

SwiftTrade – an AI-based trading platform

Peter Homola, 2025

Motivation

- Build a backtesting platform
 - Something like Python's *Backtesting*
- But in Swift

Rationale

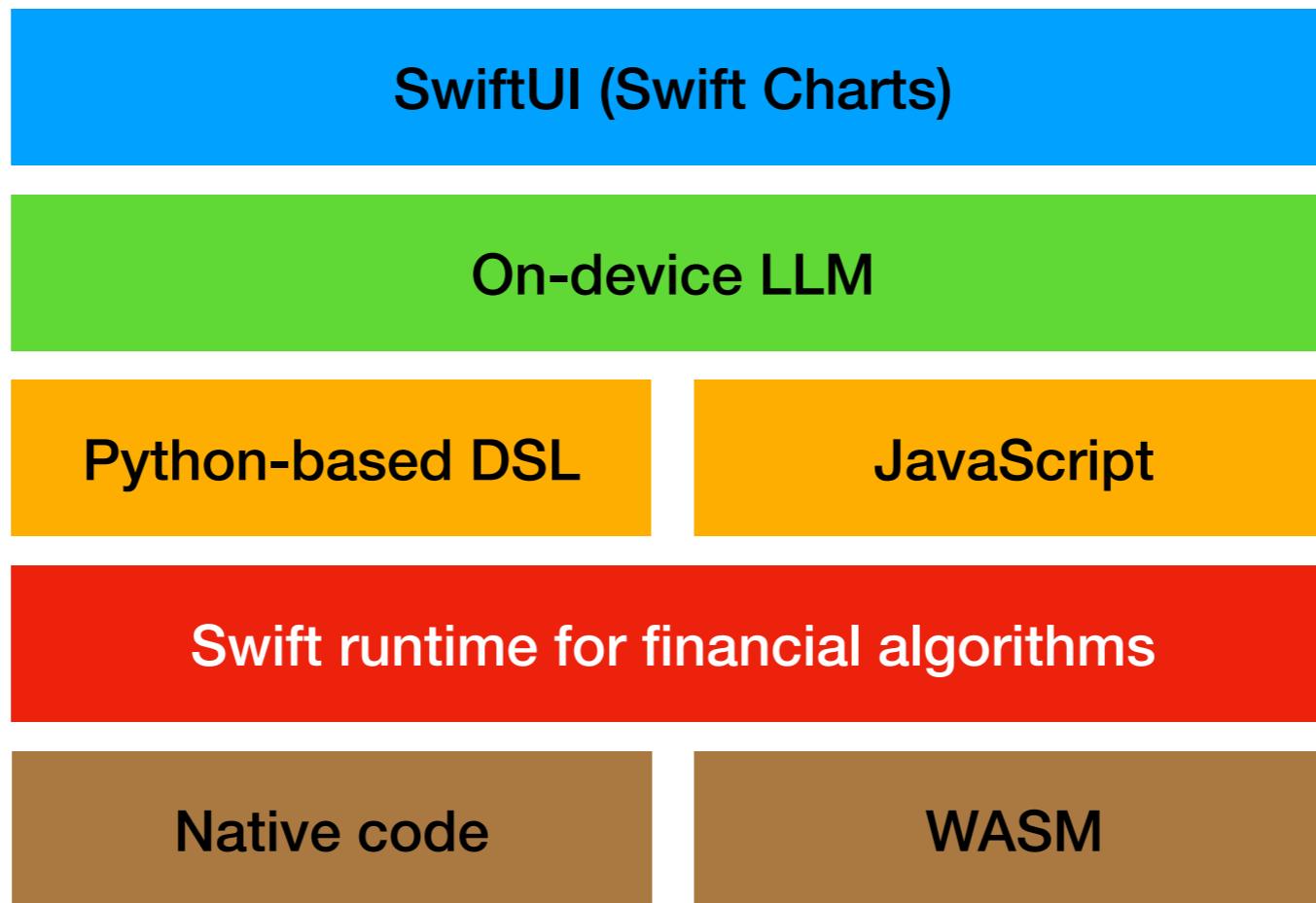
Why Swift?

- Very fast (comparable to Rust, Mojo)
- Open source
- Runs also on Linux, Windows & Android
- Compilable into fast WASM
- Has built-in fast maths library (SIMD, GPU)
- Has built-in UI framework with (candlestick) charts
- Has built-in actors
- Has built-in support for LLMs & “MCP” tools

Architecture

- User interface
 - SwiftUI & Swift Charts
- Strategy building
 - On-device LLM
- Financial data, algorithms & indicators via “MCP” tools
- SwiftData for persistent data
- Python-like DSL run via Mojo

Modules



Runtime

For algorithms & indicators

- Two languages
 - Python-like (based on Mojo)
 - JavaScript (via JavaScriptCore)
- Two modes of execution
 - Compiled (into native code)
 - Interpreted

Technical details

- Swift Concurrency
 - For asynchronous real-time data fetching
- Swift Combine
 - For observability in trading algorithms
- SwiftWasm
 - For running in WASM environments

Project output

- UI for financial data & algorithms
- “Numpy” & financial library for Swift
- “Numpy” & financial library for Mojo
- Precompiled modules for Linux, Windows & WASM