v1)	Author	Youssef	Mostafa
Policy Type	Crop insurance with seasonal handling	Generic insurance			
Subscription Deadline	✓ Yes	✗ No			
Seasonal Support	✓ Yes (season, start/end, renewal, full vs sub-season)	✗ No			
Treasury Contract	✓ Uses external ITreasury interface	✗ Not used			
Payout Mechanism	Triggered by external engine, only after marking	Direct payout by owner			
Engine Integration	✓ External payout engine contract	✗ Not present			
Subscribers Record	<code>currentSubscribers[]</code> with mapping for last season	<code>subscribers[]</code> only			
Full-Season Exclusivity	✓ Ensures single full-season policy per farmer	✗ Not handled			
Upgradeable Seasons	✓ Resets state for new season	✗ Not supported			
Events	Detailed (Created, Subscribed, Payout, StatusChanged)	Basic (Created, Subscribed, Payout, StatusChanged)			

Feature	Version 2 (v2)	Version 1 (v1)
Validation Modifiers	<code>onlyOwner</code> , <code>validPolicy</code> , <code>onlyPayoutEngine</code>	<code>onlyOwner</code> , <code>validPolicy</code>

Advanced Features in v1

TMSeasonal Policy Management

- Includes: `season`, `seasonStart`, `seasonEnd`, and `subscriptionDeadline`
- Supports resubscription across seasons

Full vs Partial Coverage Logic

- `coversFullSeason` boolean to control exclusivity
- Mapping to prevent sub-policy if full-season is already selected

Integration with External Treasury

- Uses:

```
interface ITreasury {
    function deposit(address farmer) external payable;
}
```

- Sends premiums to a treasury instead of holding contract balance

Controlled Payout Flow

- `payoutEngine` contract must explicitly call `markPolicyAsPayout()`
- Ensures external conditions (e.g. oracle feeds) can trigger payout securely

Simplified Features in v2

- Fixed payout based on `address(this).balance`
- No external calls, oracles, or seasonal dynamics
- Meant for educational/demonstration use

Let me know if you'd like to auto-generate a full `README.md` for your repo or setup GitHub Actions for deployment testing.