# The G-Set
## From a programmer's perspective

```haskell
-- | A simple grow-only set, where the merge function is just a union of the sets
newtype GSet t = GSet (Set.HashSet t)

instance stateBasedCRDTGSet :: (Hashable t) ⇒ StateBasedCRDT (GSet t) where
    merge (GSet a) (GSet b) = GSet $ a ◇ b


query :: forall t. GSet t → Set.HashSet t
query (GSet set) = set


insert :: forall t. Hashable t ⇒ t → GSet t → GSet t
insert value (GSet set) = GSet $ Set.insert value set
```

```
|\_/|
|       |
|  @ @    Woof!
|     ◇
|  _/\____∖___ ((|¯|))
|           __|
| |         -¯-'|
|_|____|  |___|
/_/__/___/____|
```

# The G-Counter

**Can you guess what it does?**