ADDRMAN

Stochastic address manager

Addresses are organized into buckets:

new	Our node has not successfully connected to them
Tried	Our node knows they are accessible

IsTerrible:

False: if Tried in last minute

True: if
- nTime >10min in future
- nTime older 30 days
- Tried N times, never a success
- N successive failures in the last week

Otherwise, False

Called in:

GetAddr_: if `filter`, we don't include Terrible addrs
↓
called with filter: GETADDR cache, getnodeaddresses without: ASMapHealthCheck

AddSingle: overwrite old entry if it's Terrible

GetAddr:

Pick random addresses in addrman

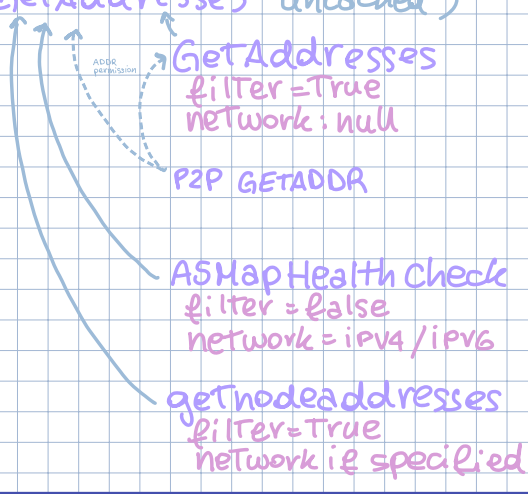
Never pick more than max_addresses, or more than max_pct % of addrman size

Optional: specify network To filter

Optional: filter out Terrible

Called in:

GetAddr_
↑
GetAddr (Addrman)
↑
GetAddresses (CConman, uncached)



AddSingle insert addr in newTable

```
if !rootable
  return false;
set penalty To 0 for self-announc
if addr in addrman
  //update ntime
  currently_online = ntime in past 24h
  update_interval = { online: 1h
                    { not online: 24h
  update if interval expired
  //update service flags
  //STOP updating if our ntime
    is fresher than addr
  //STOP updating if addr in tried
  //STOP updating if too many addr
    already
  // inserted N times: 1/2^N chance
     of inserting again
else:
  Create()
  ntime = max(0, addr.ntime - penalty)
//Determine bucket and position to insert
  new entry
if (bucket, pos) not same addr:
  if (bucket, pos) has addr:
    if addr isTerrible or
      addr present many times and
      new addr is not
      //overwrite existing entry
  if (bucket, pos) can be overwritten:
    ClearNew()
    //update addr ref counter (+=1)
```

Called in:

AddSingle (AddrmanImpl)
↑
Add_ (AddrmanImpl)
↑
Add (AddrmanImpl)
↑
Add (Addrman)

- ThreadDNSAddressSeed (DNS seed)
- ThreadOpenConnections (Fixed seeds)
- P2P ADDR msg (net_processing.cpp)
- RPC addpeeraddress

```
    //insert
  else:
    if ref_count==0:
      Delete(nId)
```

Good (mark an address as accessible and attempt to move it to the Tried Table)

```
pinfo = find addr

//update info

m_last_success = time
m_last_try = time
nAttempts = 0

//don't update ntime→would
  leak info about currently
  connected peers

//if already in Tried, don't
  do anything else

//if not in new: ?!
  (return false, assume 4
  debugging)

// get Tried bucket pos

// if pos full and test
  before evict: log,
  return false

// otherwise

MakeTried ( )

return true;
```

Called in:

- Good_ (AddrManImpl)
  - TestBeforeEvict = false: ResolveCollisions_ (AddrManImpl)
  - TestBeforeEvict = true: Good (AddrManImpl)
    - Good (AddrMan)
      - RPC addpeeraddress
      - NET_proc VERSION msg
        means we connected to a new peer. if it's outbound peer, we call good to update addrman
      - ThreadOpenConnections
        Always running thread, decides which connections to open. If decides to open FEELER connection (⇒ check if peer is alive). IT checks the Tried Table collisions and returns one colliding address. If we are already connected to it, we mark it as Good ().

Attempt

```
//Find addr

//If found, update m_last_try

//If last count attempt < last_good,
  set last count attempt to time,
  nAttempts++;
```

IDK what is going on!! ☺

Called in:

- Attempt_
  - Attempt
    - Attempt
      - ConnectNode — if failure, mark attempt