$402 PROTOCOL SPECIFICATION

> Tokenized Attention Economy with Social Scaling

Version: 3.0.0 **Status**: Living Document **Reference Implementation**: PATH402.com

OVERVIEW

$402 is a protocol for tokenized attention markets. Every participant mints their own token, creating a market for their time and content. The network scales socially through genuine relationships, not viral mechanics.

The Progression:

1. **Content Tokenization** (v1-v2): Turn URL paths into shareholder businesses
2. **Personal Tokenization** (v3): Turn individuals into attention markets

CORE PRINCIPLES

1. **Everyone has a token** - Your token is your attention market
2. **Fixed supply** - 1 billion tokens per person (no minting/burning)
3. **Time-based access** - Tokens purchase connection time (1 token = 1 second default)
4. **Creator-controlled float** - You decide how many tokens to sell
5. **Social bootstrapping** - Friends invest in friends

TOKEN ECONOMICS

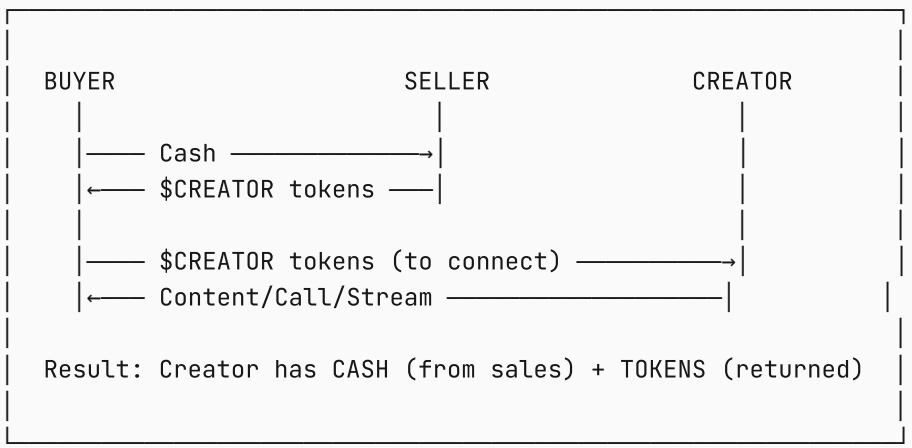
SUPPLY MODEL

```
Per-person supply: 1,000,000,000 tokens (1 billion)
No minting after genesis
No burning - tokens circulate
```

ACCESS PRICING

```
Base rate: 1 token = 1 second of connection
Configurable per creator: 1-100 tokens/second
Multicast: Same rate, split across viewers (or flat per viewer)
```

THE ECONOMIC FLOW



CREATOR FLOAT CONTROL

STRATEGY	TOKEN FLOAT	PRICE	RESULT
Exclusive	Low (hold most tokens)	High	Premium access, fewer connections
Accessible	High (sell freely)	Low	Mass audience, lower per-connection value
Balanced	Medium	Market-driven	Organic price discovery

SOCIAL SCALING

THE FRIEND INVESTMENT MODEL

Users naturally invest in people they care about:

```
I value my friend
       ↓
I buy MORE tokens than I need
       ↓
I stake tokens (earn from their success)
       ↓
I complete KYC (because I trust them)
       ↓
They succeed → I profit + maintain access
       ↓
Network grows through real relationships
```

WHY THIS WORKS

1. **Aligned incentives** - Supporting friends = potential profit
2. **Natural liquidity** - Social circles create baseline markets
3. **Trust bootstrapping** - KYC happens where trust exists
4. **Anti-spam** - Connecting costs tokens (economic friction)
5. **No platform rent** - Direct peer-to-peer settlement

STAKING & DIVIDENDS

STAKING MECHANISM

```
Stake $CREATOR tokens
       ↓
Receive share of creator's revenue
       ↓
Revenue = all token purchases for their content
       ↓
Dividend = (your_stake / total_staked) × revenue × dividend_rate
```

DIVIDEND CLAIMS & KYC

- **Basic access**: No KYC required (permissionless)

- **Dividend claims**: KYC required
- **Identity anchor**: Phone contract, government ID, or trusted verifier
- **Result**: Voluntary identity layer where money flows out

REVENUE SPLIT (DEFAULT)

```
Creator revenue from token sales:
├── 70% → Creator wallet
├── 20% → Staker dividend pool
└── 10% → Protocol treasury ($402 token holders)
```

CONNECTION TYPES

1. DIRECT CALL (1:1)

```
Caller spends: N tokens (for N seconds)
Creator receives: N tokens
Bidirectional: Both parties spend each other's tokens
Net cost: Difference in token values
```

2. MULTICAST STREAM (1:MANY)

```
Creator streams content
Each viewer spends: N tokens/second
Creator receives: N × viewers tokens
Scales to unlimited viewers
```

3. ASYMMETRIC VALUE CALLS

```
$RICHARD worth: 1000 sats/token
$BOB worth: 100 sats/token

Bob calls Richard:
- Bob spends $RICHARD tokens (expensive)
- Richard spends $BOB tokens (cheap)
- Net: Bob pays ~900 sats/second to talk to Richard

Richard calls Bob:
- Same mechanics, reversed
- Net: Richard earns ~900 sats/second
```

CLIENT FEATURES

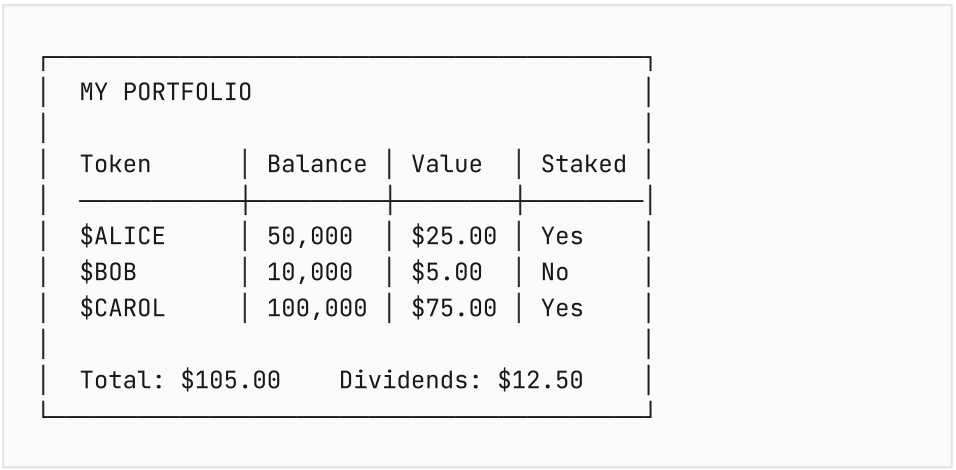
1. IDENTITY & MINTING

```
┌──────────────────────────────────────────┐
│  YOUR TOKEN: $YOURNAME                   │
│  Supply: 1,000,000,000                   │
│  Float: 100,000,000 (10% for sale)       │
│  Floor price: 500 sats                   │
│  Access rate: 1 token/second             │
└──────────────────────────────────────────┘
```

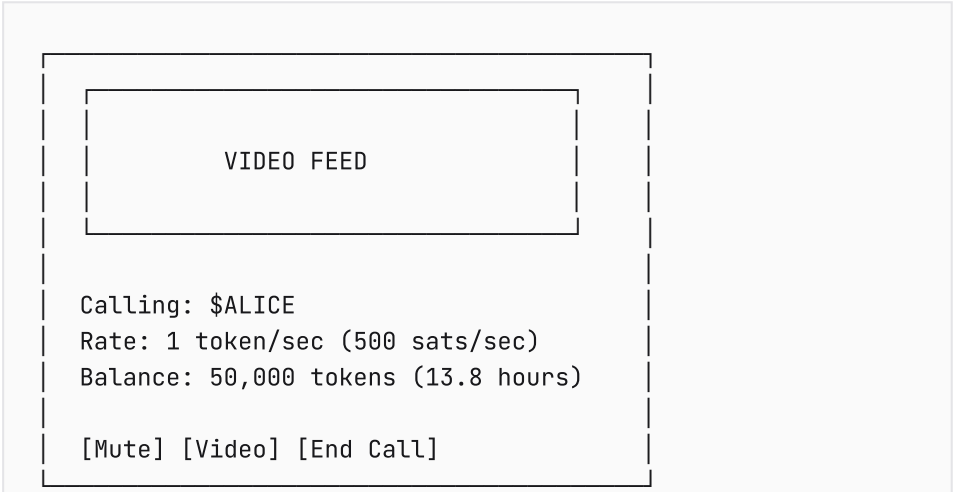
2. DISCOVERY & INDEXING

```
┌──────────────────────────────────────────┐
│  DISCOVER                                │
│  ┌──────────────────────────────┐        │
│  │ 🔍 Search tokens...          │        │
│  └──────────────────────────────┘        │
│                                          │
│  TRENDING          FRIENDS    NEW        │
│  ├─ $SPIELBERG     $ALICE     $JANE      │
│  ├─ $ROGAN         $BOB       $MARK      │
│  └─ $MRBEAST       $CAROL     $SARA      │
└──────────────────────────────────────────┘
```

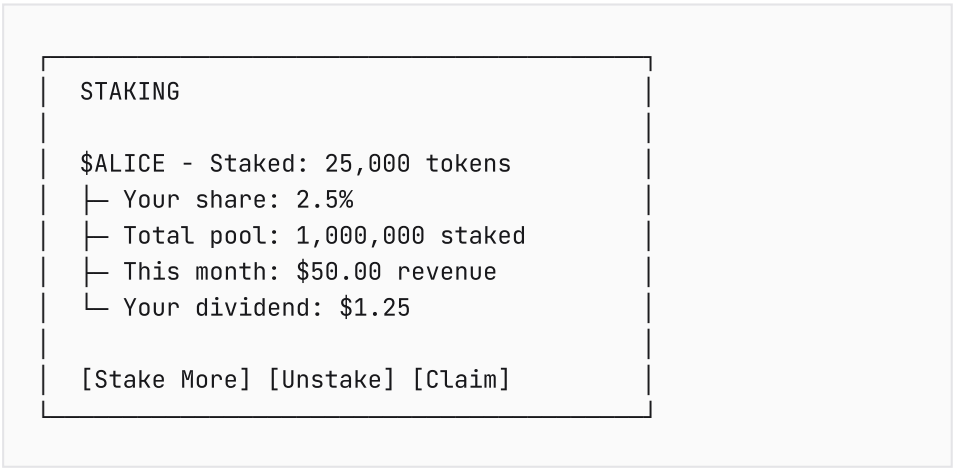
3. PORTFOLIO & RANKINGS



4. VIDEO/AUDIO CALLS



5. STAKING PANEL



6. KYC TAB

```
┌─────────────────────────────────────────┐
│  IDENTITY VERIFICATION                  │
│                                         │
│  Status: ✓ Verified                     │
│  Method: Phone contract                 │
│  Verified: 2026-02-01                   │
│                                         │
│  Linked tokens (can claim dividends):   │
│  ├─ $ALICE ✓                            │
│  ├─ $BOB ✓                              │
│  └─ $CAROL ✓                            │
│                                         │
│  [Add Verification] [Manage]            │
└─────────────────────────────────────────┘
```

PROTOCOL MESSAGES

TOKEN OPERATIONS

```
interface TokenTransfer {
  token_id: string;      // e.g., "$RICHARD"
  from: string;          // sender address
  to: string;            // recipient address
  amount: number;        // tokens transferred
  purpose: 'purchase' | 'access' | 'stake' | 'unstake' | 'divide
}
```

CONNECTION HANDSHAKE

```
interface ConnectionRequest {
  caller_token: string;   // Caller's token ID
  callee_token: string;   // Callee's token ID
  caller_balance: number; // Tokens caller holds of callee
  requested_duration: number; // Seconds requested
  connection_type: 'call' | 'stream' | 'chat';
}

interface ConnectionAccept {
  session_id: string;
  rate: number;           // Tokens per second
  accepted_duration: number;
}
```

STAKING OPERATIONS

```
interface StakeRequest {
  token_id: string;
  amount: number;
  staker: string;
}

interface DividendClaim {
  token_id: string;
  claimant: string;
  kyc_proof: string;      // Reference to KYC verification
  period: string;         // e.g., "2026-02"
}
```

BSV IMPLEMENTATION

WHY BSV ONLY

REQUIREMENT	BSV	ETH/OTHERS
1 token/second micropayments	✓ <0.001¢ fees	× $0.50+ fees
Real-time settlement	✓ Instant	× Block times
Scalability	✓ Unbounded	× Gas limits
Token standard	BSV-20	ERC-20

ON-CHAIN VS OFF-CHAIN

```
On-chain (BSV):
├── Token genesis (1bn supply)
├── Large transfers
├── Staking/unstaking
├── Dividend distributions
└── KYC attestations

Off-chain (Gossip):
├── Connection handshakes
├── Micro-transfers during calls
├── Real-time balance updates
└── Periodic settlement to chain
```

PRICING FORMULA

SQRT_DECAY MODEL

```
price = base_price / sqrt(supply_sold + 1)
```

- **Early buyers**: Lower price (reward early supporters)
- **Later buyers**: Higher price (scarcity premium)
- **Creator benefit**: Early friends get deals, mass market pays more

EXAMPLE

```
Base price: 500 sats
Supply sold: 0 → Price: 500 sats
Supply sold: 99 → Price: 50 sats
Supply sold: 9,999 → Price: 5 sats
Supply sold: 999,999 → Price: 0.5 sats
```

SECURITY CONSIDERATIONS

ANTI-SPAM

- Connection requires tokens (economic cost)
- No tokens = no access
- Spam is expensive

PRIVACY

- Basic access: Pseudonymous (just addresses)
- Dividend claims: KYC required (real identity)
- User controls identity disclosure level

SYBIL RESISTANCE

- Creating fake accounts costs nothing

- But fake accounts have no social graph
- No friends = no token buyers = worthless token
- Social proof required for value

LEGAL COMPLIANCE & LIABILITY

> **CORPORATE REGISTER VS TOKEN OWNERSHIP**
>
> The company has liability and must produce a confirmation statement each year showing an up-to-date register of members. However, Members can trade tokens that act as a claim on their shares without permission.
>
> So while the confirmation statement may say Alice holds Title, she may have already sold her interest to Bob, who has already sold it to Charlie. The confirmation statement says Alice owns the share on the register, but that is still legally compliant, even though the reality is that Charlie holds the asset. As long as each individual pays their own taxes, they are compliant too.

ROADMAP

PHASE 1: CORE PROTOCOL ✓

- [x] Token model defined
- [x] Access mechanics specified
- [x] Economic flows documented

PHASE 2: CLIENT IMPLEMENTATION

- [ ] Video/audio call module
- [ ] Portfolio tracking
- [ ] Discovery/search
- [ ] Staking UI
- [ ] KYC integration

PHASE 3: NETWORK LAUNCH

- [ ] Bootstrap peers
- [ ] Initial token minting
- [ ] Creator onboarding
- [ ] Market maker liquidity

PHASE 4: ECOSYSTEM

- [ ] Mobile apps
- [ ] Browser extension
- [ ] API for third-party apps
- [ ] Creator tools

SUMMARY

$402 creates an attention economy where:

1. **Everyone has a price** - Your token represents your time's market value
2. **Friends invest in friends** - Social relationships create liquidity
3. **Creators control access** - Your float, your rules
4. **Speculators align with supporters** - Staking rewards belief
5. **KYC is voluntary but incentivized** - Need it for dividends
6. **BSV enables micropayments** - 1 token/second is economically viable

The network scales through genuine social investment, not viral growth hacks.

PERSONAL TOKEN MINTING

BSV21 STANDARD

Personal tokens use the BSV21 fungible token standard with $402 extensions:

```
{
  "p": "bsv-21",
  "op": "deploy",
  "tick": "$RICHARD",
  "max": "1000000000",
  "dec": "0",
  "path402": {
    "accessRate": 1,
    "protocol": "path402",
    "version": "1.0.0"
  },
  "metadata": {
    "name": "RICHARD",
    "description": "Access token for Richard",
    "avatar": "https://...",
    "website": "https://..."
  }
}
```

TOKEN PROPERTIES

PROPERTY	VALUE	DESCRIPTION
`supply`	1,000,000,000	Fixed, no further minting
`decimals`	0	Whole tokens only
`accessRate`	1-100	Tokens per second for connections
`burning`	Disabled	Tokens circulate, never destroyed

PROOF OF SERVE

Nodes earn rewards through actual network contribution:

12

ACTION	DESCRIPTION	REWARD WEIGHT
`serve`	Deliver content to requesters	High
`relay`	Forward gossip messages	Medium
`index`	Maintain accurate indexes	Medium
`validate`	Verify transactions	Low

Reward Formula:

```
node_reward = (node_serves / total_network_serves) × daily_rewar
```

WHY NOT PROOF OF WORK?

PROOF OF WORK	PROOF OF SERVE
Rewards hash computation	Rewards actual service
Wastes electricity	Uses real network work
Centralizes to ASICs	Scales with usage
One winner per block	Everyone who serves earns

Version: 3.0.0 Last Updated: 2026-02-05