Algorithmic Trading System (ATS)

Introduction

Students need to know

- general concept of an electronic market
- interact with the market through transactions
- mainly addressing the buying and selling of financial securities
- notions of what a security and equity is

Introduction

terms and abbreviations will be used

- **Security:** refers to the financial product being traded. In this project, the security in question is a stock with an abbreviated name. For example, “CBA” will refer to Commonwealth Bank shares.

- **Bid order:** represents the intention of a buyer and will contain the security involved, the name of the buyer, the quantity of shares and the price.

Introduction

terms and abbreviations will be used

- **Ask order:** represents the intention of a seller and will contain the security involved, the name of the seller, the quantity of shares and the price.
- **Orderbook:** represents the list of unexecuted (i.e. pending) orders in the market.
- **Bid List:** represents the list of unexecuted (i.e. pending) bid orders in the market.

Introduction

terms and abbreviations will be used

- **Ask List:** represents the list of unexecuted (i.e. pending) ask orders in the market.
- **Spread:** refers to the difference between the highest bid price and the lowest ask price
- **Trade:** represents a match between a buyer and a seller (or that a bid order is executed against an ask order). A trade has the names of the buyer and the seller, the security involved, the quantity,

Introduction

terms and abbreviations will be used

The term “trading data” is used to describe three fundamental transactions types.

- Order processing
- Trades
- Market events

Example (Sirca file)

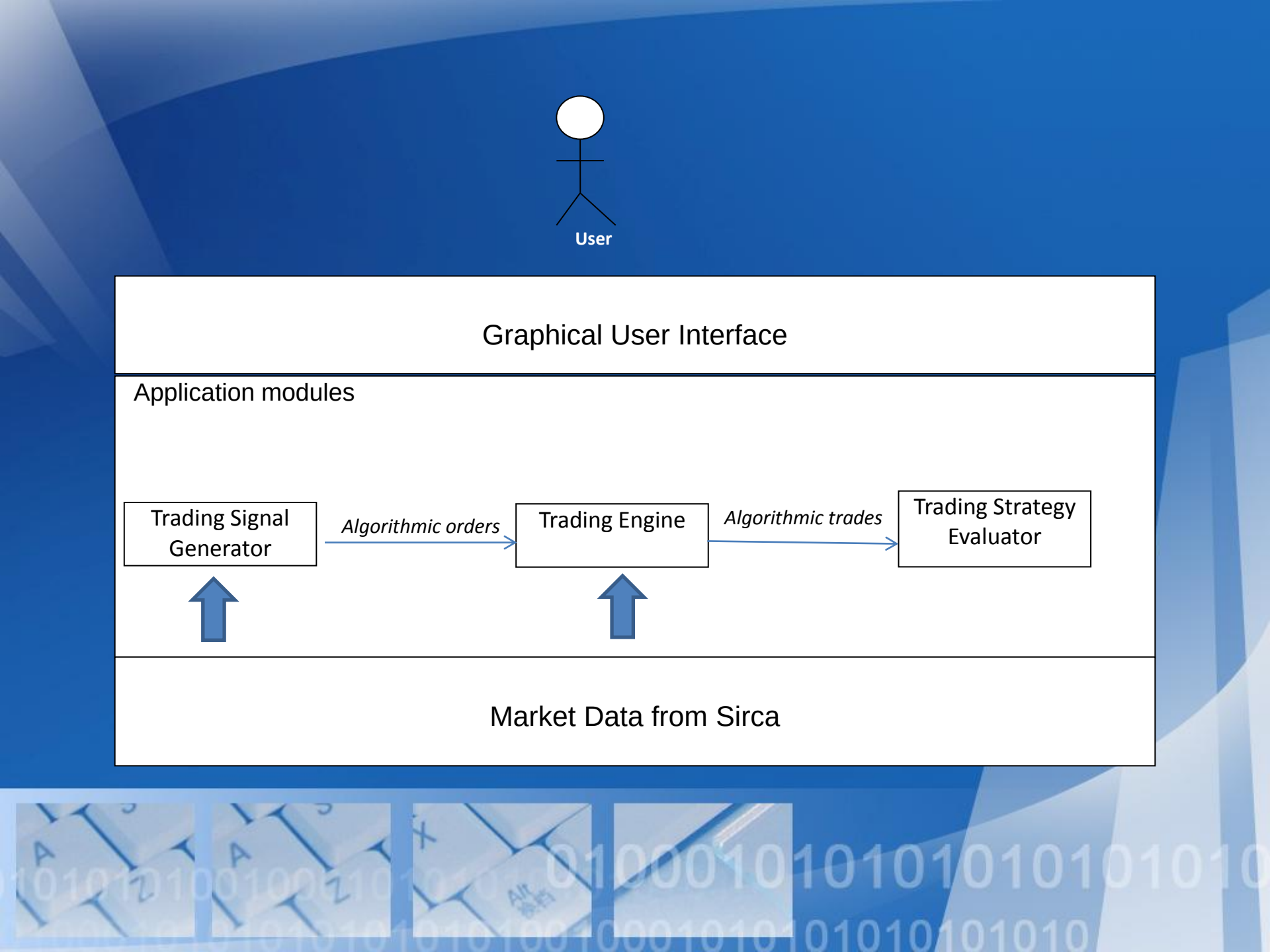
- See course web site for a sample file

Application Overview

Stage One

The ATAS preliminary architecture will consist of the following modules:

- a trading signal generator
- a trading engine
- a trading strategy evaluator


User
Graphical User Interface
Application modules
Trading Signal Generator
Algorithmic orders
Trading Engine
Algorithmic trades
Trading Strategy Evaluator
Market Data from Sirca

Requirements list

Id	Functional requirement	Comments
1	Reading a correctly formatted Sirca orders file (1 day only)	See "Introduction to orderbooks" document
2	Choosing an appropriate algorithmic trading strategy and setting its different parameters	List of algorithmic strategies will be provided as work progresses.
3	Generating algorithmic orders for 1 particular day	
4	Generating algorithmic trades for 1 particular day	List of techniques for generating algorithmic orders will be provided as work progresses.
5	Evaluating algorithmic trades and providing feedback to user	List of techniques for evaluating algorithmic trades will be provided as work progresses
6	Generating a strategy performance report	
7	GUI functions to control use cases (1-6) to load and execute an orders file	
8	GUI functions to visualise market data (spread, volume and depth)	

Requirements list

Id	Quality requirement	Comments
1	Speed of execution (transactions per second)	
2	Usability of the GUI	
3	Quality of the visualisation	
4	Quality of strategy performance report