Ecash e Bitcoin

Cos'è?

- Crittografia per aumentare privacy in sistemi custodial
- NON è un layer
- Adattabile a Bitcoin (e shitcoin)
- Adattabile a L1 ed L2
- Ha strani trade off

Trade off:

- I token sono self custodial
- Rischio di controparte

Perchè ecash?

Blind signatures for untraceable payments (David Chaum, 1983)

BLIND SIGNATURES FOR UNTRACEABLE PAYMENTS

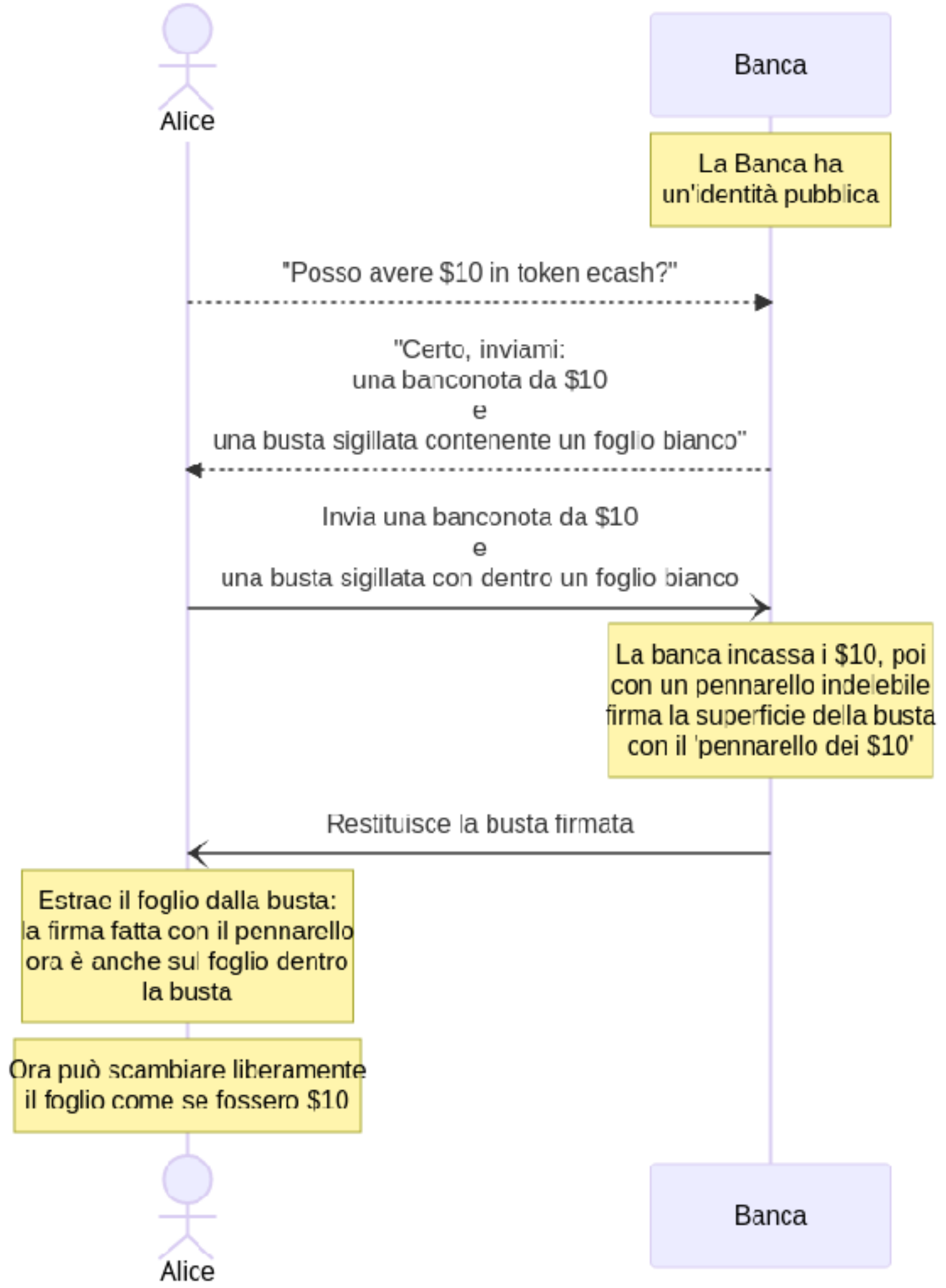
David Chaum

Department of Computer Science
University of California
Santa Barbara, CA

INTRODUCTION

Automation of the way we pay for goods and services is already underway, as can be seen by the variety and growth of electronic banking services available to consumers. The ultimate structure of the new electronic payments system may have a substantial impact on personal privacy as well as on the nature and extent of criminal use of payments. Ideally a new payments system should address both of these seemingly conflicting sets of concerns.

Chaumian ecash without RSA:
I've always seen Chaum's anonymous ecash system described in terms of RSA. RSA has this ungainly patent which probably will be around for quite some time, yet the Diffie-Hellman patent expires pretty soon. With that motivation, here's a Chaumian anonymous ecash protocol based on Diffie-Hellman.
(David Wagner, 1996)


Alice
Banca
La Banca ha un'identità pubblica
"Posso avere $10 in token ecash?"
"Certo, inviami:
una banconota da $10
e
una busta sigillata contenente un foglio bianco"
Invia una banconota da $10
e
una busta sigillata con dentro un foglio bianco
La banca incassa i $10, poi con un pennarello indelebile firma la superficie della busta con il 'pennarello dei $10'
Restituisce la busta firmata
Estrae il foglio dalla busta: la firma fatta con il pennarello ora è anche sul foglio dentro la busta
Ora può scambiare liberamente il foglio come se fossero $10
Alice
Banca

Non ha funzionato perchè...

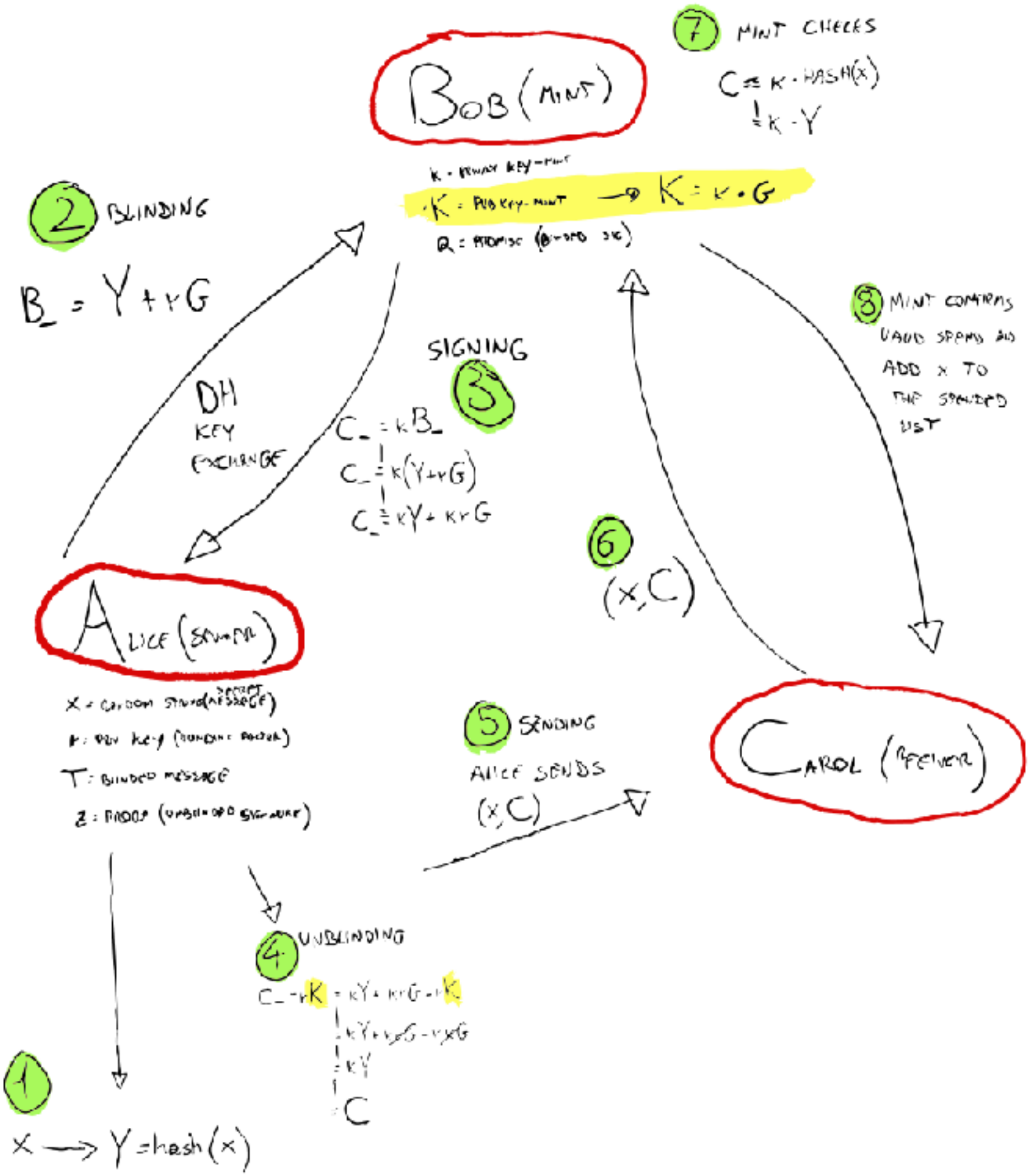
- Cold start e nessun incentivo economico (?)
- Consumatori poco interessati (?)
- Brevetti (?)
- Chaum = pessimo businessman (?)
- Problemi di scalabilità (?)

Progetti ecash

- Shitcoin: DigiCash (D), GNU Taler (A)
- Bitcoin: Cashu (A), Fedimint (A)

Cashu /vs/ Fedimint

- Single sig /vs/ federazione
- LN /vs/ OC+LN
- Custodial /vs/ Custodial (trust minimized)
- 1 Nodo LN = 1 Mint /vs/ Guardiani + LN Gateway


BOB (MINT)
7 MINT CHECKS
C ≟ k · HASH(x)
≟ k · Y
k = PRIVATE KEY-MINT
K = PUBKEY-MINT → K = k · G
Q = PROMISE (B-DHD SIG)
2 BLINDING
B_ = Y + rG
DH KEY EXCHANGE
SIGNING 3
C_ = kB_
C_ ≟ k(Y+rG)
C_ ≟ kY + krG
8 MINT CONFIRMS VALID SPEND AND ADD x TO THE SPENDED LIST
6
(x, C)
ALICE (SENDER)
x = CALCULATE SECRET(MESSAGE)
r = PRIV KEY (RANDOM FACTOR)
T = BLINDED MESSAGE
Z = PROOF (UNBLINDED SIGNATURE)
5 SENDING
ALICE SENDS
(x, C)
CAROL (RECEIVER)
4 UNBLINDING
C_ - rK = kY + krG - rK
= kY + krG - rkG
= kY
= C
1
x → Y = hash(x)

Vantaggi di ecash:

- Storia dei token inesistente
- Non è possibile fare amount correlation (al contrario degli UTXO) [cashu]
- Metodo di transazione anonimo (se utente richiede token anonimamente [aka Tor])
- Token scambiabili offband, in header HTTP, messaggi, radio
- E' possibile inviare e ricevere completamente offline in sicurezza
- Scripting es. P2PK, P2SH(?), HTLC(?)
- Proof of Liability
- Transazioni inter-mint con LN
- Supporto a stablesats

Svantaggi di ecash

- Rischio di controparte - solvenza della mint
- Money printing della mint
- Mint è di fatto un hot wallet single sig
- Mint deve essere sempre online
- Garanzia di non double spending solo facendo query alla mint
- Possibili problemi di throughput della mint
- Rischi normativi (cashu sta implementando specifiche per Auth)

Come usare ecash: Cashu

- Implementazioni client e server su cashu.space
- è tecnologia in alpha e custodial, siate cauti!!

Come usare ecash: Fedimint

- fedimint.org
- Client Fedi

Per accedere a mint e federazioni:

- bitcoinmints.org
- cashumints.space