DSL for stock market

Filip Plata

Introduction

What are stock markets?

Shop for people to trade financial instruments

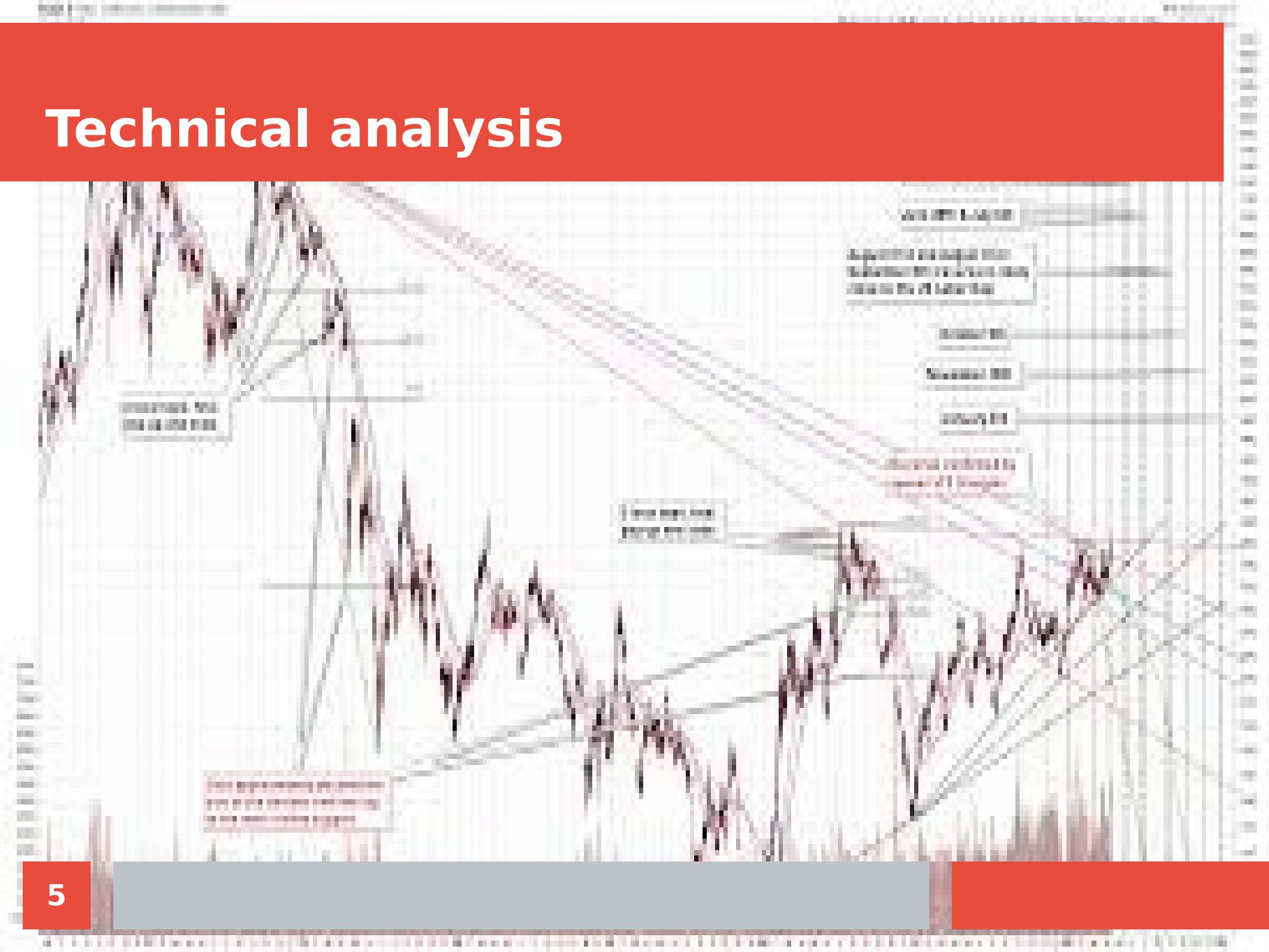
Most popular instrument is a small part of a company's ownership called stocks

People can buy and sell stocks

Investors in Poland

100,000 people in Poland

Technical analysis



Technical analysis

Set of tools used to predict future prices of shares/stocks, based only on the past values

Technical analysis

- **Popular with investors in Poland** **DOI: 10.33119/KKESSiP.2015.3.3.11**
- **Relatively easy to automate**
- **Requires stock data**
- **Requires tools to visualize the data**

Problems with TA

What is wrong with market in Poland?

- **Data is hard to obtain**
- **Tools are inflexible, expensive and often outdated**

TA in Poland is at mercy of highly overpriced tools from high street brokers

Problem with data in Poland

- **Getting it from stock exchange directly costs 24000 PLN per year, and requires agreement with exchange**
- **Website scrappers get broken by frequent changes of html**
- **Data available in form of files for download often has it own special format, and always it is prohibited to use it comercially**
- **For a private person to work with this data, it requires solid programming skills**

Inflexibility of tools

If a technique cannot be performed with the tool, then investor is out of luck

Tools receive little maintanance for years

Experimenting is very hard in such environment

Aim

Providing a flexible service of stock data at a fair price in Poland

Having 5000 MAU in a year

- Flexible means user can extend the tool itself
- Fair price means having it proportional to the users demands, no high bariers and large monthly subscriptions

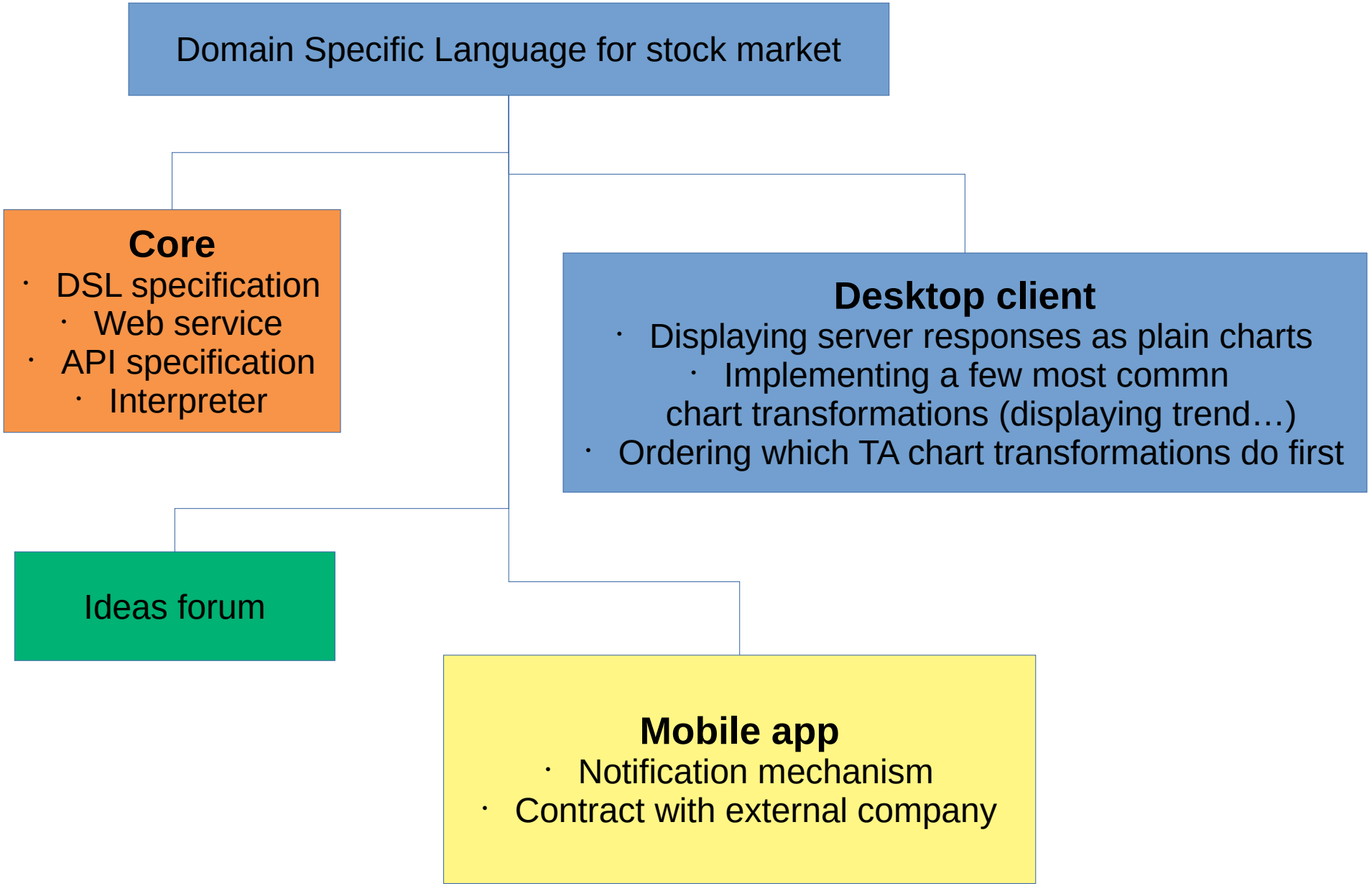
Novelty of approach

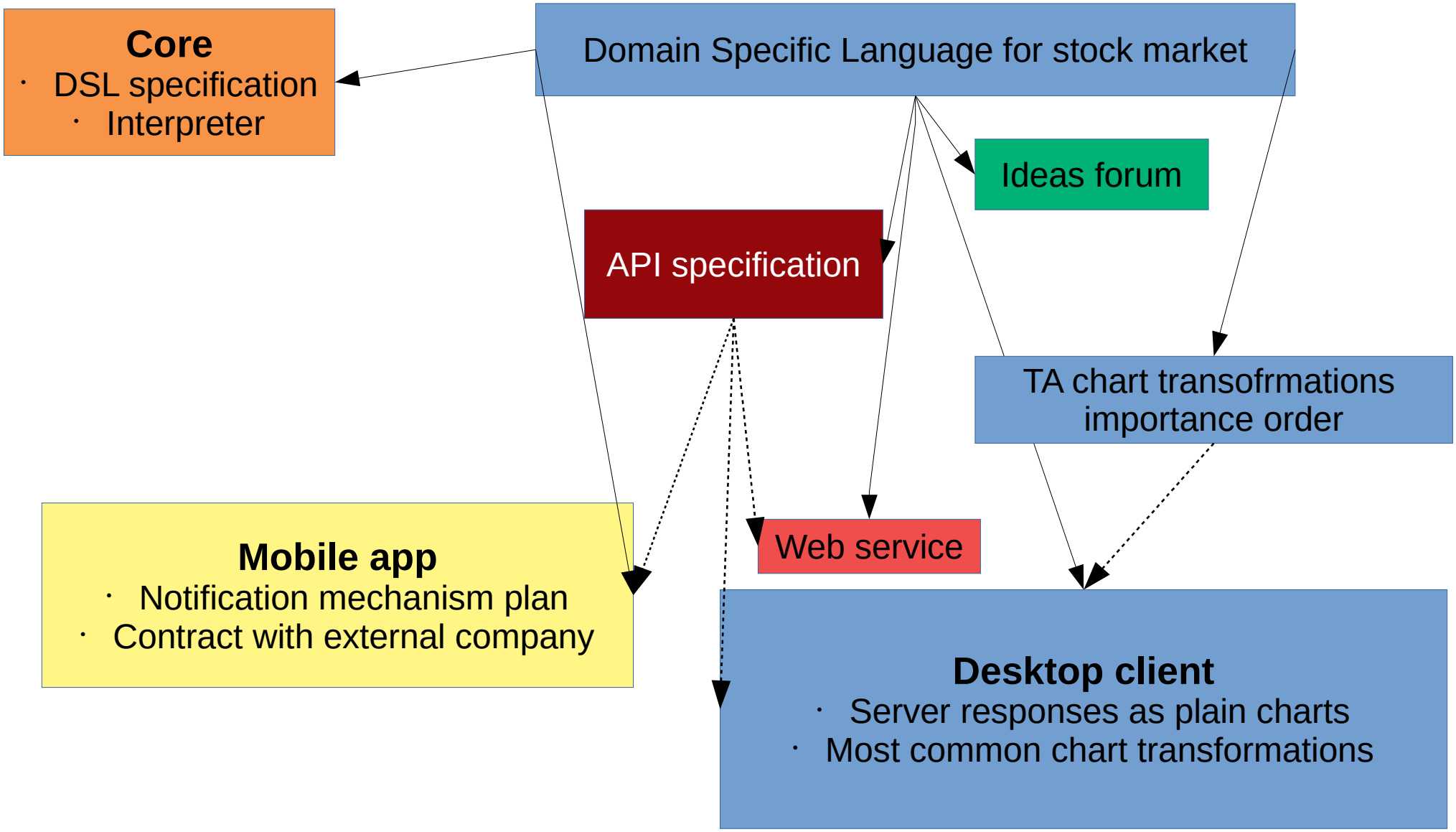
Core idea is creating a DSL for stock and an execution environment for it – with data attached

This solves both data and inflexibility problem

The tool should be engine for execution of a DSL, not a premade set of buttons and charts

Providing DSL execution on cloud would also allow for easy pricing based on resources used by user


Domain Specific Language for stock market
Core
· DSL specification
· Web service
· API specification
· Interpreter
Desktop client
· Displaying server responses as plain charts
· Implementing a few most commn chart transformations (displaying trend…)
· Ordering which TA chart transformations do first
Ideas forum
Mobile app
· Notification mechanism
· Contract with external company


Core
· DSL specification
· Interpreter
Domain Specific Language for stock market
Ideas forum
API specification
TA chart transofrmations
importance order
Mobile app
· Notification mechanism plan
· Contract with external company
Web service
Desktop client
· Server responses as plain charts
· Most common chart transformations

What we omit

The tool will not execute trades itself, only drive the process

Executing trades in somebody’s name poses legal and technical challenges

Challenges and quality

Will Polish stock exchange will agree to sell data for such project?

Will a high street bank or fintech will not start creating their own tool?

The core of the tool, which will process data must be stable and of high quality

Rest of the project needs only not to deter people from using it

Approaches

Buying technology in Poland

Outdated technology will be a hindrance, user bases are insignificant

Creting a product from scratch

Would require money and time

Due to quality requirement necessary for the core

Trying to partner with high street broker after having working MVP

Huge risk that we will not find partner, much smaller risk of getting kicked out of business

Partnership with high street broker

High risk of getting kicked out of business

Easier access to data and customers

Selected approach

Partnership with broker like BOSSA as the first necessary step

Mitigates risk of competion and gives foot in the door of stock exchange people

In case of failure, no huge losses

Overview of the solution

- DSL + execution engine – core of the project
- Web service for the language execution
- Desktop interface
- Simple mobile app for notifications
- Space for people to share their ideas

Risks

Project uncertainties

Risks:

- Developer of language core will leave – 30%, ~45,000 PLN
- Misunderstanding with mobile app provider – 30%, ~10,000 PLN
- Totally unreliable mobile app provider – 2%, ~50,000 PLN (one month of running project more)
- Polish stock market will stop selling us data – 3%, ~300,000 PLN (whole project for nothing)
- Partner broker will dump the project – 15%, ~300,000 PLN (again, project for nothing)

Summary of unexpected problems

~70,500 PLN

Business case

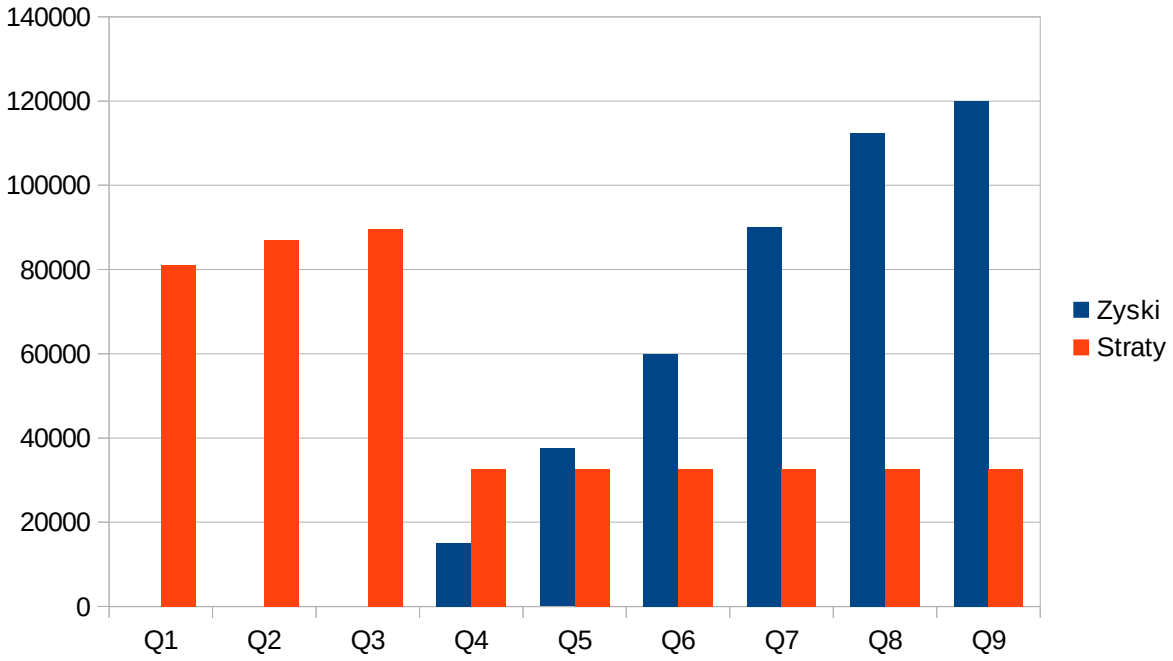
Outflow of money

- **Core developer for 7 months ~ 15,000 PLN each month**
- **GUI developer~ 9,000 PLN each month for 7 months**
- **Mobile app contract – legal advice + payment costs ~15,000 PLN in month 3**
- **~500$ each month for a year from launch – starting at month 8**
- **One junior programmer for maintanence after launch – from 8 month onward, ~7,000 PLN per month**

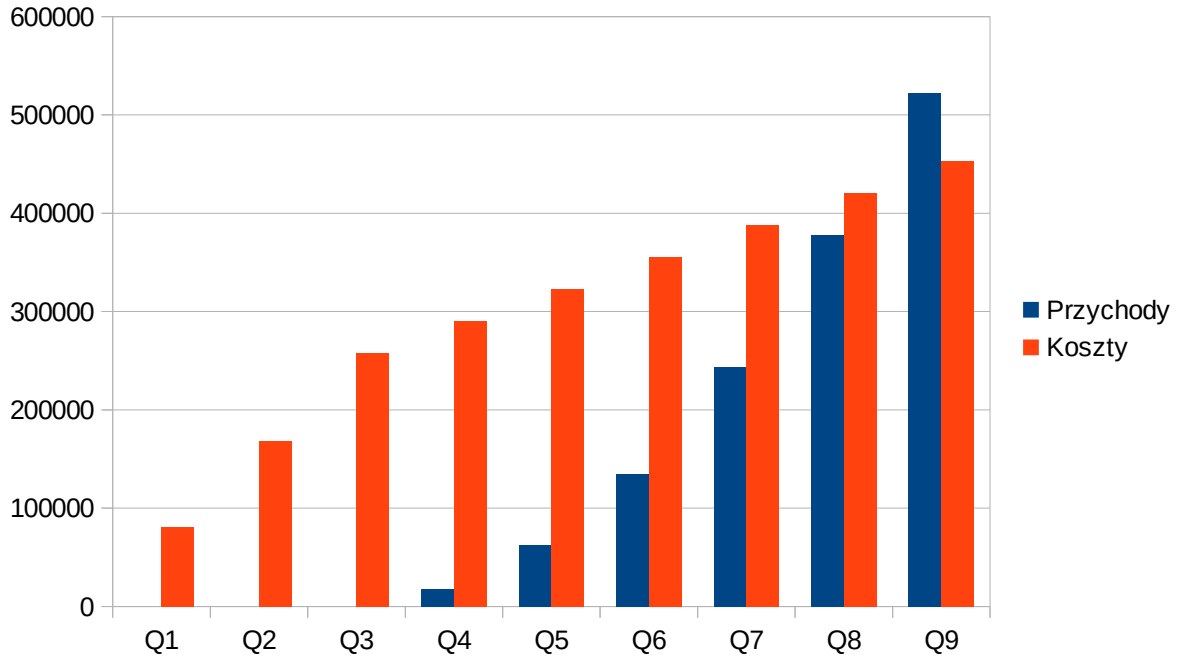
Outflow of money cd.

- **Marketing ~50,000 PLN, at launch – month 8**
- **Costs of managing at ~4,000 PLN each month of the project – apart from one employee**

PLN vs quaters



PLN vs quaters cummulated



Quality

- **Unit and integration tests as in any decent software**
- **DSL core must be covered by tests much more then the GUI**
- **Audit of core DSL code at the end**
- **Tests run on every change, at least 30% coverage for GUI, 80% for GUI**
- **Code review required for changes**

Communication

- During first stage, daily for ~15 minutes every day
- In consecutive stages weekly whole team meetings for about 1 hour, and no daily. The reasoning is that people will know what to do, there are quite clear interfaces
- Slack for messaging
- Sprint planning and reviews with only project manager and person representing partner broker - ~1h each
- Every month review of the project with the board ~ 4h

Configuration

Just before starting project, project manager has to setup the following:

- **Gerrit**
- **Jira**
- **Slack**
- **Jenkins for CI**
- **Aws testing server, always with latest version of server. GUI developer will use a simple mockup done over the evening, so no staging evironment**

Product stages

- **List of products in stages**
- **Image summarizing them**

First stage – two months

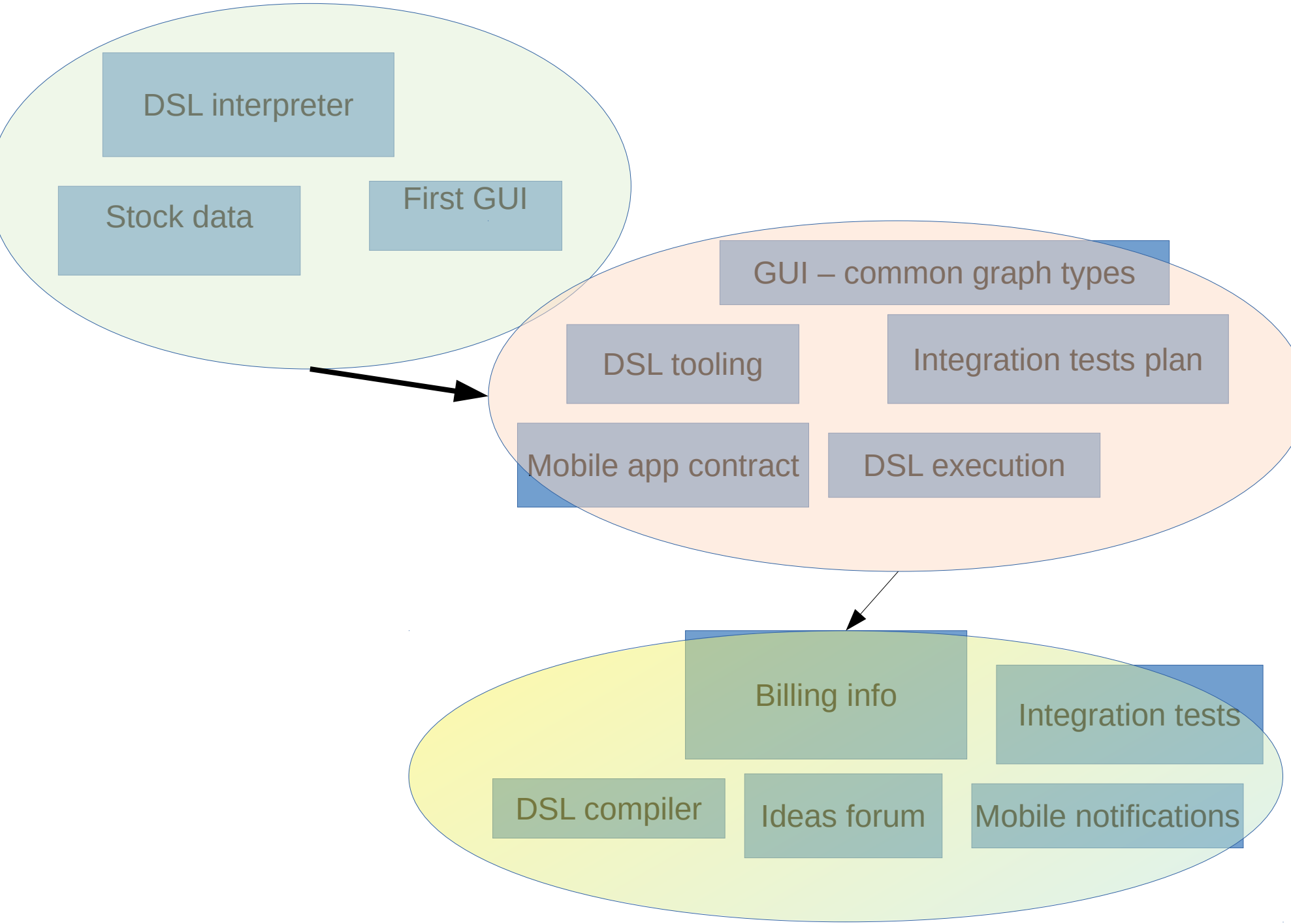
- **DSL specification and interpreter working only on correct input**
- **Pipeline for loading data from stock market into database**
- **First version of GUI with layout and displaying simplest charts (trend) based on mockup data**

Second stage – three months

- DSL corectness checker returning info about error location
- Contract with mobile app company – and they start working on it
- Web service allowing DSL execution and returning resulting streams of stock data from database
- Integration tests plan
- GUI – advanced graph transformations and types of charts

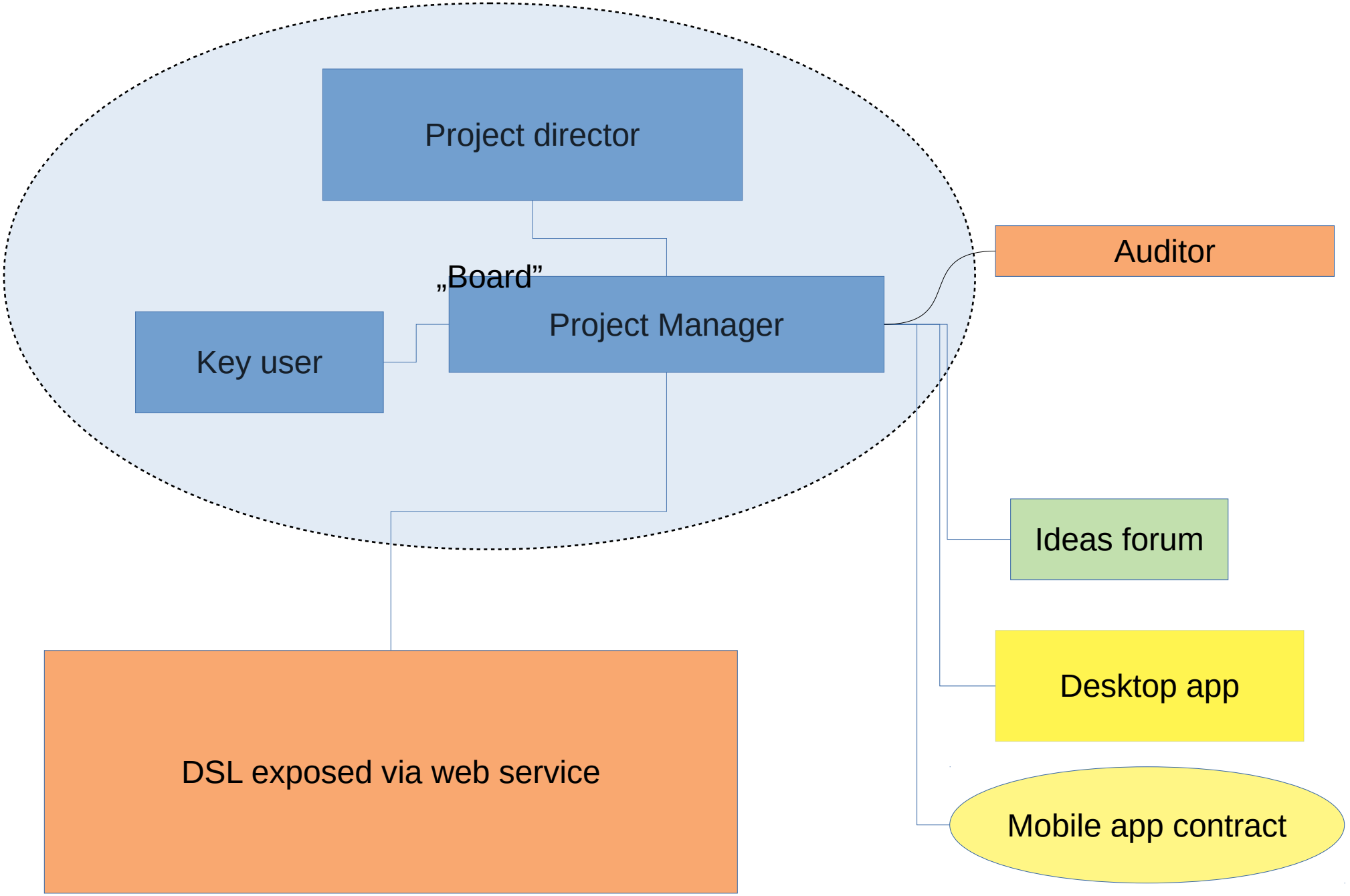
Third stage – two months

- DSL compiler
- Running DSL when specified events occur
- Web service for DSL measures resource consumption for billing
- Sending notifications from web service to mobile app
- Creating/executing integration tests
- Integrating GUI with full functionality of Web service
- Ideas forum


DSL interpreter
Stock data
First GUI
GUI – common graph types
DSL tooling
Integration tests plan
Mobile app contract
DSL execution
Billing info
Integration tests
DSL compiler
Ideas forum
Mobile notifications

Who is needed?

- **Key user/marketing guy – getting in touch with potential users at various meetings, shaping the requirements with parent broker**
- **Windows desktop developer**
- **Highly skilled programmer for DSL**
- **One programmer – past stock data, simple web service**
- **Project director – from parent broker**
- **Project manager also as a team leader**
- **Auditor at the very end for core DSL**


Project director
„Board”
Project Manager
Key user
Auditor
Ideas forum
Desktop app
Mobile app contract
DSL exposed via web service

Assigning responsibilities

Project manager responsible for mobile app contract, specs for ideas forum

Both core developer and manager responsible for DSL specificiation

Windows GUI developer for GUI

Core dev for DSL server

Programmer for loading past stock data and helping with DSL also responsible for creation of ideas forum after being provided specs