Vantis: The "Buy & Keep" Card — Smart Credit on Soroban

1. Executive Summary

Vantis is a non-custodial financial product built on the **Stellar** network that allows users to collateralize digital assets (Bitcoin, Stellar Lumens, yXLM) to fund consumer purchases via a white-label credit card.

Unlike crypto-debit cards that require asset liquidation at the point of sale, Vantis utilizes a **"Just-in-Time" (JIT)** funding mechanism linked to Soroban smart contracts. By integrating **Rain** as the card issuer and leveraging **Stellar Anchors** for repayment, Vantis connects on-chain yield generation with Visa payment rails. This enables users to borrow against their assets for spending and repay using standard bank transfers, maintaining their underlying crypto positions.

2. Problem Statement: Limitations of the Prepaid Model

Current crypto-payment solutions typically rely on a "Prepaid Model," which presents specific economic constraints:

- **Tax Implications:** Prepaid cards trigger immediate asset liquidation at the point of sale, creating capital gains tax events for each transaction.
- **Opportunity Cost:** Users reduce their asset exposure to fund daily purchases, forfeiting potential future appreciation.
- **Liquidation Risks:** Standard DeFi lending platforms often employ strict liquidation parameters that can result in asset loss during short-term market volatility.

3. The Solution: Credit with Fiat Repayment

Vantis operates on a secured credit model. Users borrow against their crypto assets rather than selling them.

1. **Non-Custodial Credit:** Users deposit assets into a Smart Account they control, utilizing OpenZeppelin standards on Soroban.
2. **Yield Offset:** The architecture utilizes yield-bearing collateral (e.g., yXLM) to offset borrowing costs.**Yield Mechanism:** If the yield generated by the collateral ($C \times r_{yield}$) exceeds the borrowing cost ($P \times r_{borrow}$), the effective interest rate becomes negative.
3. **Traditional Rails Repayment:** Users pay monthly balances using standard fiat rails (ACH, SEPA, Bank Transfer). Vantis leverages **Stellar Anchors** to convert this fiat into

USDC to settle the on-chain debt.

4. Technical Architecture

4.1 Core Components

The architecture consists of four primary components:

1. **Stellar Smart Accounts (User Custody):** Leverages OpenZeppelin’s Smart Account standards. The user retains custody but grants **Rain** (via the Vantis backend) restricted "Signer" privileges to initiate borrow transactions.
2. **Blend Finance (Lending Engine):** An isolated lending pool where users deposit collateral and borrow USDC.
3. **Rain (Payment Rails):** Acts as the spending bridge. Rain approves transactions via webhook and accepts USDC for settlement directly on the Stellar network.
4. **Stellar Anchors (Repayment Rails):** Uses **SEP-24 (Hosted Deposit and Withdrawal)** to integrate with regulated financial institutions (Anchors). The Anchor issues stablecoins directly to the user's Smart Account to repay the Blend loan upon receipt of fiat.

4.2 Stellar Primitives for Repayment

Vantis utilizes specific Stellar features for the repayment flow:

- **SEP-24 (Interactive Anchors):** Facilitates the connection between the user's bank account and the Stellar network.
- **Path Payments (Strict Send):** Enables atomic conversion if the repayment currency differs from the debt currency (e.g., EUR stablecoin to USDC) in a single transaction.

4.3 System Context Diagram

```
graph TD
  User[User / Cardholder]

  subgraph "Vantis Protocol (Stellar Network)"
    SA[Smart Account - User Controlled]
    Blend[Blend Lending Pool]
    Oracle[Reflector Oracle]
  end

  subgraph "Off-Chain Infrastructure"
    VantisBE[Vantis Backend / Decision Engine]
    Rain[Rain Issuer Platform]
  end

  subgraph "Payment Network"
    Visa[Visa Network]
    Merchant[Merchant POS]
```

```
      Bank[Traditional Banking Rails]
    end

    subgraph "Stellar Anchors"
      Anchor[SEP-24 Anchor]
    end

    %% Spending Flow
    User -- Deposits Collateral --> SA
    SA -- Staked in --> Blend
    User -- Swipes Card --> Merchant
    Merchant -- Auth Request --> Visa
    Visa -- ISO 8583 --> Rain
    Rain -- Webhook JIT Funding --> VantisBE
    VantisBE -- Checks LTV --> SA
    SA -- Borrows USDC --> Blend
    Blend -- Sends USDC --> Rain
    Rain -- Settles --> Visa

    %% Repayment Flow
    User -- Pays Bill (Fiat) --> Bank
    Bank -- Wire/ACH --> Anchor
    Anchor -- Mints Stablecoin --> SA
    SA -- Auto-Repay Debt --> Blend
```

5. Asset Protection & Risk Engine

Vantis implements specific risk parameters designed to prioritize asset retention.

5.1 Liquidation Mitigation Architecture

The protocol employs a multi-layered system to manage collateral risk:

Layer 1: Volatility-Adjusted LTV (Prevention)

The protocol calculates a **Safe Borrow Limit ($B_{safe}$)** based on the asset's historical volatility ($\sigma$), rather than a static Loan-to-Value ratio.

$$B_{safe} = V_{collateral} \times (LTV_{base} - k\sigma \sqrt{T})$$

- **Function:** Automatically reduces borrowing power for new transactions during periods of high volatility.

Layer 2: The Yield Cushion (Passive Defense)

Collateral is held in yield-bearing assets (yXLM), increasing the collateral balance over time through accrued interest.

- **Function:** Improves the Health Factor incrementally independent of price action.

Layer 3: Automated Stop-Loss (Active Defense)

Users can opt-in to a pre-liquidation swap protocol. If the asset price drops to a critical level:

1. The Smart Account automatically sells a portion of the volatile asset (XLM) into stablecoin (USDC).
2. This action reduces debt exposure without incurring the **5-10% liquidation penalty** associated with external liquidations.

5.2 Risk Mitigation Sequence

The protocol follows a structured sequence to restore account health.

```
graph TD
    State1[Healthy State <br/> Hf > 1.1] -->|Market Drop| State2
    State2[Warning State <br/> 1.0 < Hf < 1.1] -->|User Notified| Alert[Push Notification: <br/> 'Top Up or Repay']

    State2 -->|Continued Drop| State3
    State3[Pre-Liquidation Trigger <br/> Hf ~= 1.02] -->|Auto-Swap| Stable[Swap XLM to USDC <br/> No Penalty Fee]

    State3 -->|Flash Crash| State4
    State4[Partial Liquidation <br/> Hf < 1.0] -->|Dutch Auction| Restore[Sell ONLY enough to reach Hf 1.05]

    style State1 fill:#e6fffa,stroke:#38b2ac,stroke-width:2px
    style State2 fill:#fffaf0,stroke:#ed8936,stroke-width:2px
    style State3 fill:#fff5f5,stroke:#fc8181,stroke-width:2px
    style State4 fill:#feb2b2,stroke:#c53030,stroke-width:2px
```

5.3 Partial Liquidation Mechanism

In the event a liquidation is required (Health Factor $< 1.0$), Vantis utilizes **Partial Liquidation**.

- **Mechanism:** The protocol sells only the minimum amount of collateral required to restore the Health Factor to **1.05**.
- **Outcome:** Prevents total position closure, retaining the remaining collateral balance.

6. Key Benefits

By integrating traditional repayment rails with risk management protocols, Vantis offers specific advantages:

6.1 Tax Efficiency

- **Asset Retention:** Users repay loans with fiat income, avoiding the need to sell crypto assets to cover debt.
- **Non-Taxable Events:** Loan repayment is generally not a taxable event, unlike the sale of assets.

6.2 Operational Integration

- **Standardized Payment Flow:** The user experience mirrors a standard credit card billing cycle (spend, then pay via bank transfer).
- **Automated Management:** The protocol handles risk parameters, volatility adjustments, and yield routing via smart contract logic.

7. Implementation Roadmap

Phase 1: POC (Proof of Concept)

- **Goal:** Validate technical feasibility of the borrow loop on Testnet.
- **Smart Contracts:** Deploy OpenZeppelin Smart Accounts and an isolated Blend lending pool (USDC/yXLM).
- **Issuer Simulation:** Connect Vantis Backend to Rain's Developer Sandbox.

Phase 2: MVP (Minimum Viable Product)

- **Goal:** Production launch with core assets.
- **Mainnet Deployment:** Deploy finalized Smart Account factories.
- **Rain Production:** Switch to Rain production environment with real USDC settlement.
- **Anchor Integration:** Integrate with a primary Stellar Anchor to enable Fiat On-Ramp for repayment.

Phase 3: V1 (Version 1)

- **Goal:** Scaling and automation.
- **Automated Stop-Loss:** Implement the pre-liquidation logic in Smart Account policies.
- **Automated Repayment:** Implement logic where the Smart Account detects incoming Anchor payments and automatically routes them to the Blend debt repayment function.

8. Conclusion

Vantis utilizes **Soroban smart contracts**, **Rain's native stablecoin support**, and **SEP-24 Anchors** to provide an alternative to the traditional "sell-to-spend" model. By allowing users to spend against their assets while employing **Volatility-Adjusted LTV** and **Automated Stop-Losses**, Vantis combines liquidity access with decentralized collateral management.