Basic idea

For an average person, the finality of crypto is perhaps one of the more scary things. If you lose your account keys, your money is gone forever. Similarly, if you mistakenly send money to a wrong address, there is no way to get it back. Yes, it is one's personal responsibility to protect themselves against such woes - but people are people, and everybody makes mistakes.

There are some solutions that try to prevent such problems at a wallet level, but there could be a mechanism that can prevent them at the base protocol level as well. Here is how it could work:

1. You can submit a claim against any account and provide a new public key as a part of the claim.
2. If your claim is successful, the account is destroyed, and all money (and other data) stored in the account is moved to the new key you provided.

This idea is not intended for any specific blockchain - it's more theoretical at this point.

More detailed explanation

To prevent the above mechanism from being abused, we need to put some safeguards in place:

1. When you submit a claim, you must "stake" some funds together with it. The required funds could be proportional to the account's balance (e.g. 5% - 10%).

a. If your claim is successful, these funds are returned to you,

b. But if your claim fails, you lose the staked funds (they are "burned").

1. The claim is recorded in the blockchain - so, everyone can see which accounts have claims submitted against them.
2. If a long time (e.g. 12 months) passes after a claim is submitted against an account, and the account still shows no activity, the claim succeeds.

a. But, if the account's owner executes a transaction against it (something that requires usage of the account's private key), the claim fails immediately.

Example

Let's say I had an equivalent of $10,000 in an account and I lost the key. I would submit a claim against the account and stake $500 (5% of balance) together with it. Since nobody else has the key I lost, nobody would be able to cancel my claim. My claim would mature in 12 months, and I would receive $10,500 under the new key I provided with the claim. Yes, it would take me a year to get the money back - but it's much better than losing it forever.

Just as I can submit a claim against my account, so can anybody else - but they are unlikely to do so. This is because I can very easily cancel their claim against my account. All I need to do is execute a transaction that proves that I still have the private key, and they'd be out of $500. So, unless you are absolutely sure that an account's owner is not able to use their private key, you would not want to submit claims against their account.

Alternative to rent

The above mechanism can be adjusted to achieve goals similar to "storage rent". Specifically:

1. Remove accounts that users forget/stop caring about from the sate.
2. Impose small costs on accounts that consume state storage.

To achieve the first goal, no significant changes to the "claiming" mechanism are needed. Since anyone can submit a claim against any account, people could monitor accounts and submit claims against accounts that they believe are stale.

For example, if an account hasn't had any activity for months and has a very low balance (e.g. within 10x of current average transaction fee), it is very likely that the owner has abandoned the account. If I'm right, I could make a little money by submitting a claim against this account. If I'm wrong, I would lose an amount roughly equivalent to a transaction fee.

We can incentivize this behavior more explicitly like so:

If an account hasn't had any activity for extended period of time (e.g. 3 months), and if the balance held in the account is lower than 10x (or 20x etc.) of the average transaction fee, a claim submitted against the account succeeds immediately. (no need to wait for a year for a claim to mature).

The average fee can be calculated as an average over the last 6 months to reduce volatility. It also can be calculated on per-byte basis, so that accounts consuming more storage would have to hold more funds to avoid being claimed by others. This

would also naturally impose higher costs on accounts that consume more storage (goal #2

).