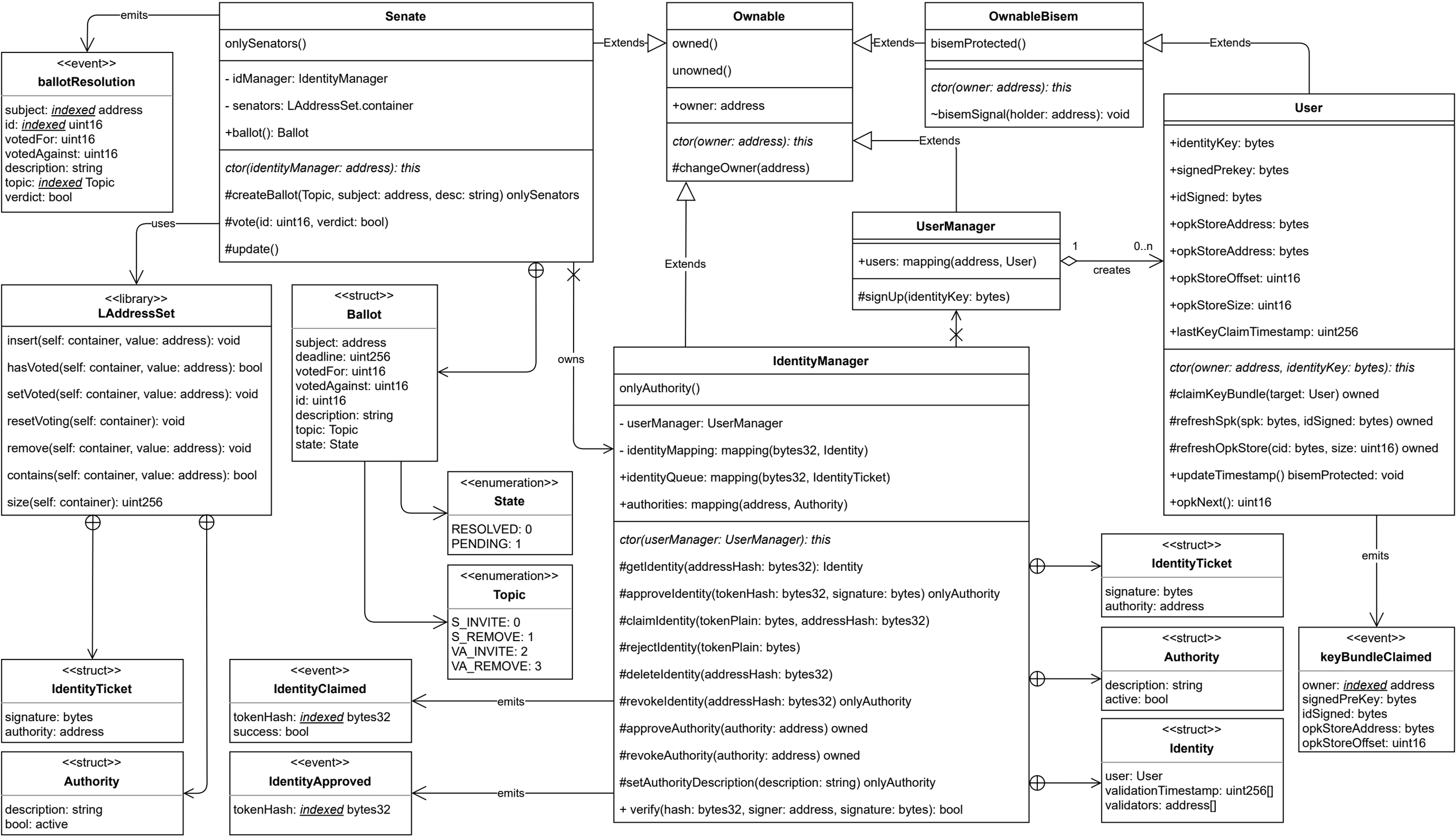
emits
Senate
onlySenators()
- idManager: IdentityManager
- senators: LAddressSet.container
+ballot(): Ballot
ctor(identityManager: address): this
#createBallot(Topic, subject: address, desc: string) onlySenators
#vote(id: uint16, verdict: bool)
#update()
Extends
Ownable
owned()
unowned()
+owner: address
ctor(owner: address): this
#changeOwner(address)
Extends
OwnableBisem
bisemProtected()
ctor(owner: address): this
~bisemSignal(holder: address): void
Extends
<<event>>
ballotResolution
subject: indexed address
id: indexed uint16
votedFor: uint16
votedAgainst: uint16
description: string
topic: indexed Topic
verdict: bool
uses
Extends
User
+identityKey: bytes
+signedPrekey: bytes
+idSigned: bytes
+opkStoreAddress: bytes
+opkStoreAddress: bytes
+opkStoreOffset: uint16
+opkStoreSize: uint16
+lastKeyClaimTimestamp: uint256
ctor(owner: address, identityKey: bytes): this
#claimKeyBundle(target: User) owned
#refreshSpk(spk: bytes, idSigned: bytes) owned
#refreshOpkStore(cid: bytes, size: uint16) owned
+updateTimestamp() bisemProtected: void
+opkNext(): uint16
Extends
UserManager
+users: mapping(address, User)
#signUp(identityKey: bytes)
1
0..n
creates
<<library>>
LAddressSet
insert(self: container, value: address): void
hasVoted(self: container, value: address): bool
setVoted(self: container, value: address): void
resetVoting(self: container): void
remove(self: container, value: address): void
contains(self: container, value: address): bool
size(self: container): uint256
<<struct>>
Ballot
subject: address
deadline: uint256
votedFor: uint16
votedAgainst: uint16
id: uint16
description: string
topic: Topic
state: State
owns
IdentityManager
onlyAuthority()
- userManager: UserManager
- identityMapping: mapping(bytes32, Identity)
+identityQueue: mapping(bytes32, IdentityTicket)
+authorities: mapping(address, Authority)
ctor(userManager: UserManager): this
#getIdentity(addressHash: bytes32): Identity
#approveIdentity(tokenHash: bytes32, signature: bytes) onlyAuthority
#claimIdentity(tokenPlain: bytes, addressHash: bytes32)
#rejectIdentity(tokenPlain: bytes)
#deleteIdentity(addressHash: bytes32)
#revokeIdentity(addressHash: bytes32) onlyAuthority
#approveAuthority(authority: address) owned
#revokeAuthority(authority: address) owned
#setAuthorityDescription(description: string) onlyAuthority
+ verify(hash: bytes32, signer: address, signature: bytes): bool
<<enumeration>>
State
RESOLVED: 0
PENDING: 1
<<enumeration>>
Topic
S_INVITE: 0
S_REMOVE: 1
VA_INVITE: 2
VA_REMOVE: 3
<<struct>>
IdentityTicket
signature: bytes
authority: address
<<struct>>
Authority
description: string
active: bool
<<struct>>
Identity
user: User
validationTimestamp: uint256[]
validators: address[]
emits
<<event>>
keyBundleClaimed
owner: indexed address
signedPreKey: bytes
idSigned: bytes
opkStoreAddress: bytes
opkStoreOffset: uint16
<<struct>>
IdentityTicket
signature: bytes
authority: address
<<struct>>
Authority
description: string
bool: active
<<event>>
IdentityClaimed
tokenHash: indexed bytes32
success: bool
emits
<<event>>
IdentityApproved
tokenHash: indexed bytes32
emits