

## %gall Agent Structure

%gall Agent (app/example-agent-action.hoon)

```
/- graph-store, example-agent-name
/+ default-agent, dbug
|%
+$ versioned-state
  $% state-0
  ==
+$ state-0
  $: [%0 internal-value=@ud]
  ==
+$ card card:agent:gall
--
%- agent:dbug
=| state-0
=* state -
^ agent:gall
=<
|_ =bowl:gall
+* this .
  default ~(. (default-agent this %|) bowl)
  main ~(. +> bowl)
++ on-init :: define initial state & starting values
  ^- (quip card _this)
  `this(internal-value 1)
++ on-save on-save:default :: expose state for saving
++ on-load on-load:default :: load old state, upgrade if necessary
++ on-peek on-peek:default :: expose scry namespace
++ on-poke on-poke:default :: respond to poke
++ on-watch on-watch:default :: handle new subscription
++ on-leave on-leave:default :: handle unsubscribe request
++ on-agent on-agent:default :: handle sign from agent
++ on-arvo on-arvo:default :: handle Arvo signs
++ on-fail on-fail:default :: handle errors
-
:: Helper core
|_ =bowl:gall
++ do-thing ~
--
```

## %gall Agent Structure

### Mark (mar/example-agent-action/action.hoon)

```
/- example-agent-name
|_ act=action:example-agent-action
++ grab
  |%
  ++ noun action:hermes
  --
++ grow
  |%
  ++ noun act
  --
++ grad %noun
--
```

### Structure (sur/example-agent-action.hoon)

```
|%
+$ example-agent-action
$%
  :: All actions are examples here.
  [%view who=@p]
  [%stop-view who=@p]
  [%increment-internal-value-remote who=@p]
  [%increment-internal-value ~]
==
--
```

### Bowl Definition

```
++ bowl :: standard app state
$: $: our=ship :: host
    src=ship :: guest
    dap=term :: agent
    == ::
$: wex=boat :: outgoing subscriptions
    sup=bitt :: incoming subscriptions
    == ::
$: act=@ud :: change number
    eny=@uvJ :: entropy
    now=@da :: current time
    byk=beak :: load source
== ==
```