```haskell
-- StateBasedCRDT is a bounded semilattice, i.e. a monoid with additional laws
-- Laws don't appear here, but they will in tests
class Monoid t ≤ StateBasedCRDT t

-- In that case, merge is a synonym for append,
-- but the compiler will only let us use it for CRDTs, not just any semigroup
merge :: forall t. StateBasedCRDT t ⟹ t → t → t
merge = append
```

```haskell
-- Merging multiple states is a simple as calling 'fold'
mergeAll :: forall t. StateBasedCRDT t ⇒ Array t → t
mergeAll = fold

-- fold :: forall f t. Foldable f ⇒ Monoid t → f t → t
-- Other utility functions are ready to use, such as foldMap
```