# L01: Digital Payments & Mobile Money Economics

## Extended Slides – BSc Digital Finance Course

Digital Finance

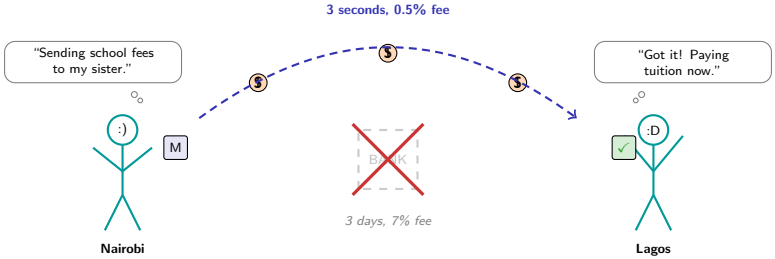
# What Will You Be Able to Do After This Lecture?

By the end of this extended lecture, you will be able to:

1. **Explain payment system clearing mechanics** — distinguish bilateral netting from gross settlement, compute net positions in a multi-bank obligation matrix, and evaluate the liquidity–risk trade-off between RTGS and DNS
2. **Analyse mobile money unit economics** — model two-sided market pricing using the Rochet–Tirole interchange fee framework, compute float revenue as a function of transaction velocity, and assess the sustainability of low-value, high-volume business models
3. **Quantify network effects in payment platforms** — apply Metcalfe's law and logistic growth models to estimate critical mass thresholds, simulate adoption trajectories, and decompose interchange fee structures
4. **Decompose cross-border payment costs** — break down FX spreads into mid-market rate, bid-ask, and markup layers, trace correspondent banking chains, and compare legacy SWIFT rails with real-time alternatives
5. **Construct financial inclusion indices** — build composite indicators from World Bank Findex data, weight multiple dimensions (access, usage, quality), and critically evaluate whether mobile wallets constitute meaningful inclusion
6. **Implement payment system computations in Python** — write correct, concise code for hash verification, interchange fee optimisation, float modelling, network simulation, remittance cost calculation, and composite index construction

---

**Each objective maps to one of the five sections that follow. The lecture alternates between theory, data, and code.**




*A $200 remittance once took 3 days and cost $14 in fees. Today it takes 3 seconds and costs $1. What changed?*

*The infrastructure behind that 3-second transfer is the subject of this lecture.*

# How Does Netting Reduce Trillions in Daily Obligations to a Fraction?

**Problem:** $n$ banks owe each other pairwise obligations. Gross settlement requires transferring every obligation individually. Netting computes the net position of each bank and settles only the difference.

**Bilateral obligation matrix.** Let $N_{ij}$ denote the gross obligation of bank $i$ to bank $j$:

$$\mathbf{N} = \begin{pmatrix} 0 & N_{12} & N_{13} \\ N_{21} & 0 & N_{23} \\ N_{31} & N_{32} & 0 \end{pmatrix} \quad \text{e.g.} \quad \mathbf{N} = \begin{pmatrix} 0 & 80 & 50 \\ 60 & 0 & 70 \\ 40 & 30 & 0 \end{pmatrix}$$

**Net position** of bank $i$ after multilateral netting:

$$P_i = \sum_{j \neq i} N_{ji} - \sum_{j \neq i} N_{ij} = (\text{total received}) - (\text{total paid})$$

**Example:** $P_1 = (60 + 40) - (80 + 50) = -30$ (net payer), $P_2 = (80 + 30) - (60 + 70) = -20$ (net payer), $P_3 = (50 + 70) - (40 + 30) = +50$ (net receiver).

**Efficiency gain:**

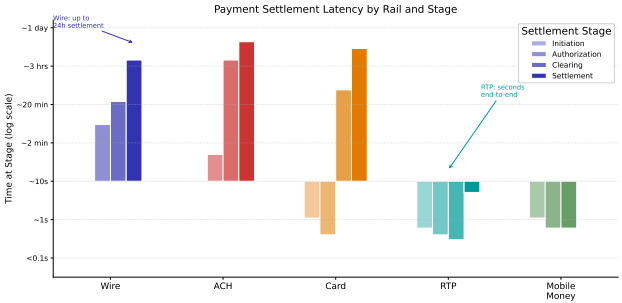
- **Gross settlement:** $\sum_{i \neq j} N_{ij} = 80 + 50 + 60 + 70 + 40 + 30 = 330$ units transferred
- **Net settlement:** $\sum_{i:P_i<0} |P_i| = 30 + 20 = 50$ units transferred
- **Reduction:** $1 - 50/330 = 84.8\%$ fewer fund transfers required

Multilateral netting is why systems like CLS Bank can settle \$6 trillion in daily FX obligations with only \$60 billion in actual fund movements — a 99% reduction.

---

**Netting reduces systemic liquidity needs by 80–99%, but introduces counterparty risk: if one bank fails before settlement, all net positions unwind.**

# How Fast Is “Instant” – And Where Does Latency Hide?




**Latency decomposition by payment rail**

- **Card networks (Visa/MC):** Authorization $<$1s, but settlement T+1 to T+2 (24–48h). “Instant” is an illusion backed by credit risk
- **RTGS systems (SIC, Fedwire):** True real-time gross settlement, but restricted to interbank wholesale. Latency: seconds to minutes
- **ACH / SEPA batch:** Deferred net settlement with cutoff times. Latency: 4–24h depending on submission window
- **Instant schemes (FedNow, SEPA Instant):** End-to-end $<$10 seconds, 24/7/365. Net settlement near-real-time
- **Mobile money (M-Pesa):** On-network $<$3 seconds. Off-network (interoperable) transfers: 1–24 hours
- **Cross-border (SWIFT gpi):** 40% within 5 minutes, but tail latency: 2–5 business days through correspondent chains

**Key insight:** “Instant” is defined differently at each layer – authorization, clearing, and settlement each have their own clock.

**Users experience authorization speed. Banks experience settlement speed. The gap between them is where risk and cost accumulate.**

# How Do You Prove a Payment Message Has Not Been Tampered With?

```python
import hashlib, json, hmac

def create_payment(sender, receiver, amt, key):
    """Create payment message with HMAC."""
    msg = {"sender": sender,
           "receiver": receiver,
           "amount": amt,
           "currency": "USD"}
    payload = json.dumps(msg, sort_keys=True)
    tag = hmac.new(
        key.encode(), payload.encode(),
        hashlib.sha256).hexdigest()
    return {"payload": payload, "hmac": tag}

def verify_payment(payment, key):
    """Verify integrity of payment message."""
    tag = hmac.new(
        key.encode(),
        payment["payload"].encode(),
        hashlib.sha256).hexdigest()
    return hmac.compare_digest(tag, payment["hmac"])
```

**Integrity verification in payment systems**

- **Why hashing matters:** Every SWIFT message, ACH batch, and card authorization carries integrity checks. A single flipped bit could redirect millions
- **HMAC vs plain hash:** Plain SHA-256 detects accidental corruption. HMAC (Hash-based Message Authentication Code) also proves the sender's identity using a shared secret key
- **Deterministic serialization:** `sort_keys=True` ensures identical JSON byte sequences regardless of dict insertion order — critical for reproducible hashes
- **Timing-safe comparison:** `hmac.compare_digest` prevents timing side-channel attacks that could leak the correct hash byte-by-byte
- **Real-world usage:** ISO 20022 messages use XML digital signatures (X.509 certificates) rather than symmetric HMAC, but the principle is identical

**Every payment message in SWIFT, ACH, and card networks carries cryptographic integrity checks. Tampering detection is the foundation of payment security.**

# Why Do Payment Systems Collapse Under Load – and How Do You Predict It?

**Model:** A payment processor handles transactions as a single-server queue (M/M/1). Transactions arrive at rate $\lambda$ (Poisson), processing time is exponential with rate $\mu$.

**Key quantities:**

$$\rho = \frac{\lambda}{\mu} \quad \text{(utilization)} \qquad L_q = \frac{\rho^2}{1-\rho} \quad \text{(avg queue length)} \qquad W = \frac{1}{\mu - \lambda} \quad \text{(avg time in system)}$$

**Numerical example:** Payment switch, $\mu = 1{,}000$ txn/s.

- Normal load $\lambda = 800$: $\rho = 0.8$, $L_q = 3.2$ txn, $W = 5$ ms
- Peak load $\lambda = 950$ (Black Friday): $\rho = 0.95$, $L_q = 18.05$ txn, $W = 20$ ms
- Near capacity $\lambda = 990$: $\rho = 0.99$, $L_q = 98.01$ txn, $W = 100$ ms

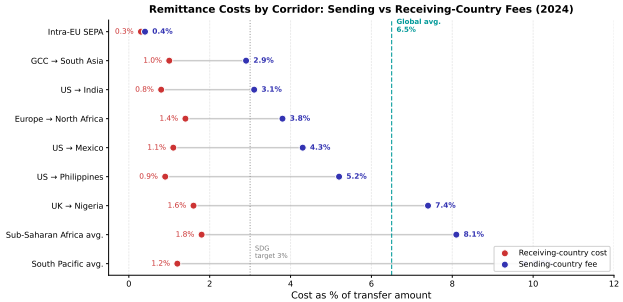
**The non-linearity trap:** Going from 80% to 99% utilization increases wait time by **20x**. Payment systems that "work fine" at normal load suddenly collapse during peaks. The fix is capacity headroom ($\mu \gg \lambda$) or load shedding (reject low-priority transactions when $\rho > 0.85$).

**M/M/c extension:** With $c$ parallel servers, $\rho = \lambda/(c\mu)$. Visa's VisaNet uses $c > 100$ nodes for 65,000+ txn/s peak capacity. Horizontal scaling transforms a non-linear problem into a linear one.

---

**Queuing theory explains why payment systems need 3–5x headroom above average load. The last 5% of utilization creates 90% of the latency.**

# Why Does Sending $200 Cost $1 in Some Corridors and $20 in Others?




**Remittance cost variation by corridor**

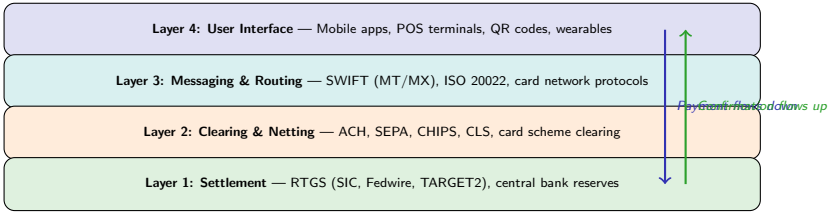
- **SDG target:** UN SDG 10.c: reduce remittance costs below 3% by 2030. The global average remains above 6%
- **Cheapest corridors:** GCC-to-South-Asia (UAE → India: ∼1%) benefit from high volume, competition, and digital channels
- **Most expensive:** Sub-Saharan Africa (South Africa → Malawi: >15%) – low volume, limited competition, cash-dependent last mile
- **Cost components:** Transfer fee (flat + percentage), FX markup over mid-market rate, correspondent charges
- **Digital disruption:** Mobile money corridors (M-Pesa to M-Pesa in East Africa) achieve below 1%, proving the problem is infrastructure
- **Regulatory barriers:** AML compliance costs are fixed overhead that falls disproportionately on small transfers

**The poverty penalty:** The poorest pay the highest percentage fees because compliance costs are fixed, not proportional.

**Remittance costs are a tax on poverty. The $200 benchmark matters because it is the median remittance size globally (World Bank, 2024).**

# What Are the Four Layers Every Payment Must Traverse?

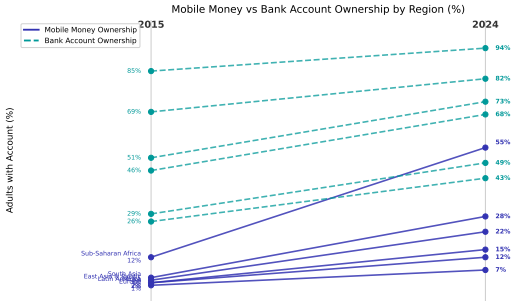
**Payment infrastructure stack.** Every digital payment – tap, transfer, or remittance – traverses four functional layers. Understanding the stack explains what FinTechs can disrupt and what they must ride on top of.




- **Layer 1 (Settlement):** The only layer with true finality. RTGS systems (SIC, Fedwire, TARGET2) provide irrevocable transfer of central bank money
- **Layer 2 (Clearing):** Computes net obligations. Deferred net settlement reduces liquidity needs 80–99% but introduces intraday credit exposure
- **Layer 3 (Messaging):** Carries payment instructions between institutions. ISO 20022 migration (2022–2025) enriches data from 140 characters (MT103) to structured XML
- **Layer 4 (Interface):** Apple Pay, Twint, M-Pesa sit here – they innovate on experience while depending on layers below
- **FinTech positioning:** Most FinTechs innovate at Layers 3–4 (Stripe, Wise). Very few touch Layers 1–2, which require central bank access
- **Mobile money exception:** M-Pesa built its own Layer 1 (trust account) and Layer 2 (proprietary clearing) – bypassing banking infrastructure entirely

**FinTechs build at Layers 3–4. Central banks control Layers 1–2. Mobile money operators like M-Pesa are rare exceptions that built their own full stack.**

# Which Countries Leapfrogged Banks Entirely – and Why?




**Mobile money adoption trajectories**

- **Leapfrogging pattern:** Countries with low bank branch density and high mobile penetration adopt mobile money fastest. Kenya went from 0 to 80% adult adoption in 10 years
- **Necessary conditions:** (1) Supportive regulation (light-touch licensing), (2) dominant mobile operator willing to invest, (3) unmet demand for payment and transfer services
- **Kenya (M-Pesa):** Launched 2007 by Safaricom. Now processes $30B+ annually. More M-Pesa agents than bank branches and ATMs combined
- **Bangladesh (bKash):** 60M+ users. Agent-led model with heavy subsidies to build network density
- **India (UPI):** Government-led interoperable system. 10B+ monthly transactions by 2024. Not mobile money per se, but serves similar function
- **Stalled markets:** Nigeria and South Africa — strong bank lobbying delayed mobile money licensing for years
- **Europe/US absence:** Existing card infrastructure and banking penetration eliminated the leapfrog opportunity

**Mobile money adoption is inversely correlated with existing bank infrastructure. You leapfrog banks only when there are no banks to leap over.**

# What Interchange Fee Maximises Platform Revenue on Both Sides?

```python
import numpy as np

def optimal_interchange(eps_m, eps_c, c_m, c_c):
    """Rochet-Tirole optimal interchange fee.
    eps_m: merchant demand elasticity (>0)
    eps_c: consumer demand elasticity (>0)
    c_m: marginal cost, merchant side
    c_c: marginal cost, consumer side
    Returns: optimal interchange fee a*."""
    # Platform sets fee to balance both sides
    # Higher elasticity side gets lower fee
    total_cost = c_m + c_c
    # Lerner-style markup inversion
    a_star = (eps_m * c_c - eps_c * c_m) \
             / (eps_m + eps_c)
    return a_star

# Example: card network pricing
eps_m, eps_c = 1.5, 0.5 # merchants elastic
a = optimal_interchange(eps_m, eps_c, 0.02, 0.01)
print(f"Optimal interchange: {a:.4f}")
# Merchants pay more because consumers are
# less elastic (sticky to card brand)
```

**Two-sided market pricing logic**

- **Core idea:** A payment platform serves two sides — merchants and consumers. The optimal price is NOT cost-based; it depends on relative elasticities
- **Subsidy structure:** The more price-sensitive (elastic) side gets subsidized. Consumers get free accounts; merchants pay 1.5–3% per transaction
- **Why consumers pay less:** Consumer elasticity $\epsilon_c$ is low (sticky to card brand). Merchant elasticity $\epsilon_m$ is higher (can switch acquirers or refuse cards)
- **Interchange flow:** Acquirer (merchant's bank) pays issuer (consumer's bank). This subsidy funds rewards programs that attract consumers
- **Regulation impact:** EU Interchange Fee Regulation (2015) capped interchange at 0.2% (debit) and 0.3% (credit), overriding market-optimal pricing
- **M-Pesa twist:** No interchange — Safaricom captures both sides. Consumers pay tiered fees; merchants receive for free (inverted subsidy)

**Two-sided market pricing explains why consumers get free accounts while merchants pay 2–3%. The subsidy flows to the more elastic side.**

# Why Is the Socially Optimal Interchange Fee Not Zero?

**Model setup (Rochet & Tirole, 2003).** A platform connects merchants ($M$) and consumers ($C$). Each side pays a fee: merchants pay $p_M$, consumers pay $p_C$. Total fee $p = p_M + p_C$ is fixed by competition; the platform chooses the **fee structure** (how to split $p$).

**Demand functions.** Let $D_M(p_M)$ and $D_C(p_C)$ denote demand on each side, with elasticities $\epsilon_M$ and $\epsilon_C$:

$$D_M(p_M) = D_M^0 \cdot p_M^{-\epsilon_M}, \qquad D_C(p_C) = D_C^0 \cdot p_C^{-\epsilon_C}$$

**Platform profit:**

$$\pi = (p_M - c_M) \cdot D_M(p_M) + (p_C - c_C) \cdot D_C(p_C) \quad \text{subject to} \quad p_M + p_C = p$$

**Optimal fee structure** (first-order condition, Lerner rule on each side):

$$\frac{p_M - c_M}{p_M} = \frac{1}{\epsilon_M}, \qquad \frac{p_C - c_C}{p_C} = \frac{1}{\epsilon_C}$$

**Interchange fee** $a = p_M - c_A$ (acquirer cost): the transfer from acquirer to issuer that implements the optimal split. The key result:

$$a^* = \frac{\epsilon_M \cdot c_C - \epsilon_C \cdot c_M}{\epsilon_M + \epsilon_C}$$

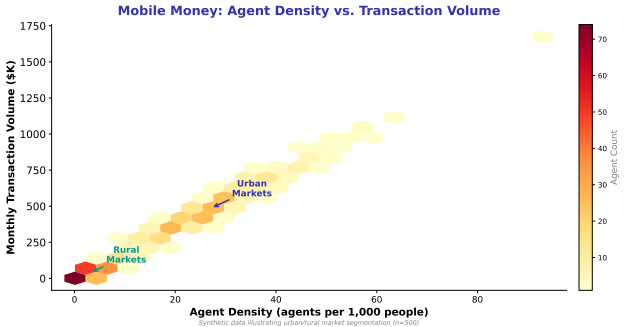
**Interpretation:** If merchants are more elastic ($\epsilon_M > \epsilon_C$), $a^*$ shifts cost to the consumer side (via rewards funded by interchange). A zero interchange fee is optimal **only if** both sides have identical elasticity and identical costs — a condition never observed in practice.

**Policy tension:** Regulators cap interchange to reduce merchant fees, but this shrinks consumer rewards and may reduce total transaction volume. The welfare effect is ambiguous.

---

**Rochet–Tirole (2003) showed that the socially optimal interchange fee is generally nonzero. Fee structure matters as much as fee level in two-sided markets.**

# Why Does Agent Density Determine Mobile Money Success or Failure?




**Agent networks as infrastructure**

- **What agents do:** Cash-in, cash-out, account registration. Physical bridge between cash and digital economies
- **Density threshold:** Adoption accelerates above 100 agents per 100,000 adults. Below this, cash-out friction kills usage
- **Kenya leads:** 250,000+ M-Pesa agents vs 3,000 bank branches. Average Kenyan lives within 1 km of an agent
- **Agent economics:** Commissions of 0.3–1.0% per transaction. Busy Nairobi agent earns **$**300–500/month
- **Liquidity management:** Agents must pre-fund e-float. Running out of cash or e-float means turning away customers
- **Network design:** Safaricom uses tiered super-agents that manage float rebalancing across retail agent clusters
- **Bank comparison:** 10,000 bank branches cost **$**5–10B. 100,000 agents cost **$**50–100M. Agents are 100x cheaper per distribution point

**Mobile money is an agent business disguised as a technology product. Network density, not app design, determines adoption.**

# How Does M-Pesa Earn Interest on Money That Is Always Moving?

```python
import numpy as np

def float_revenue(total_balance, velocity,
                  interest_rate, days=365):
    """Model trust account float revenue.
    total_balance: aggregate user balances ($)
    velocity: avg turnover per year
    interest_rate: annual rate on trust acct
    Returns: annual float interest revenue."""
    # Average float = balance / velocity
    # Higher velocity = less idle float
    avg_float = total_balance / velocity
    annual_rev = avg_float * interest_rate
    daily_rev = annual_rev / days
    return annual_rev, daily_rev

# M-Pesa example (approximate 2024 figures)
bal = 2.5e9      # $2.5B total user balances
vel = 15.0       # money turns over 15x/year
rate = 0.08      # 8% Kenya treasury rate
ann, day = float_revenue(bal, vel, rate)
print(f"Avg float: ${bal/vel/1e6:.0f}M")
print(f"Annual revenue: ${ann/1e6:.1f}M")
print(f"Daily revenue: ${day/1e3:.0f}K")
```

**Float economics in mobile money**

- **Trust account:** Regulators require mobile money operators to hold 100% of user balances in a ring-fenced trust account at a commercial bank
- **The float:** At any moment, the aggregate balance earns interest. Even if individual balances are small ($5–50), the sum is billions
- **Velocity matters:** High transaction velocity means money moves fast and average idle float is low. M-Pesa's velocity ($\sim$15x/year) is 3x higher than traditional bank deposits ($\sim$5x/year)
- **Revenue significance:** Float interest is M-Pesa's second-largest revenue source after transaction fees, contributing 10–15% of total revenue
- **Interest rate sensitivity:** In low-rate environments (e.g., Europe at 0%), float revenue disappears. This partly explains why mobile money thrives in high-rate emerging markets
- **Regulatory question:** Should float interest belong to the operator or the users? Kenya's 2022 amendment requires partial sharing with users

**Float revenue explains why mobile money is profitable in high-interest-rate emerging markets but unviable in low-rate developed economies.**

# What Does the Velocity of Mobile Money Tell Us About Financial Behaviour?

**Quantity theory applied to mobile money.** The equation of exchange:

$$M \cdot V = P \cdot Q$$

where $M$ is aggregate mobile money balances, $V$ is velocity (turnovers per period), $P$ is price level, $Q$ is real transaction quantity.

**Velocity in practice:**

$$V = \frac{\text{Total transaction value per year}}{\text{Average mobile money balance}} = \frac{PQ}{M}$$

**Comparison across systems:**

| System | Annual txn value | Avg balance | Velocity |
|---|---|---|---|
| M-Pesa (Kenya) | \$37.5B | \$2.5B | 15.0x |
| bKash (Bangladesh) | \$12.0B | \$1.6B | 7.5x |
| Bank deposits (US) | \$18T | \$3.6T | 5.0x |

**Float revenue formula:** Annual float revenue at interest rate $r$:

$$R_{\text{float}} = \frac{M}{V} \cdot r = \frac{M \cdot r}{V}$$

**Paradox:** High velocity means heavy usage (good for fee revenue) but low float (bad for interest revenue). Total revenue $R_{\text{total}} = R_{\text{fees}} + R_{\text{float}}$ is maximised at an interior velocity.

**Behavioural insight:** Mobile money velocity of 15x (vs bank deposit 5x) reveals users treat wallets as transaction accounts, not savings vehicles. Access to payments $\neq$ access to savings.

---

**High velocity means heavy usage but low float. The operator optimises total revenue (fees + float) by encouraging transactions while discouraging immediate cash-out.**

# Is a Payment Network Worth $n^2$ or $n \log n$ – and Why Does It Matter?

**Metcalfe's law.** The value of a network with $n$ users is proportional to the number of possible connections:

$$V_{\text{Metcalfe}} = k \cdot \binom{n}{2} = k \cdot \frac{n(n-1)}{2} \approx \frac{k \cdot n^2}{2}$$

**Assumption:** Every pair derives equal value. This overstates value for payment networks where transaction frequency follows a power law.
**Zipf correction (Briscoe et al., 2006).** If the $i$-th most valuable connection has value $\propto 1/i$:

$$V_{\text{Zipf}} = k \cdot n \cdot \sum_{i=1}^{n-1} \frac{1}{i} \approx k \cdot n \cdot \ln(n)$$

**Comparison at scale:**

| **Network size** $n$ | $n^2/2$ | $n \ln n$ | **Ratio** |
|---|---|---|---|
| 100 | 5,000 | 461 | 10.9x |
| 1,000 | 500,000 | 6,908 | 72.4x |
| 1,000,000 | $5 \times 10^{11}$ | $1.38 \times 10^7$ | 36,200x |

**Implication:** Metcalfe's law justifies massive subsidies for early users (e.g., PayPal's $10 sign-up bonus). Zipf's law suggests the marginal user adds diminishing value – growth should eventually be capped.

**Which model fits?** Facebook fits Metcalfe, Tencent fits $n \log n$, payment networks fall between. The answer depends on usage concentration.

---

**Metcalfe's law justifies subsidizing user acquisition. Zipf's law warns that marginal users add less value. The truth determines when to stop spending on growth.**

# How Do You Simulate the S-Curve That Every Payment Network Follows?

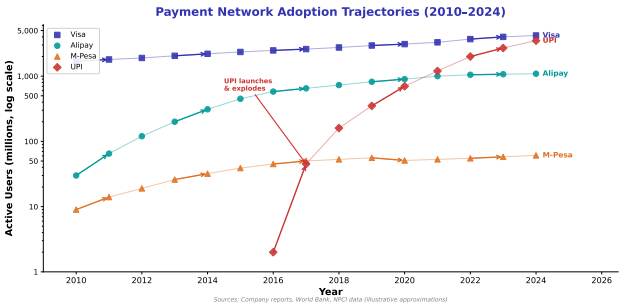
```python
import numpy as np

def logistic_adoption(t, K, r, t0):
    """Logistic growth model for adoption.
    K: carrying capacity (max users)
    r: growth rate
    t0: midpoint (inflection year)
    Returns: users at time t."""
    return K / (1 + np.exp(-r * (t - t0)))

# Simulate three payment networks
years = np.arange(2007, 2030)
mpesa = logistic_adoption(years, 35e6, 0.45, 2013)
upi = logistic_adoption(years, 350e6, 0.55, 2021)
pix = logistic_adoption(years, 150e6, 0.70, 2022)

# Critical mass threshold (10% of K)
for name, K, data in [("M-Pesa", 35e6, mpesa),
                      ("UPI", 350e6, upi),
                      ("Pix", 150e6, pix)]:
    idx = np.argmax(data > 0.1 * K)
    print(f"{name}: critical mass in "
          f"{years[idx]}")
```

**Logistic growth in payment networks**

- **S-curve universality:** Every successful payment network follows the same pattern – slow start, explosive middle, saturation
- **Three parameters:** Carrying capacity $K$ (market size), growth rate $r$ (viral coefficient), midpoint $t_0$ (inflection year)
- **Critical mass:** Where network effects become self-sustaining, typically 10–15% of the target population. Before: requires subsidies; after: organic growth
- **Growth rate comparison:** Pix ($r = 0.70$) reached critical mass faster than M-Pesa ($r = 0.45$) – government mandated participation and zero fees
- **Saturation:** As $n \to K$, growth decelerates. Platform must find new revenue (lending, insurance) or face margin compression
- **Viral coefficient:** Each user recruits others by sending money – a referral loop absent in single-sided products

**The logistic model explains why payment networks appear to "suddenly" succeed after years of slow growth. The inflection point is when network effects kick in.**

# Which Networks Reached Critical Mass Fastest – and What Did They Sacrifice?

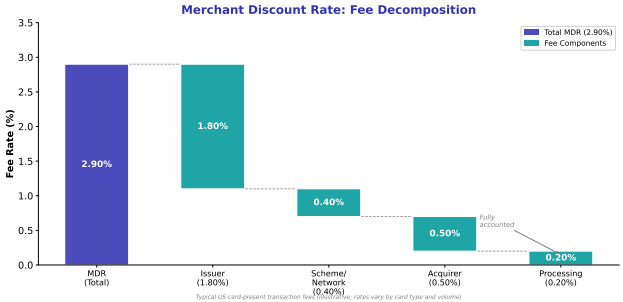



**Adoption trajectories compared**

- **M-Pesa (2007–):** Organic agent-led growth in Kenya. 7 years to 20M users. Safaricom invested $50M+ in agent subsidies
- **UPI (2016–):** Government-mandated interoperability across Indian banks. 5 years to 300M users. Zero merchant fees via government subsidy
- **Pix (2020–):** Brazilian central bank mandated bank participation. 2 years to 140M users. Fastest adoption in payment history
- **WeChat Pay (2013–):** Embedded in social network (1B users). Red envelope 2014 onboarded 100M users in one week
- **Pattern:** Government-mandated interoperability (UPI, Pix) is faster than private networks (M-Pesa) but may undermine commercial sustainability
- **Trade-off:** Speed vs profitability. Free networks grow faster but struggle to monetize; fee-charging networks are self-sustaining

**The fastest path to critical mass is government mandate (Pix, UPI). The most profitable path is organic growth with transaction fees (M-Pesa). You rarely get both.**

# Where Does Each Basis Point of the Merchant Discount Rate Go?




**Fee decomposition: who captures what?**

- **Merchant discount rate (MDR):** Total fee per transaction: 1.5–3.0% for credit cards, 0.5–1.0% for debit
- **Interchange fee:** Largest component (60–70% of MDR). Flows acquirer to issuer. Funds rewards, fraud protection, free banking
- **Scheme fee:** Visa/Mastercard assessment (5–10% of MDR). Funds network infrastructure and brand marketing
- **Acquirer margin:** Retained by merchant's processor (20–30% of MDR). Covers risk underwriting, terminals, settlement
- **EU regulation:** IFR (2015) capped interchange at 0.2% (debit) and 0.3% (credit). Merchant fees fell; consumer rewards reduced
- **US comparison:** Durbin (2010) capped debit at 21c + 0.05%. Credit averaging 1.8–2.2% remains unregulated
- **Zero-fee models:** UPI, Pix, FedNow charge zero interchange. Sustainability depends on cross-subsidization

**Interchange fees are the most contested number in payments. Every basis point represents billions of dollars flowing between merchants, banks, and consumers.**

# How Do You Launch a Network Nobody Wants to Join Because Nobody Has Joined?

**The chicken-and-egg problem.** Merchants won't accept a method no consumers use; consumers won't adopt a method no merchants accept. This coordination failure is the central challenge of every payment network launch.

**Strategies that worked:**

1. **Subsidize one side:** PayPal paid $10–20 per new account (2000). Cost: $60–70M. 5M users in 15 months. The subsidized side depends on relative elasticity
2. **Killer use case:** M-Pesa launched as a remittance tool. P2P transfers required only a sender – the receiver was forced to join to collect
3. **Embed in existing network:** WeChat Pay in a 1B-user messaging app. Alipay in Taobao. No cold-start problem
4. **Government mandate:** Pix required all Brazilian banks with 500,000+ accounts. UPI mandated interoperability across India

**Strategies that failed:**

1. **Build both sides simultaneously:** Google Wallet (2011) tried to sign merchants and consumers at once. Neither reached critical mass. Relaunched as Google Pay (2018) by riding card rails
2. **Premium positioning:** Mondex and Digicash offered superior technology but required new terminals. Tech does not overcome coordination failure
3. **Closed-loop isolation:** Starbucks card succeeded within its ecosystem but could not extend. Closed loops avoid the problem but limit market size

**Critical mass formula:** Network succeeds when per-user benefit $f(n^*) > c$ (cost of joining). Below $n^*$, collapses; above it, self-sustaining.

---

**Every successful payment network solved the chicken-and-egg problem through subsidy, embedding, mandate, or a killer use case. Technology alone is never sufficient.**

# Where Do the Hidden Costs in a Currency Conversion Actually Hide?

**FX spread decomposition.** When a consumer sends $200 from the US to India, the total cost has three additive components:

**1. Transfer fee** (explicit, quoted upfront):

$$C_{\text{fee}} = \alpha + \beta \cdot S \quad \text{where } \alpha \text{ is fixed fee, } \beta \text{ is percentage fee, } S \text{ is send amount}$$

**2. FX markup** (implicit, hidden in exchange rate):

$$C_{\text{FX}} = S \cdot \frac{r_{\text{mid}} - r_{\text{offered}}}{r_{\text{mid}}} = S \cdot \delta$$

where $r_{\text{mid}}$ is the interbank mid-market rate, $r_{\text{offered}}$ is the rate given to the consumer, and $\delta$ is the markup.

**3. Correspondent charges** (deducted in transit, often unknown until arrival):

$$C_{\text{corr}} = \sum_{k=1}^{K} f_k \quad \text{(sum of } K \text{ intermediary bank fees)}$$

**Total cost and effective rate:**

$$C_{\text{total}} = C_{\text{fee}} + C_{\text{FX}} + C_{\text{corr}}, \qquad \text{Effective cost} = \frac{C_{\text{total}}}{S} \times 100\%$$

**Example:** Send $200 via Western Union (US $\rightarrow$ India). Transfer fee: $5 ($\alpha = 5, \beta = 0$). FX markup: $\delta = 2.5\%$, so $C_{\text{FX}} = \$5.00$. No correspondent charges (Western Union is end-to-end). Total: $10 (5.0%).

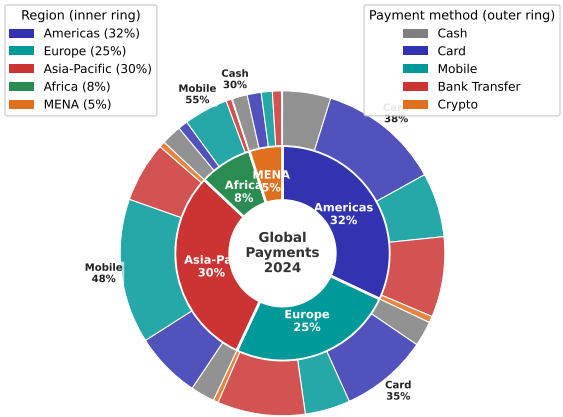
**Transparency problem:** Most providers advertise "zero fee" but embed 3–5% in the exchange rate markup. Wise was the first to show the mid-market rate alongside the offered rate, making $\delta$ visible to consumers.

---

**The FX markup is typically 2–3x the explicit transfer fee. "Zero-fee" transfers are almost never zero-cost.**

# Which Payment Rails Carry the Most Value – and Which Carry the Most Transactions?

**Payment Volume by Region & Method — Sunburst (2024)**




**Volume vs value decomposition**

- **Value concentration:** RTGS (Fedwire, TARGET2, SIC) carry 80–90% of value but <1% of transaction count. Single Fedwire transfer averages $4.5M
- **Transaction concentration:** Cards and mobile carry 80%+ of count but small share of value. Average Visa transaction: $85
- **Mobile money niche:** M-Pesa average $12. 40M+ transactions/day but daily value under $500M
- **Cash persistence:** 60–80% of transactions in emerging markets, 15–20% in advanced economies. Invisible in electronic data
- **Real-time growth:** UPI, Pix, SEPA Instant are fastest-growing, cannibalizing card and ACH
- **Cross-border gap:** 1–2% of count but 20–30% of value and 40–50% of revenue

**Cross-border payments are 1% of transactions but 40% of revenue. This explains why every FinTech wants to disrupt them.**

# Can You Compute the True Cost When Providers Hide Half of It?

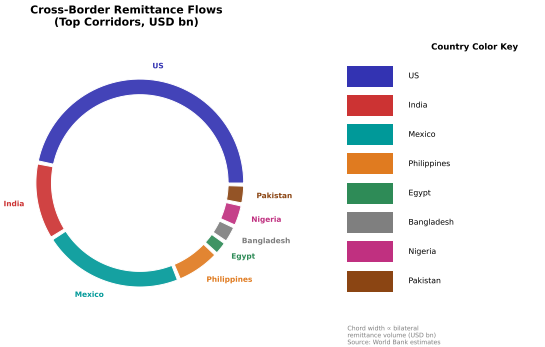
```python
def remittance_cost(send_amt, flat_fee,
                    pct_fee, mid_rate,
                    offered_rate,
                    corr_fees=None):
    """Compute total remittance cost.
    send_amt: amount sent in source currency
    flat_fee: fixed transfer fee
    pct_fee: percentage fee (0.01 = 1%)
    mid_rate: interbank mid-market FX rate
    offered_rate: rate given to customer
    corr_fees: list of correspondent charges
    Returns: dict with cost breakdown."""
    fee = flat_fee + pct_fee * send_amt
    fx_markup = send_amt * (mid_rate -
                offered_rate) / mid_rate
    corr = sum(corr_fees or [])
    total = fee + fx_markup + corr
    received = (send_amt - fee - corr) \
               * offered_rate
    return {"fee": fee, "fx_markup": fx_markup,
            "corr": corr, "total": total,
            "pct": total / send_amt * 100,
            "received": received}
```

**Decomposing the true cost**

- **Three cost layers:** Explicit fee (visible), FX markup (semi-hidden), and correspondent charges (fully hidden until arrival)
- **Why decomposition matters:** A provider advertising "$0 fee" with a 4% FX markup is more expensive than one charging $5 with mid-market rates on a $200 transfer
- **Mid-market rate source:** Bloomberg, Reuters, or XE.com provide the interbank reference rate. The difference between this and the offered rate is pure profit for the provider
- **Correspondent opacity:** SWIFT transfers can pass through 2–4 correspondent banks, each deducting $10–25. The sender cannot predict the final received amount
- **Regulatory response:** EU Payment Services Directive (PSD2) requires upfront disclosure of total cost including FX markup. US has no equivalent requirement for remittances
- **Comparison tool:** The World Bank's Remittance Prices Worldwide database tracks costs across 365 corridors and 4,000+ provider-corridor combinations quarterly

**True cost = explicit fee + FX markup + correspondent charges. Most comparison sites show only the first. The World Bank database is the gold standard for full-cost data.**

# Which Corridors Dominate Global Remittance Flows – and Why?

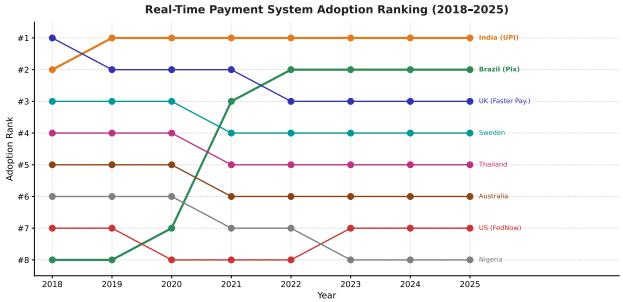



**Bilateral remittance flow patterns**

- **Top corridor:** US → Mexico ($63B annually). Driven by 11M+ Mexican-born residents in the US. Remittances exceed Mexico's oil export revenue
- **GCC–South Asia:** UAE/Saudi → India, Pakistan, Bangladesh. 25M+ migrant workers sending 30–50% of earnings home
- **Intra-Africa:** South Africa → Zimbabwe, Nigeria → Ghana. Most expensive corridors despite shortest distances, due to regulatory barriers and limited competition
- **Europe–North Africa:** France → Morocco, Italy → Tunisia. Post-colonial migration patterns drive flows
- **Total market:** $656B in officially recorded remittances (2024). Informal channels (hawala, hand-carry) add an estimated 30–50% more
- **Concentration:** Top 20 corridors account for 40% of global remittance value. The long tail of 300+ smaller corridors is where costs are highest
- **Digital disruption:** Digital remittances grew from 15% to 40% of volume (2019–2024). Mobile-to-mobile corridors (M-Pesa to MTN) in East Africa lead digital penetration

**Remittances ($656B) exceed foreign direct investment to developing countries. They are the largest source of external finance for many nations.**

# Why Did Real-Time Payments Explode in Some Countries and Stall in Others?




**Real-time payment scheme adoption**

- **India (UPI):** 0 to 10B monthly transactions in 7 years. Government mandated participation + zero-fee subsidy. World's largest real-time payment system
- **Brazil (Pix):** Central bank mandated all banks with 500,000+ accounts. 150M users in 2 years. Free for individuals, near-free for merchants
- **UK (Faster Payments):** Pioneer (2008) but slower curve. Voluntary participation limited reach initially. Now 4B+ transactions/year
- **US (FedNow / RTP):** Late entrant (FedNow 2023). Fragmented between FedNow and RTP (The Clearing House). Voluntary bank adoption
- **Success factors:** (1) Mandate vs voluntary, (2) free vs fee-based, (3) card-dominant vs cash-dominant market
- **Cannibalization:** Banks fear RTP cannibalizes card interchange revenue – explains slow voluntary adoption in card-heavy markets

**Mandated participation + zero fees = fastest adoption (UPI, Pix). Voluntary participation + fees = slow adoption (US, EU). Incentive design determines outcomes.**

# Why Is the 50-Year-Old Correspondent Banking Model Still Alive?

**Correspondent banking:** A bank (respondent) holds an account at another bank (correspondent) in a foreign jurisdiction. Cross-border payments route through chains of these bilateral relationships.

**Why it persists:**

- **Regulatory compliance:** Correspondent banks perform KYC/AML on behalf of respondents. Replacing them requires replicating this compliance infrastructure
- **Liquidity provision:** Correspondents provide intraday credit and FX liquidity. Alternatives must solve the liquidity problem, not just the messaging problem
- **Network density:** 11,000+ banks connected via 1M+ correspondent relationships. Building an alternative network from scratch is a chicken-and-egg problem
- **Legal certainty:** Correspondent banking operates under established legal frameworks (UCP 600 for trade finance, New York law for USD clearing). Novel systems face legal uncertainty

**Emerging alternatives:**

- **SWIFT gpi (2017):** Not a new network but an upgrade — adds end-to-end tracking, SLA commitments, and pre-validation. 90% of gpi payments credited within 24 hours
- **Wise (TransferWise):** Avoids correspondents entirely by matching currency flows locally. Holds local accounts in 80+ countries. Cost: 0.5–1.5%
- **Ripple / XRP:** Uses cryptocurrency as a bridge currency to eliminate pre-funded nostro accounts. Adoption limited to niche corridors (Philippines → Mexico)
- **Project mBridge (BIS):** Multi-CBDC platform connecting central banks of China, Thailand, UAE, and Saudi Arabia. Direct central-bank-to-central-bank settlement without correspondents. Pilot completed 2024
- **Stablecoin rails:** USDC and USDT enable near-instant cross-border value transfer. Regulatory status uncertain; compliance infrastructure nascent

**De-risking crisis:** Since 2012, major correspondent banks have exited 25% of their relationships (BIS, 2024). Small countries and high-risk jurisdictions are losing access to the global payment system. Alternatives are not replacing correspondent banking — they are filling gaps left by its retreat.

**Correspondent banking is expensive and slow, but it provides compliance, liquidity, and legal certainty. Alternatives must solve all three, not just speed.**

# How Do You Measure Something as Multidimensional as Financial Inclusion?

**Composite index construction.** Financial inclusion spans access (can you reach a service?), usage (do you use it?), and quality (does it meet your needs?). A composite index aggregates multiple indicators into a single score.

**Weighted composite indicator:**

$$I_j = \sum_{i=1}^{d} w_i \cdot x_{ij}$$

where $I_j$ is the score for country $j$, $w_i$ is the weight for dimension $i$, $x_{ij}$ is the normalized indicator, and $d$ is the number of dimensions.
**Normalization** (min-max scaling to $[0, 1]$):

$$x_{ij} = \frac{X_{ij} - X_i^{\min}}{X_i^{\max} - X_i^{\min}}$$

**World Bank Findex dimensions (Demirguc-Kunt et al., 2022):**

| Dimension | Indicator | Weight |
|---|---|---|
| Access | Account ownership (% age 15+) | $w_1 = 0.25$ |
| Access | Mobile money account (% age 15+) | $w_2 = 0.15$ |
| Usage | Digital payment made (% age 15+) | $w_3 = 0.20$ |
| Usage | Saved at financial institution (%) | $w_4 = 0.15$ |
| Quality | Borrowed formally (% age 15+) | $w_5 = 0.15$ |
| Quality | Financial resilience (emergency funds) | $w_6 = 0.10$ |

**Constraint:** $\sum_{i=1}^{d} w_i = 1$. Weight selection is inherently subjective – equal weights assume all dimensions matter equally; expert weights reflect policy priorities.

**Financial inclusion indices aggregate access, usage, and quality. Weight selection determines whether mobile money "counts" as meaningful inclusion.**

# Can You Build a Financial Inclusion Score from Raw World Bank Data?

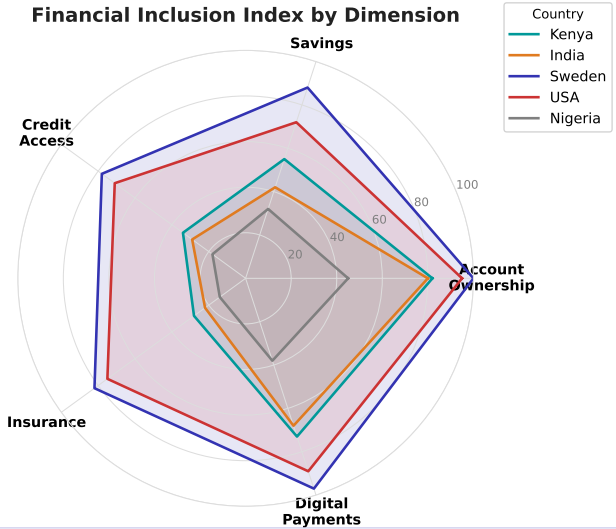
```python
import numpy as np

def inclusion_index(indicators, weights):
    """Compute composite financial inclusion.
    indicators: dict {name: raw_values array}
    weights: dict {name: weight}
    Returns: normalized index per country."""
    norm = {}
    for name, vals in indicators.items():
        lo, hi = vals.min(), vals.max()
        norm[name] = (vals - lo) / (hi - lo)
    score = sum(weights[n] * norm[n]
                for n in indicators)
    return score

# Sample data (5 countries, 4 indicators)
data = {"account": np.array([.95,.82,.71,.55,.32]),
        "digital": np.array([.90,.75,.60,.40,.18]),
        "savings": np.array([.55,.40,.30,.20,.08]),
        "mobile": np.array([.05,.12,.45,.65,.72])}
w = {"account":.30, "digital":.25,
     "savings":.25, "mobile":.20}
idx = inclusion_index(data, w)
labels = ["Sweden","Brazil","India","Kenya","Niger"]
for c, s in zip(labels, idx):
    print(f"{c}: {s:.3f}")
```

**Building a composite index**

- **Min-max normalization:** Maps each indicator to $[0, 1]$ so dimensions with different scales (percentages, counts, ratios) are comparable
- **Weight sensitivity:** Increasing the mobile money weight from 0.20 to 0.40 would move Kenya from rank 4 to rank 2. The choice of weights embeds a policy judgment about what counts as inclusion
- **Kenya paradox:** Low on traditional metrics (account, savings) but highest on mobile money. Whether Kenya is "financially included" depends entirely on how you weight mobile wallets
- **Sweden ceiling effect:** Scores near 1.0 on traditional metrics but near 0 on mobile money (no need for it). High inclusion score masks the fact that 5% are digitally excluded
- **Robustness check:** Good practice is to report the index under multiple weight vectors and check whether country rankings are stable. Unstable rankings signal that the index is measuring the weights, not the countries

**A composite index is only as meaningful as its weights. Always check sensitivity: if changing weights flips the ranking, the index is measuring assumptions, not reality.**

# Which Countries Excel on Access but Fail on Quality?

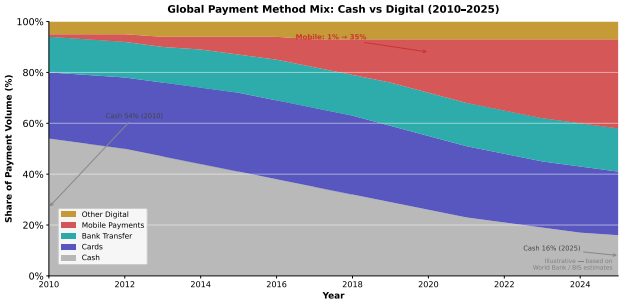



**Multi-dimensional inclusion profiles**

- **Radar chart logic:** Each axis = one dimension. Area covered reveals both level and balance of financial inclusion
- **Nordic pattern:** Near-circular, high-area. High and balanced across all dimensions. Little room for improvement
- **Kenya pattern:** Spiky – high on mobile money and digital payments, low on formal savings and credit. Access without depth
- **India pattern:** Account ownership surged after Jan Dhan Yojana (500M+ accounts). But 40% dormant – access without usage
- **Sub-Saharan Africa:** Mobile money axis dominates. Traditional banking near zero. Inclusion is mobile-money-driven
- **Access vs quality gap:** Findex 2021: 25% of account holders made zero transactions in the past year
- **Policy:** Target usage and quality, not just account opening. Quantity without quality is an expensive illusion

**A high inclusion score with a spiky profile means access without depth. True inclusion requires a balanced, circular radar shape across all dimensions.**

# Is the Death of Cash a Liberation or an Exclusion?

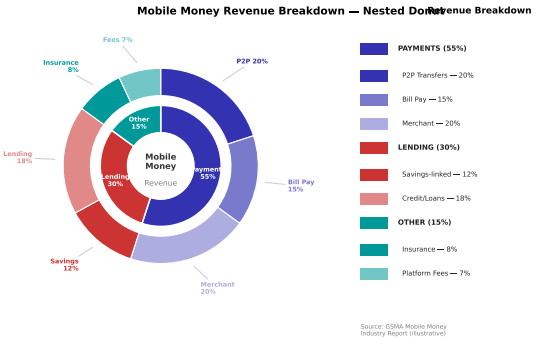



**Cash displacement patterns**

- **Sweden leads:** Cash is below 10% of point-of-sale transactions. The Riksbank is developing e-krona (CBDC) partly to ensure a public alternative to private digital money
- **China's mobile leap:** Cash went from 60% to under 20% in a decade, driven by Alipay and WeChat Pay. QR codes replaced cash, not cards
- **India's forced experiment:** Demonetization (2016) removed 86% of cash overnight. Digital payments surged but cash usage recovered within 18 months for the poorest segments
- **Cash resilience in emerging markets:** Sub-Saharan Africa, South Asia, and Latin America remain 60–80% cash. Informality, distrust of digital systems, and agent liquidity constraints sustain cash
- **The exclusion risk:** Cashless societies risk excluding the elderly, unbanked, digitally illiterate, and undocumented populations. Sweden legislated that banks must provide cash services
- **Privacy trade-off:** Cash is anonymous; digital payments create surveillance records. Central banks designing CBDCs must balance AML compliance with privacy rights

**The cashless transition is fastest where it is least needed (Nordics) and slowest where it would help most (Sub-Saharan Africa). Infrastructure, not preference, drives the gap.**

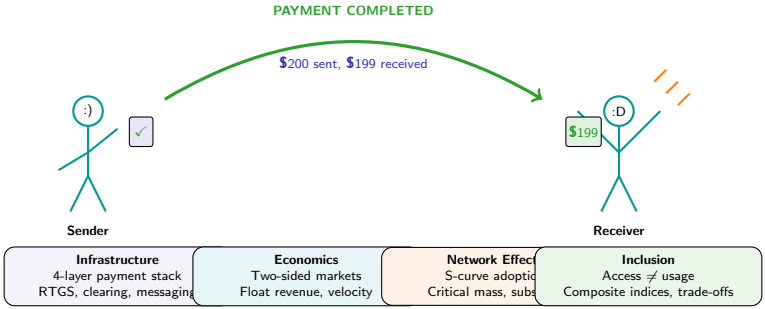
# Where Does the Revenue Come from When the Average Transaction Is $12?




**Mobile money revenue mix**

- **Transaction fees:** 55–65% of revenue. Tiered pricing: P2P transfers 1–3%, cash-out a flat fee. Higher-value transactions subsidize smaller ones
- **Float interest:** 10–15% of revenue. Significant at high rates (Kenya: 8–10% treasury). Near zero in low-rate markets
- **Merchant payments:** 8–12% of revenue. "Lipa na M-Pesa" replaces retail cash. Lower fee (0.5–1%) but higher volume
- **Lending (M-Shwari, Fuliza):** 10–15% and growing fastest. Micro-loans $5–50 using transaction history as credit score. Annualized rates: 40–100%
- **International remittances:** 3–5% of revenue. Higher margin (2–4% vs 1% domestic) but lower volume
- **Data and API services:** Emerging. Anonymized data sold to FMCG for demand forecasting; API access for third-party developers
- **Super-app trajectory:** Payments → savings → credit → insurance → investment. Revenue diversification follows WeChat/Alipay

**Transaction fees built the business. Lending will define its future. The super-app trajectory turns a payment rail into a full financial services platform.**




*The payment took 3 seconds. Understanding why it works — and for whom it does not — took 32 slides.*