TLDR

: We propose an imperfect availability scheme for large data. It is intended as a stopgap for TrueBit-like computational markets until fully decentralised availability solutions for large data are ready.

Construction

Whenever a task giver supplies a TrueBit task with large data D

(large program or large inputs) the task giver does the following:

- Puts D

on IPFS, referenced by its hash H(D)

- Requests archive.org

to archive ipfs.io/[H(D)]

- Produces a short Oraclize proof P

that D

is available on some https://archive.org

URL U

- Supplies H(D)

, P

, U

onchain for verifiers

Discussion

Verifiers will be able to retrieve D

if:

1. ipfs.io

is a faithful IPFS gateway

1. archive.org

is a faithful public and permanent historical mirror

For extra reliability more than one IPFS gateway (or other service that maps files to hashes) can be used simultaneously, as well as more than one storage provider.

Various bells and whistles can be used, e.g. SNARKs to remove the need for ipfs.io

or to encrypt H(D)

and U

for stealthiness.