Programmable Money: Money That Thinks

Module 3: The Trust Problem — Companion Lecture

Prof. Dr. Joerg Osterrieder

Digital Finance — BSc Course

Companion lecture — explores what happens when currency carries its own rules.

Why does your $100 bill not know if it is paying rent or funding crime?

The limitations of "dumb" money:

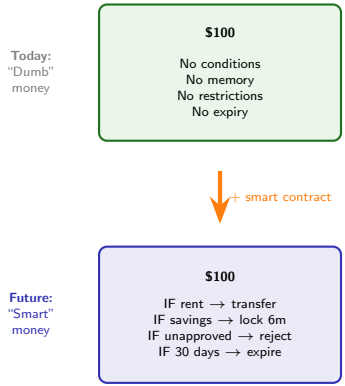
- A banknote carries no conditions — once spent, the sender loses all control
- Cash cannot enforce spending rules, expiry dates, or recipient restrictions
- Wire transfers have no built-in logic — they move value, nothing more
- Governments cannot target stimulus to specific goods or regions

Real-world consequences:

- **Aid leakage:** commonly cited estimates place diversion at 20–30% of aid disbursements (Source: UN OCHA and Transparency International reviews); exact figures are inherently hard to measure
- **Tax fraud:** cash economy enables unreported transactions
- **Welfare misuse:** cash benefits can be spent on anything
- **Corporate expense fraud:** employees misuse company cards

The core limitation:
Money today is a *bearer instrument* — whoever holds it controls it. There is no way to embed rules into the money itself.

Key insight: Physical cash and traditional bank transfers are "dumb" — they carry value but no instructions.



Money has been "dumb" for 5,000 years — programmable money adds conditional logic to value transfer for the first time.

Imagine your salary arriving as programmable tokens that manage your budget automatically

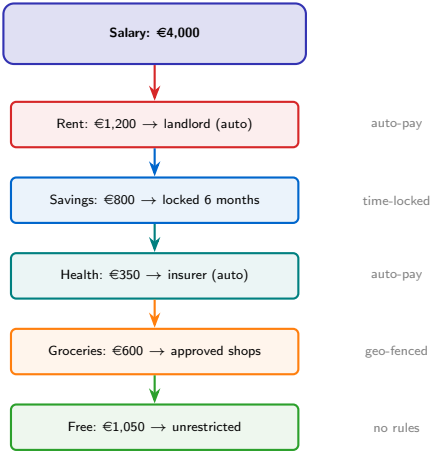
The scenario:
It is the 1st of the month. Your salary of €4,000 arrives — not as a single bank transfer, but as programmable tokens with built-in rules:

- **Rent (30%):** €1,200 auto-transfers to your landlord
- **Savings (20%):** €800 locks in a savings contract for 6 months — you cannot touch it early
- **Health insurance:** €350 auto-pays your insurer
- **Groceries (15%):** €600 tagged for approved supermarkets only
- **Discretionary (25%):** €1,050 unrestricted — spend on anything

What changes:

- No missed rent payments — the money moves itself
- No willpower needed for saving — the lock is in the code
- Budgeting happens *in the money*, not in a separate app

The question: Helpful automation or loss of freedom?

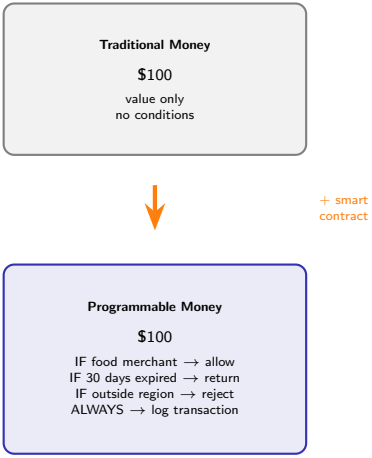


Programmable salary automates budgeting at the money layer — but the same rules that help you save could also restrict your freedom if set by someone else.

What is programmable money — and how do IF-THEN rules get embedded in tokens?

Definition: Programmable Money

Digital currency with embedded smart contract logic that enforces conditions on how, when, where, and by whom the money can be spent. The rules execute automatically — no intermediary needed.



Types of programmable rules:

- **Time conditions:** expires after 30 days, unlocks on a date
- **Spending restrictions:** valid only at approved merchants
- **Geographic limits:** spendable only within a city or region
- **Conditional release:** released when delivery is confirmed
- **Non-transferable:** cannot be sent to another person
- **Audit trail:** every transaction recorded permanently

The formula:

`Digital currency + smart contract = programmable money`

(A **smart contract** is a program stored on a blockchain that executes automatically when conditions are met — like a vending machine for financial agreements.)

Key insight: Traditional money separates value from rules. Programmable money merges them.— the rules travel with the value.

How did Shenzhen distribute 10 million RMB in programmable digital red envelopes?

China's e-CNY pilot (Shenzhen, October 2020):

- Shenzhen government distributed 10 million RMB ($1.5 million) in digital red envelopes
- 50,000 citizens won 200 RMB each via lottery
- Downloaded the e-CNY wallet app to receive tokens

Programmable rules embedded in the tokens:

- **Expiry:** must be spent within 7 days or tokens vanish
- **Merchant restriction:** valid only at 3,389 approved retailers
- **Non-transferable:** cannot be sent to another person
- **Geography:** spendable only in Luohu district, Shenzhen

Results:

- 88% of recipients spent their tokens
- Stimulated $2.6 million in additional spending
- Government tracked every transaction in real time

Key insight: This was stimulus that expired — forcing spending, targeting local businesses, and generating a complete audit trail.

Feature	e-CNY Red Envelope
Amount	200 RMB per person
Recipients	50,000 citizens
Total distributed	10 million RMB
Valid period	7 days
Merchants	3,389 approved
Geography	Luohu district only
Transferable	No
Redemption rate	88%
Tracking	Full real-time audit

Comparison with traditional stimulus:
Cash stimulus cheques in the US had no expiry, no spending restrictions, and no real-time tracking. Studies estimate 33% was saved, not spent.

The trade-off: Higher spending rate, but complete government surveillance of every transaction.

Shenzhen's digital red envelopes showed that programmable money can achieve 88% spending rates — but at the cost of full transaction surveillance.

Worked example: $10M in programmable disaster relief — from 30% leakage to under 5%

Scenario: Earthquake in Southeast Asia. UN distributes $10 million in relief funds to 50,000 affected households.

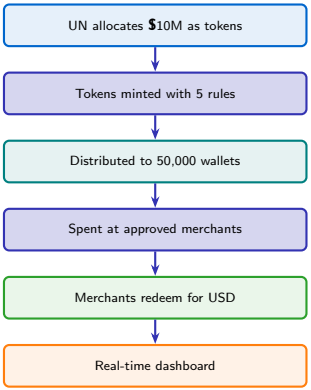
Traditional cash distribution:

- Cash transported to distribution points
- Recipients sign paper receipts
- Estimated leakage: 30% (diversion, corruption, theft)
- Audit happens months later via sampling
- Net reaching victims: $7 million

Programmable token distribution:

- $200 per household as programmable tokens
- **Rule 1:** food and medicine purchases only
- **Rule 2:** expires after 30 days (prevents hoarding)
- **Rule 3:** geo-fenced to affected region
- **Rule 4:** real-time audit trail for every transaction
- **Rule 5:** non-transferable (prevents resale)

Result (hypothetical scenario): leakage modelled at under 5%.
Net reaching victims: ∼$9.5 million.



	Cash	Tokens
Budget	$10M	$10M
Leakage	30%	<5%
Reaches victims	$7M	$9.5M
Audit	Months	Real-time

Leakage figures modelled from UN OCHA / Transparency International aid-diversion ranges.

Programmable relief tokens can save $2.5M per $10M distributed by eliminating diversion — but recipients lose the freedom to spend as they choose.

The dark side: when programmable money becomes a tool of control and surveillance

Government surveillance:

- Every transaction recorded and traceable
- Government sees what you buy, where, and when
- No possibility of anonymous transactions
- Spending patterns reveal political views, health, relationships

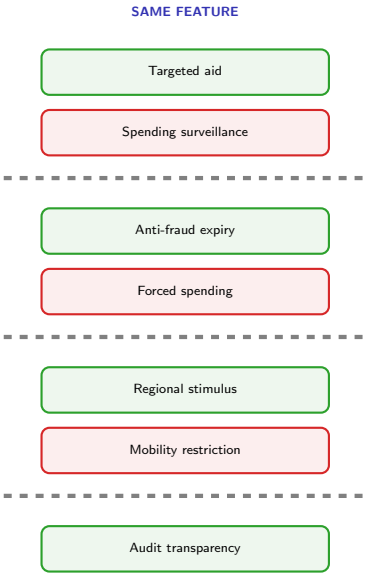
Financial coercion:

- **Forced spending:** expiry dates mean you must spend or lose value
- **Geo-fencing:** restricts mobility and cross-border transactions
- **Merchant restrictions:** government decides what you can buy
- **Negative interest:** programmed value decay to force spending

Social credit integration:

- Low social score = restricted spending categories
- Dissidents could have money programmed to exclude transport
- Tax debts could auto-deduct from your wallet
- Programmable money + social credit = unprecedented control

Warning: The same features that make disaster relief efficient also make financial authoritarianism possible.



Where is programmable money being deployed — from central banks to corporate payroll?

Central Bank Digital Currencies (CBDCs):

- **China e-CNY:** most advanced, 260M+ wallets, pilot in 25+ cities
- **EU digital euro:** ECB preparation phase, launch target 2027–2028
- **Bahamas Sand Dollar:** live since 2020 (first CBDC)
- **Nigeria eNaira:** live since 2021, limited adoption
- 130+ countries exploring CBDCs (BIS survey)

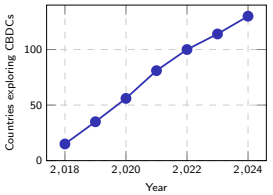
DeFi programmable tokens:

- Stablecoins with built-in yield (e.g., DAI / Sky savings rate)
- Vesting contracts: tokens unlock over time for employees
- Conditional payments: escrow released on delivery confirmation
- *2026 regulatory context:* EU MiCA stablecoin rules in force since June 2024; US GENIUS Act (signed 2025) licences payment stablecoin issuers federally

Corporate and institutional:

- JPMorgan Onyx: programmable payments for institutions
- Programmable payroll: salary tokens with auto-deductions
- Supply chain payments: conditional on delivery verification

Project	Type	Status
e-CNY	CBDC	Pilot (25+ cities)
Digital euro	CBDC	Preparation
Sand Dollar	CBDC	Live (2020)
eNaira	CBDC	Live (2021)
DAI / Sky	DeFi	Live
JPM Onyx	Corp	Live (B2B)



Source: BIS, Atlantic Council. Figures illustrative.

130+ countries are exploring programmable money through CBDCs — the question is no longer "if" but "how" and "with what safeguards."

Is programmable money a tool for financial inclusion or financial control?

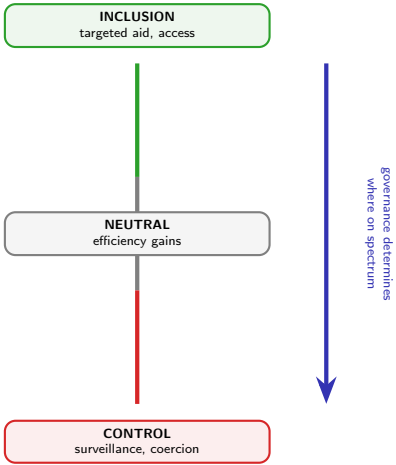
The inclusion argument:

- **Targeted welfare:** ensures benefits reach food and medicine
- **Financial literacy in code:** budgeting automated, not taught
- **Reduced corruption:** real-time audit eliminates diversion
- **Access for unbanked:** CBDC wallets without bank accounts
- **Lower remittance costs:** programmable cross-border transfers

The control argument:

- **Surveillance money:** every transaction visible to the state
- **Weaponised finance:** freezing accounts without court orders
- **Behavioural nudging:** negative interest rates force spending
- **Censorship:** blocking donations to opposition groups
- **Social engineering:** reward "good" spending, penalise "bad"

Key insight: The difference between inclusion and control is not the technology — it is the governance framework around it.



Programmable money sits on a spectrum from inclusion to control — democratic governance and privacy safeguards determine the outcome.

So what? One sentence to remember about programmable money

Programmable money can enforce rules automatically —

but the same technology that enables targeted aid

also enables unprecedented financial surveillance.

Positive potential	Negative potential	What matters most
Targeted welfare	Transaction surveillance	Democratic governance
Reduced aid leakage	Forced spending (expiry)	Privacy safeguards
Automated budgeting	Mobility restriction	Opt-out rights
Anti-corruption audit	Social credit coupling	Independent oversight

The debate over programmable money is not technical — it is political. Who writes the rules? Who can change them? Who oversees the system?

Your turn: efficient policy or authoritarian control?

Discussion Question

A government proposes replacing cash welfare with programmable tokens. The tokens:

- Expire in 30 days (cannot be saved)
- Can only purchase food, medicine, and utilities
- Cannot be transferred to another person
- Generate a full transaction log for audit

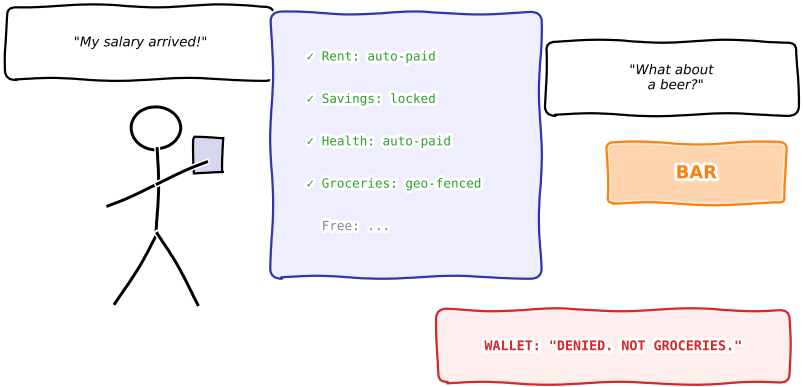
Debate: *Is this efficient social policy that protects taxpayer money — or authoritarian control that removes the dignity of choice from welfare recipients?*

Consider: Who decides what counts as "essential"? What if someone needs to buy a birthday gift for their child?

Further Reading

- Auer et al. (2022), "Central bank digital currencies: motives, economic implications and the research frontier," *BIS Working Papers*
- Brunnermeier et al. (2019), "The Digitalization of Money"
- Atlantic Council CBDC Tracker: `atlanticcouncil.org/cbdctracker`

Your Money Has Opinions



When money can say "no" — the promise and peril of programmable currency.

Learning Objectives

After completing this lecture, you will be able to:

1. **Explain** why traditional money is "dumb" and what it means to embed rules into currency [Understand]
2. **Describe** the five rule types (conditional, time-based, geo-fenced, identity-gated, auditable) [Understand]
3. **Apply** programmable money concepts to real scenarios: welfare, disaster relief, payroll [Apply]
4. **Analyze** the dual-use nature of each programmable feature (inclusion vs. control) [Analyze]
5. **Evaluate** whether programmable welfare is efficient governance or authoritarian overreach [Evaluate]

Bloom's levels covered: Understand, Apply, Analyze, Evaluate

Objectives follow Bloom's taxonomy: Understand → Apply → Analyze → Evaluate.

A $100 Bill Does Not Care

Right now, a $100 bill has **zero intelligence**:

- It does not know who holds it
- It does not know what it bought
- It cannot refuse a transaction
- It has no memory of where it has been
- It cannot expire, redirect, or report

The same $100 pays rent and buys contraband.

Banks add rules *after the fact* — fraud detection, AML (Anti-Money Laundering) checks, account freezes. But the money itself? **Completely passive.**

What you see: Traditional money carries value but no logic. Programmable money carries both value *and* executable rules.

A banknote is a bearer instrument: whoever holds it owns it, no questions asked. That simplicity is both a feature and a flaw.

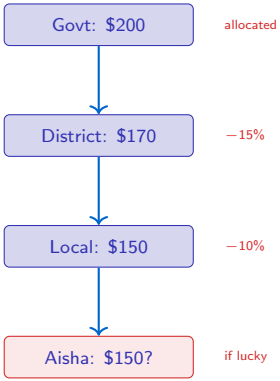
Meet Aisha: $200/Month That Never Arrives

Aisha receives $200/month in government benefits.

The journey of her money:

1. Government allocates $200 to district office
2. District official skims 15% "admin fee"
3. Local distributor takes another 10%
4. Aisha receives $150 — **if she is lucky**
5. Sometimes she receives nothing: "System error"

The World Bank estimates 25–30% of welfare spending in developing nations is lost to leakage.



India's Aadhaar-linked Direct Benefit Transfer reduced welfare leakage by an estimated 40% — but it used identity verification, not programmable money.

Meet Chen: Chasing Invoices for a Living

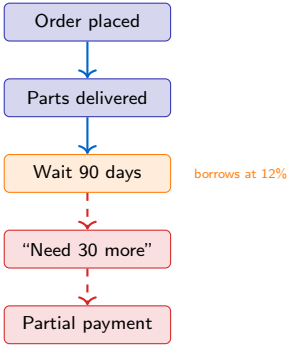
Chen runs a small manufacturing shop in Shenzhen.

His cash flow reality:

- Large buyer orders $50,000 of parts
- Payment terms: "Net 90 days"
- Day 90: "We need another 30 days"
- Day 120: partial payment arrives
- Chen borrows at 12% to cover wages in between

Late payments cause 50,000+ UK small business failures per year. (Federation of Small Businesses, 2023)

What if the money itself enforced payment terms?



Programmable escrow: funds auto-release on delivery confirmation.

The Prompt Payment Code in the UK is voluntary. Programmable money could make payment terms enforceable by code, not by lawsuit.

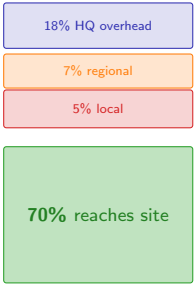
Meet Eva: "Where Did My Donation Go?"

Eva donates €500 to disaster relief after an earthquake.

What happens to her money:

- NGO (Non-Governmental Organization) headquarters takes 18% overhead
- Regional office takes 7% coordination fee
- Local partner takes 5% "distribution costs"
- **Eva's €500 becomes €350** at the disaster site
- She never learns what it actually purchased

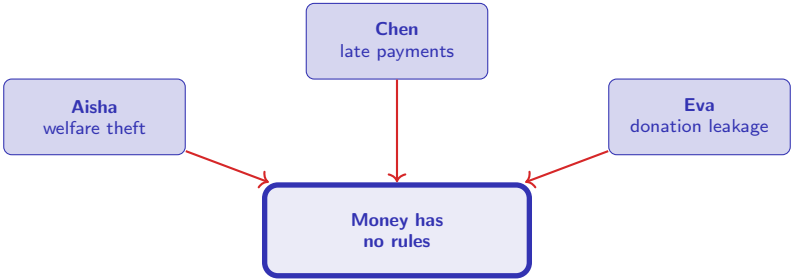
What if Eva's donation could only be spent on food, shelter, or medicine — and she could see every transaction?



What you see: 30% of Eva's donation never reaches the disaster zone. Each intermediary extracts a slice.

GiveDirectly, a cash-transfer charity, reports overhead of <3% by sending money directly to recipients' mobile wallets.

The Common Thread: Money Without Rules



What if we could embed rules into the money itself?

Three different people, three different problems, **one root cause**: money is a passive carrier of value. It obeys whoever holds it. Rules are imposed by institutions *around* money, not *within* it.

Traditional finance solves this with intermediaries (banks, auditors, regulators). Each adds cost. Programmable money asks: can the rules be built in?

Poll: If Your Money Could Enforce Rules...

If your money could enforce ONE rule automatically,

what would you program?

Some ideas from past students:

- "Auto-save 20% before I can spend"
- "Rent paid on the 1st, no excuses"
- "Donations can only buy food or medicine"
- "Subscription auto-cancels if unused 60 days"

Think about:

- What problem does your rule solve?
- Who benefits?
- Could your rule be *abused* by someone else?
- Would you accept this rule if *someone else* programmed it for you?

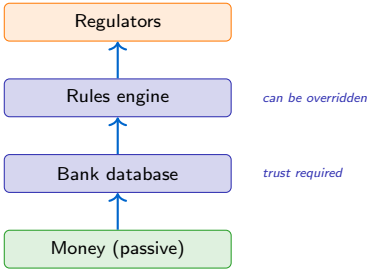
The last question is the key one. Programmable money is powerful when you set the rules — and dangerous when someone else does.

Banks Already Add Rules — Just Slowly and Expensively

Today's financial system already "programs" money — but externally:

- **Spending limits** — your card declines above a threshold
- **Fraud detection** — algorithms flag unusual transactions
- **Account freezes** — regulators can block accounts
- **Escrow** — third parties hold funds until conditions met
- **Standing orders** — auto-payments on fixed dates

The difference: these rules live in bank databases, not in the money. They require trust in the bank. They can be overridden, delayed, or circumvented.



What you see: Rules are layered *on top of* money, not embedded. Each layer adds cost and a point of failure.

The cost of these external rule layers: 1–3% of every transaction in fees, plus days of settlement delay.

What If the Rules Lived Inside the Money?

- Aisha's welfare goes **directly to her wallet**, identity-verified, no middlemen
- Chen's buyer deposits into **smart escrow**: auto-releases on delivery
- Eva's donation is **geo-fenced to the disaster zone**, every purchase logged

This is programmable money. Currency that carries its own rules. Let's see how it works.

Transition: from the problem (dumb money) to the solution (programmable money). Act 2 covers the architecture and real cases.

What Is Programmable Money?

Definition

Programmable money is digital currency with embedded, self-executing rules that govern how, when, where, and by whom it can be spent — without requiring a trusted intermediary to enforce those rules.

Key properties:

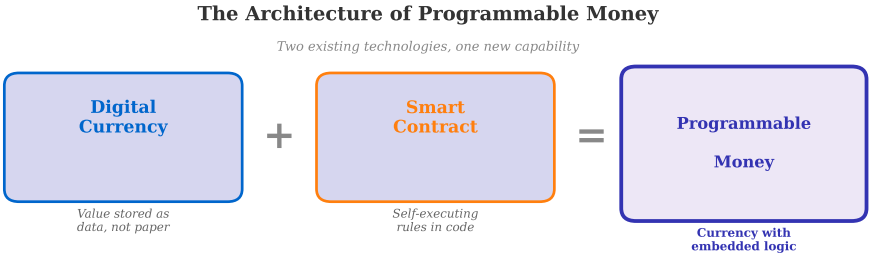
1. **Self-enforcing** — rules execute automatically
2. **Transparent** — rules are visible before acceptance
3. **Immutable** — rules cannot be changed after issuance (or only via pre-agreed governance)
4. **Composable** — rules can be combined

What it is NOT:

- Not just "digital payments" (PayPal has no embedded rules)
- Not just "cryptocurrency" (Bitcoin carries no spending conditions)
- Not just a Central Bank Digital Currency, or CBDC (a CBDC may or may not be programmable)

The distinction matters: digital $\neq$ programmable. A CBDC could be as "dumb" as a banknote or as "smart" as a conditional contract.

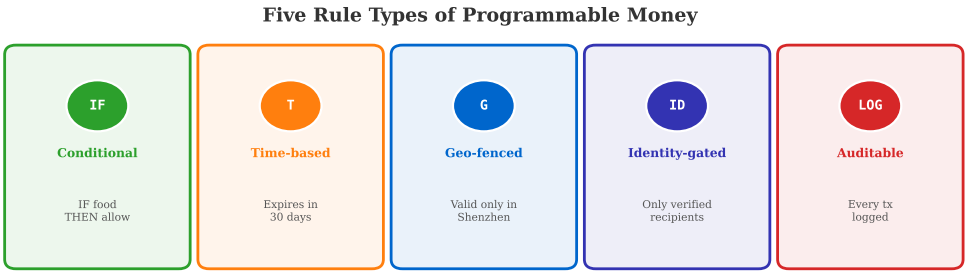
Architecture: Two Ingredients, One New Capability



Neither ingredient is new. Digital currency exists (e-money, CBDC pilots, stablecoins). Smart contracts exist (Ethereum since 2015). The innovation is **combining them**: currency that executes code at the moment of transaction.

This is why programmable money is not a single technology but a convergence of two existing ones — digital value storage and self-executing contracts.

The Five Rule Types



Every programmable money application uses one or more of these five primitives. They can be **combined**: a welfare token might be conditional (food only) + time-based (expires in 30 days) + geo-fenced (local stores only) + identity-gated (verified recipient) + auditable (every purchase logged).

These five rule types are the "instruction set" of programmable money — like AND, OR, NOT in Boolean logic.

Rule Type 1: Conditional — "Only If..."

Conditional rules restrict *what* money can purchase.

Examples:

- Welfare tokens that only work at grocery stores
- Scholarship funds that only pay tuition and books
- Corporate expense cards limited to travel and lodging
- Insurance payouts restricted to medical providers

How it works: each merchant has a category code. The token checks the code before approving.

Pseudocode

```
function spend(amount, merchant):
  if merchant.category
    in ALLOWED_LIST:
      transfer(amount)
      log(tx)
  else:
    reject(''Category
      not approved'')
```

The rule is in the money, not in the bank's database.

The US SNAP program (food stamps) already restricts purchases to food — but enforcement is at the POS terminal, not in the money itself. Programmable money moves the rule into the token.

Rule Types 2–3: Time-Based and Geo-Fenced

Time-based rules:

- Money that **expires** after a set period
- Funds that **unlock** on a future date
- Escrow that **auto-releases** on milestone
- Salary that **vests** over time

Effect: creates urgency (spend or lose) or patience (locked savings).

Geo-fenced rules:

- Stimulus tokens valid only in a specific city
- Tourism vouchers for a designated district
- Disaster relief restricted to the affected zone
- Employee allowances for on-campus use

Effect: money circulates locally, boosting targeted economies.

Combined example: Shenzhen e-CNY red envelopes (2020) were geo-fenced to Luohu district *and* expired in 7 days. The combination maximized local spending velocity.

Time-based and geo-fenced rules are the most commonly piloted programmable features, because they directly serve economic stimulus goals.

Rule Types 4–5: Identity-Gated and Auditable

Identity-gated rules:

- Only verified citizens can receive welfare tokens
- Only licensed pharmacies can accept health vouchers
- Only the intended recipient can spend the funds
- Prevents "ghost beneficiaries" (people who exist only on paper)

Benefit: eliminates the 25–30% leakage from welfare fraud.
Risk: excludes those without formal identity documents.

Auditable rules:

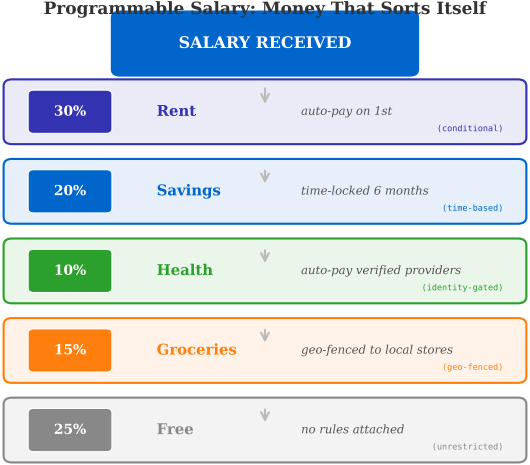
- Every transaction is logged immutably
- Donors can trace their contribution end-to-end
- Regulators get real-time compliance data
- Audits become continuous, not annual

Benefit: radical transparency reduces corruption.
Risk: total financial surveillance.

Notice the pattern: every rule type has a beneficial use *and* an authoritarian potential. We return to this in Act 3.

Identity-gating requires a digital identity system. This links programmable money to broader debates about digital ID, privacy, and state power.

Worked Example: The Programmable Salary



Case Study: Shenzhen e-CNY Red Envelopes (October 2020)

China's People's Bank of China (PBoC) ran the world's first large-scale programmable money experiment.

The design:

- 50,000 citizens selected by lottery in Luohu district
- Each received 200 RMB (≈$30) in e-CNY
- Tokens were **geo-fenced**: valid only at 3,389 merchants in Luohu
- Tokens had a **7-day expiry**: use it or lose it
- Every transaction was **logged** on PBoC infrastructure

Luohu district was chosen for its dense retail environment. The 3,389 participating merchants covered groceries, restaurants, pharmacies, and retail.

e-CNY Results: What Programmability Achieved

Outcome metrics:

- **88% redemption rate** within 7 days
- Total spending: 8.76 million RMB
- Average transaction: 40 RMB
- 62,788 individual transactions
- Incremental spending (beyond the 200 RMB): 9.01 million RMB

Key insight: recipients spent *more than* the 200 RMB at participating merchants. The tokens acted as a catalyst, not just a subsidy.

What the rules achieved:

1. **Expiry** created urgency: 88% spent vs. typical 60–70% for paper vouchers
2. **Geo-fencing** kept money local: 100% spent in Luohu
3. **Logging** gave real-time data: PBoC knew spending patterns within hours
4. **Identity-gating** prevented resale: tokens tied to lottery winners

Compare: traditional paper vouchers in similar programs had 60–70% redemption and no spending data.

The 88% redemption vs. 60–70% for paper vouchers demonstrates the behavioral impact of expiry rules. The money's code changed human behavior.

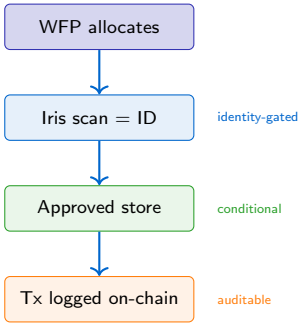
Case Study: UN World Food Programme — Building Blocks (Jordan)

Since 2017, the UN World Food Programme (WFP) has used blockchain-based programmable vouchers in Jordan's Azraq and Zaatari refugee camps.

How it works:

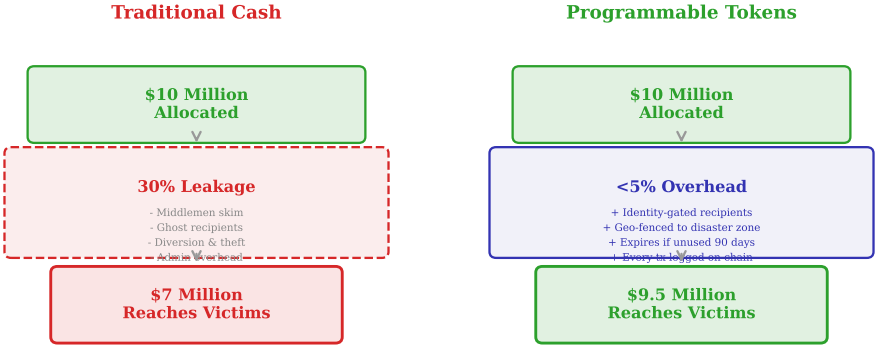
- 1 million+ refugees receive monthly food vouchers
- Vouchers are **identity-gated** via iris scans
- Valid only at **approved merchants** (conditional)
- Every transaction **logged** on a private Ethereum fork
- No bank accounts needed — refugees are unbanked

Results: WFP saved $2.4 million/month in bank transfer fees. Fraud reduced to near zero.



Building Blocks is the world's largest deployment of blockchain-based humanitarian aid. It uses 3 of the 5 rule types simultaneously.

Worked Example: $10M Disaster Relief — Cash vs. Tokens



What you see: identical allocation, radically different outcomes. Programmable tokens use identity-gating, geo-fencing, expiry, and audit logging to reduce leakage from 30% to under 5%.

The $2.5 million difference ($9.5M vs. $7M reaching victims) would feed approximately 35,000 people for a month at typical humanitarian aid costs.

Worked Example: Calculating the Savings

Traditional cash distribution:

Allocated = $10,000,000

Middleman skim = 15% = $1,500,000

Ghost recipients = 8% = $800,000

Diversion/theft = 5% = $500,000

Admin overhead = 2% = $200,000

Total leakage = 30% = $3,000,000

Reaches victims = $7,000,000

Programmable token distribution:

Allocated = $10,000,000

Middleman skim = 0% (ID-gated)

Ghost recipients = 0% (ID-gated)

Diversion/theft = 2% (geo-fenced)

Tech overhead = 3% = $300,000

Total overhead = 5% = $500,000

Reaches victims = $9,500,000

Net improvement: $2.5 million more reaches people who need it.

These figures are illustrative composites based on World Bank, Transparency International, and WFP data. Real leakage varies by country and program.

Think-Pair-Share: Design Programmable Scholarship Money

Scenario

Your university receives a €1 million donation earmarked for student scholarships. The donor wants to ensure funds are used for education, not entertainment. Design the programmable rules.

Your task (3 minutes):

1. **Which rule types** would you use? (conditional, time-based, geo-fenced, identity-gated, auditable?)
2. **What categories** should be allowed? (tuition, books, housing, food, transport, . . . ?)
3. **What should happen** to unspent funds at semester end?
4. **What edge cases** worry you? (Birthday gift? Emergency? Mental health?)

The hard question: Should the scholarship cover a coffee with friends? A cinema ticket? Where is the line between "education-related" and "not"?

This exercise forces you to confront the limits of programmability: every rule creates winners and losers. The edge cases are where the ethics live.

Beyond Welfare: Corporate Programmable Money

Supply chain payments:

- Milestone-based escrow for contractors
- Auto-release on IoT (Internet of Things) delivery confirmation
- Multi-party settlement (supplier $\rightarrow$ shipper $\rightarrow$ insurer)

Corporate treasury:

- Employee expense tokens (category-limited)
- Inter-subsidiary transfers with compliance rules
- Tax escrow: VAT auto-segregated at point of sale

Insurance:

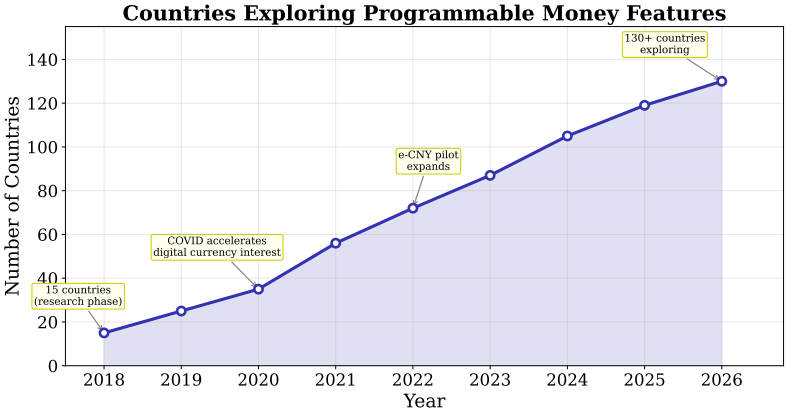
- Parametric insurance: payout triggers automatically when earthquake > 6.0 on Richter scale
- Health claims: tokens valid only at approved providers
- Reinsurance settlement in hours, not months

Carbon credits:

- Tokens that can only offset verified emissions
- Auto-retire on use (no double-counting)
- Transparent audit trail for ESG (Environmental, Social, Governance) reporting

Programmable money is not just a government tool. Enterprises are piloting it for supply chains, treasury, insurance, and sustainability compliance.

The Global Race: Who Is Building Programmable Money?



What you see: the number of countries exploring programmable features in digital currency has grown from 15 (2018) to over 130 (2026). Not all are building CBDCs — some explore private stablecoins or tokenized deposits.

Source: Atlantic Council CBDC Tracker (2026). "Exploring" includes research, pilot, and launched phases. Programmability depth varies widely.

Key Players and Their Approaches

Player	Approach	Programmability Focus
China (e-CNY)	State-issued CBDC	Expiry, geo-fencing, spending categories
EU (digital euro)	Research/pilot phase	Privacy-preserving; limited programmability
Nigeria (eNaira)	Launched 2021	Welfare disbursement, merchant categories
Singapore (Orchid)	"Purpose-bound money"	Vouchers with programmable conditions
Private sector	Stablecoins (USDC, USDT)	Smart-contract composability, DeFi
UN/WFP	Blockchain vouchers	Identity-gated humanitarian aid

Note: the CBDC-specific architecture (retail vs. wholesale, two-tier models, BIS Money Flower) is covered in Module 6: Digital Currencies. This lecture focuses on the *programmability layer*, regardless of who issues the money.

Singapore's "purpose-bound money" concept is the clearest articulation of programmability as a feature distinct from digital currency issuance.

So Far, So Good — But There's a Dark Side

Everything we've celebrated — spending restrictions, expiry, geo-fencing, identity-gating, audit trails — has a **mirror image**:

- Spending restrictions → surveillance of every purchase
- Expiry → forced spending, no saving
- Geo-fencing → mobility restriction
- Identity-gating → exclusion of the undocumented
- Audit trails → social credit scoring

Act 3 examines the authoritarian potential of the same features we just praised. The technology is neutral; governance determines the outcome.

Dark Side 1: Total Financial Surveillance

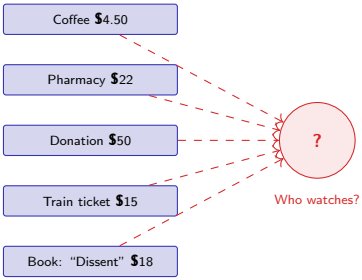
The promise: every transaction logged for transparency and anti-fraud.

The reality: every purchase you make — every coffee, every medicine, every donation — is visible to the issuer.

What this enables:

- Government knows you bought a pregnancy test
- Employer knows you visited a job interview location
- Insurer knows you bought alcohol and cigarettes
- Political party knows you donated to the opposition

Cash is the last anonymous payment method. Programmable money could eliminate financial privacy entirely.



What you see: every transaction flows to a central observer. The question is not *whether* this data exists, but *who* has access.

The ECB has stated the digital euro will have "cash-like privacy" for small transactions. But "cash-like" is not "cash" — and the threshold is set by regulators.

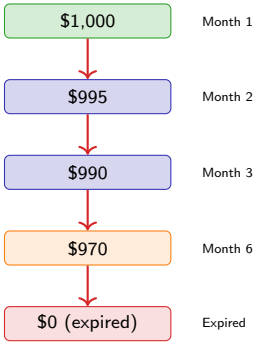
Dark Side 2: Expiry = Forced Spending

The promise: expiry creates urgency, boosts economic velocity.

The dark version:

- Money that **expires** means you **cannot save**
- During economic crisis, citizens might be forced to spend rather than build reserves
- Negative interest rates become trivially enforceable: your balance shrinks 0.5%/month automatically
- The poor — who need savings most — are hit hardest

Historical parallel: Silvio Gesell's "demurrage currency" (1916) proposed money that loses value over time to encourage spending. It was tried in Wörgl, Austria (1932–1933) during the Great Depression, then banned by the central bank.



Your savings evaporate by design.

Japan and the ECB have discussed negative interest rates as stimulus. With cash, people hoard bills to avoid it. With programmable money, there is no escape.

Dark Side 3: Geo-Fencing = Mobility Restriction

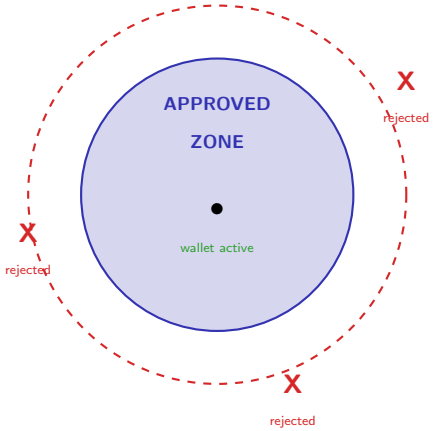
The promise: keep stimulus money circulating locally.

The dark version:

- Money that only works in your city = **you cannot leave your city**
- Dissidents could have money that excludes transport
- Migrant workers could be restricted to factory zones
- Refugees could be confined to camp boundaries

When your money cannot travel, neither can you.

This is not hypothetical. China's *hukou* system already restricts public services by region. Programmable money could extend this to private transactions.



Geo-fencing for economic stimulus is opt-in and temporary. Geo-fencing imposed by a state on its citizens is a digital border.

Dark Side 4: Transaction Logs + Scoring = Social Credit

If every transaction is logged *and* linked to identity, a government could:

1. **Score citizens** based on spending patterns
2. **Reward** "desirable" behavior (healthy food = bonus)
3. **Punish** "undesirable" behavior (alcohol = penalty)
4. **Restrict** access based on score (low score = no train tickets)

This is not science fiction. China's Social Credit System (SCS) already restricts travel for millions of citizens flagged as "untrustworthy."

Programmable money would give the system **real-time enforcement** instead of after-the-fact penalties.

What you see: spending data + identity + scoring = programmable social control.

By 2023, Chinese courts had restricted over 26 million air tickets and 6 million high-speed rail tickets for individuals on the SCS blacklist.

Scenario: The Dissident's Wallet

Imagine this:

A journalist publishes an article critical of the government. Within hours:

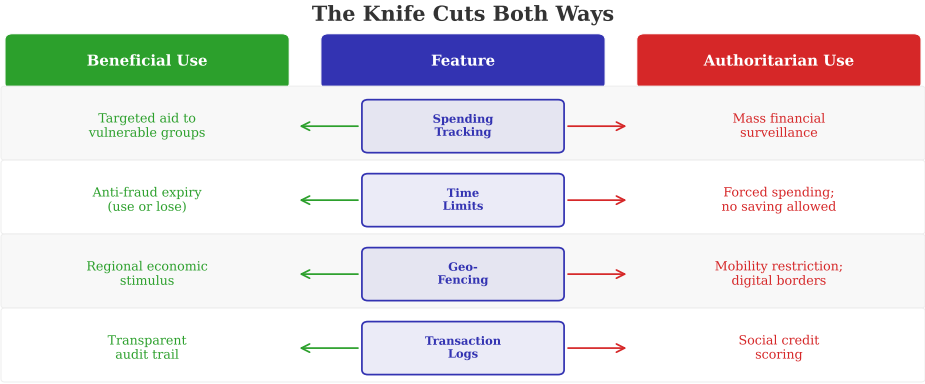
1. Her programmable money is **geo-fenced** to her home district
2. Her transport category is **disabled** — she cannot buy bus, train, or flight tickets
3. Her money is given a **7-day expiry** — she must spend immediately or lose it
4. Her **transaction logs** are shared with investigators
5. Merchants who accept her money are **flagged** for review

No bank account frozen. No court order. No human decision.
The code does it all.

Every rule we praised for welfare efficiency is now a weapon. The technology did not change — only the *intent*.

This is the core ethical dilemma: the same code that protects Aisha's welfare can silence a journalist. The rules are identical; the governance is different.

The Knife Cuts Both Ways



What you see: every programmable feature serves two masters. The technology is genuinely dual-use. Governance — not code — determines which side wins.

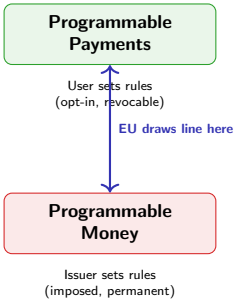
The European Debate: How Much Programmability?

The European Central Bank (ECB) and European Commission have been debating the digital euro's programmability since 2021.

Key positions:

- ECB President Lagarde (2023): "The digital euro will *not* be programmable money" — but will support "programmable payments" (conditional triggers)
- Distinction: *programmable money* (issuer sets rules) vs. *programmable payments* (user sets rules)
- European Parliament: concerns about surveillance, financial exclusion, and central bank overreach

The EU is drawing a line: users can program their own payments, but the currency itself should not carry issuer-imposed restrictions.



The distinction between "programmable money" and "programmable payments" was first articulated by the BIS in its 2023 Annual Economic Report, Chapter III.

Can We Have Programmability Without Surveillance?

Technical solutions being explored:

- **Zero-knowledge proofs (ZKPs):** prove you spent on food without revealing *which* food or *which* store
- **Tiered privacy:** small transactions (<€50) anonymous; large transactions require ID
- **Hardware enclaves:** rules enforced on-device, data never leaves the phone
- **Time-limited logs:** transaction data auto-deletes after compliance period (e.g., 90 days)

ZKP Analogy

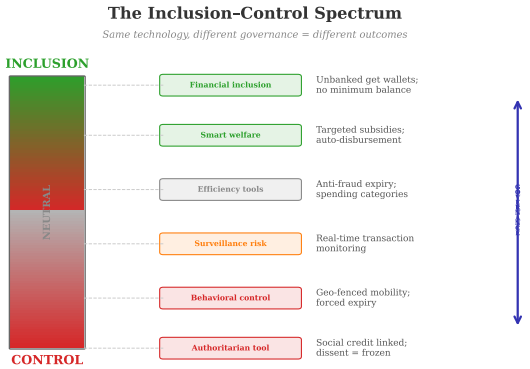
You show a bouncer you are over 18 without revealing your birth date, name, or address. The bouncer verifies "over 18" = TRUE and lets you in.

Similarly, a ZKP can verify "spent on food" = TRUE without revealing what, where, or when.

These techniques add complexity and cost, but they are technically feasible today.

The ECB's digital euro prototype uses "selective disclosure" — a simpler form of privacy that reveals only what regulators need. Full ZKPs remain a research goal.

The Inclusion-Control Spectrum



What you see: the same technology sits on a spectrum from inclusion to control. Where any given deployment lands depends on governance design: who sets the rules, who can change them, who audits the auditors.

The spectrum is not binary. Most real deployments mix beneficial and concerning features. The question is: what safeguards prevent drift toward control?

Five Governance Design Principles

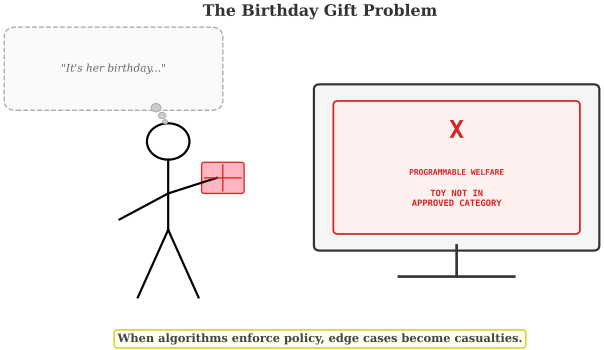
To keep programmable money on the inclusion side of the spectrum:

1. **Opt-in over imposition:** users should choose their own spending rules; issuers should not impose them without consent
2. **Minimal data collection:** collect only what compliance requires; delete when no longer needed
3. **Separation of powers:** the entity that issues money should not be the entity that monitors transactions
4. **Sunset clauses:** every restriction should have an expiry date and require active renewal
5. **Override mechanisms:** humans must be able to override automated rules in emergencies (the "birthday gift" exception)

The irony: programmable money needs *human* governance guardrails to prevent *algorithmic* overreach.

These principles echo the EU's approach to AI regulation (the EU AI Act, 2024): risk-based, human-in-the-loop, with mandatory transparency.

The Birthday Gift Problem



A parent on welfare wants to buy a toy for her daughter's birthday. The programmable welfare token rejects the purchase: "Not in approved category."

The algorithm is correct. Toys are not food or medicine. The rule worked as designed.
But is it just? Every rigid rule creates edge cases where efficiency becomes cruelty.

This is the "trolley problem" of programmable money: is automatic [illegible] worth the cost of denying edge case humanity?

Discussion: Should the Birthday Gift Be Allowed?

Arguments FOR strict rules:

- Taxpayer money should fund essentials
- Exceptions create loopholes
- 30% leakage costs lives; strict rules save them
- Rules apply equally to everyone
- "If you allow toys, what about alcohol?"

Arguments AGAINST strict rules:

- Dignity matters: welfare recipients are human
- Children deserve birthdays
- A $10 toy does not threaten the system
- Paternalism breeds resentment, not compliance
- "If you ban toys, do you ban birthday cakes?"

The deeper question: who should decide what "essential" means — a legislature, an algorithm, or the recipient?

There is no right answer. The purpose of this discussion is to show that programmable money forces society to codify values that were previously left ambiguous.

The Code Is Political

"Code is law" is incomplete.

Code is policy — and policy is political.

Every line of code in programmable money is a **policy decision**:

- Defining "food" as an approved category is a *nutritional policy*
- Setting a 7-day expiry is a *monetary policy*
- Geo-fencing to a district is a *regional development policy*
- Logging every transaction is a *surveillance policy*

Engineers writing smart contracts are, whether they realize it or not, writing legislation.

Lawrence Lessig's "Code is Law" (1999) argued that software architecture shapes behavior as powerfully as legal regulation. Programmable money proves him right.

Five Key Takeaways

1. **Traditional money is "dumb":** it carries value but no logic. Rules are imposed by external intermediaries at high cost.
2. **Programmable money embeds rules in the currency itself:** conditional, time-based, geo-fenced, identity-gated, and auditable logic travel with every token.
3. **Real-world pilots show measurable benefits:** e-CNY achieved 88% redemption (vs. 60–70% for paper vouchers); WFP saved \$2.4M/month in Jordan.
4. **Every feature is dual-use:** spending tracking enables both targeted aid and mass surveillance. The technology is neutral; governance determines the outcome.
5. **The hardest questions are political, not technical:** who sets the rules? Who can override them? What counts as "essential"?

If you remember one thing: programmable money is a governance challenge disguised as a technology challenge.

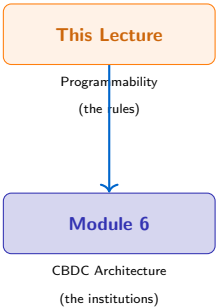
Bridge: From Programmability to CBDCs

This lecture covered **what programmability does** and **why it matters**.

Module 6 (Digital Currencies) will cover:

- **Who issues** programmable money (central banks, private sector, hybrid models)
- The BIS Money Flower taxonomy
- Retail vs. wholesale CBDC architecture
- The digital euro design process
- Project Helvetia (Swiss National Bank)
- Cross-border CBDC interoperability

Think of it this way: this lecture was about the *rules*. Module 6 is about the *institutions* that write and enforce them.



Deduplication: this lecture intentionally avoids CBDC architecture, BIS taxonomy, and central bank pilots. Those topics are covered in Module 6.

Final Discussion: Efficient or Authoritarian?

Scenario

A government introduces programmable welfare tokens with these rules:

- Valid for food, medicine, and children's clothing only
- Expires in 30 days if unspent
- Geo-fenced to the recipient's registered district
- Every transaction logged for anti-fraud auditing

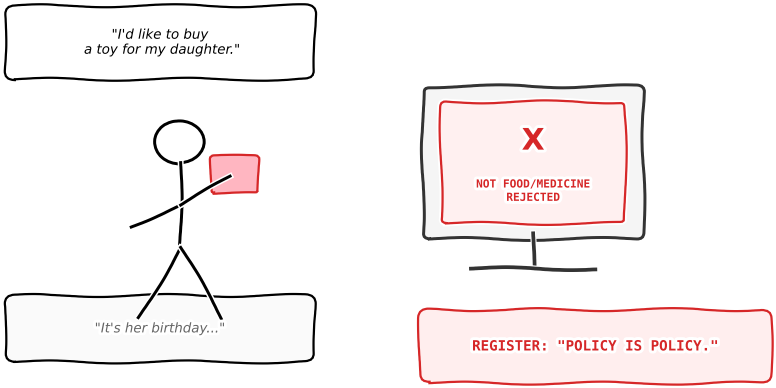
Discuss in groups (5 minutes):

1. Is this **efficient governance** or **authoritarian control**?
2. What happens to the parent who wants to buy a birthday gift?
3. What happens to the family that needs to travel for medical care outside their district?
4. Would you accept these rules if applied to *your* income?

The last question is always the test: would you accept for yourself what you design for others?

John Rawls' "veil of ignorance": design rules as if you don't know whether you'll be the programmer or the recipient.

Closing Thought



Efficient policy or algorithmic cruelty? You decide.

Efficient policy or algorithmic cruelty? The answer depends on who writes the code — and who has to live under it.

Thank You — Selected References


Bank for International Settlements (2023). *Annual Economic Report*, Chapter III: Blueprint for the future monetary system.


European Central Bank (2023). *A stocktake on the digital euro — Progress report.*


World Food Programme (2022). *Building Blocks: Blockchain for Zero Hunger.*


Gesell, S. (1916). *The Natural Economic Order.* (Demurrage currency proposal.)


Lessig, L. (1999). *Code and Other Laws of Cyberspace.* Basic Books.


Atlantic Council (2026). *Central Bank Digital Currency Tracker.* `https://www.atlanticcouncil.org/cbdctracker/`


Monetary Authority of Singapore (2023). *Purpose Bound Money Technical Whitepaper.*


Rawls, J. (1971). *A Theory of Justice.* Harvard University Press.

Further reading on CBDCs, digital euro architecture, and cross-border interoperability is covered in Module 6: Digital Currencies.