BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2019
http://excel.fit.vutbr.cz

CrypTrader: Distributed system for algorithmic trading

Michal Hornický

Abstract

The success of cryptocurrencies like Bitcoin has created many new opportunities. In the wake of its succes, many online exchanges were created, that have unprecedented ease of use and acces to everyone, contrasting existing financial exchanges. Day-trading*on these exchanges is easy, and has a large potential because of the extreme volatility of these new markets. The system described in this paper aims to create a product, that would simplify day-trading on these exchanges for new users, and provide tools for automatization for more experienced users. These goals combined with the problem domain pose some interesting requirements for the implementation. This papers aims to outline what were these problems, and how did they influence the implementation.

Keywords: Keyword1 — Keyword2 — Keyword3

Supplementary Material: Demonstration Video — Downloadable Code

*xhorni14@fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

[Motivation]

Financial markets are complex systems, in which, market players interact with each other to determine prices of individual assets. Advances in financial technologies, like the advent of blockchain technology, and corresponding proliferation of cryptoccurencies , like Bitcoin[**?**] have changed nature of trading.

As a result of these advances, financial markets are now more approachable than ever, and thus present a significant opportunity. One example of services that successfully exploit this opportunity are cryptocurrency exchanges. They are a whole new kind of marketplace, that provides several advantages to its users. These exchanges usually provide approachable Web based user interface for everyone and, HTTP/WebSocket API for advanced users.

In order to capitalize on these advances, we must use advanced trading techniques. One of these is algorithmic trading. Basis of algorithmic trading, is utilization of some kind of algorithm, along with market data, in order to determine most profitable actions, that should be performed on the market.

[Problem definition]

This approach, has several requirements. One of them is large amount of computing power, since used algorithms might be extremely complex. Latency is also a big concern, since this space is extremely competitve, and a party, which is able to perform optimal actions sooner than all other parties, will net a larger profit. Thanks to these requirements, usage of this technique is not easy, or cheap.

However, advances in development and usage of distributed systems, might be an easy solution to these problems. Cloud computing[**?**] is now more widespread, and easy to use than ever. Thanks to new technologies

like docker[1] and kubernetes[**?**], the creation and management of distributed systems is easy, and systems created with these technologies can be easily secured, are scalable and provide other benefits for developers creating them compared to more monolithic architectures.

[Existing solutions] Discuss existing solutions, be fair in identifying their strengths and weaknesses. Cite important works from the field of your topic. Try to define well what is the *state of the art*. You can include a Section 2 titled "Background" or "Previous Works" and have the details there and make this paragraph short. Or, you can enlarge this paragraph to a whole page. In many scientific papers, *this* is the most valuable part if it is written properly. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent congue enim eu eros dictum sagittis. Aliquam ligula arcu, gravida at augue et, aliquet condimentum nulla. Morbi a lectus arcu. Nam ac commodo nisi, a accumsan nunc. Nam sed ante vel nulla elementum lobortis. Aliquam sed laoreet risus. Etiam ipsum odio, gravida eget sapien dictum, eleifend aliquet ex. Duis dapibus vitae enim vitae bibendum. Phasellus eget pulvinar massa. Mauris ornare urna. Maecenas porttitor libero ut turpis porttitor, auctor porta ligula rhoncus. Etiam a turpis blandit, eleifend dolor eget, egestas ligula. Nullam sollicitudin pulvinar mi sit amet interdum. Etiam in ultrices ante. Suspendisse potenti. Duis vel nisi eget tellus volutpat tempor. Suspendisse potenti. Duis vel nisi eget tellus volutpat tempor.

[Our solution] Make a quick outline of your approach – pitch your solution. The solution will be described in detail later, but give the reader a very quick overview now. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi laoreet risus a egestas imperdiet. Ut egestas nibh non fermentum vestibulum. Nullam quis eleifend ex, sed maximus nisl. Mauris maximus non dolor id tristique. Nunc pulvinar congue gravida. Nullam lobortis viverra leo sed commodo. Nulla in elit congue, ullamcorper metus non, eleifend risus. Vivamus porttitor, ex nec porttitor pretium, libero turpis ultrices dui, eu efficitur ante ipsum vel justo. Vivamus nec nulla nisi. Aenean quis mauris vitae metus gravida congue.

[Contributions] Sell your solution. Pinpoint your achievements. Be fair and objective. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer sit amet neque vel mi sodales interdum nec a mi. Aliquam eget turpis venenatis, tincidunt purus eget, euismod neque. Nulla et porta tortor, id lobortis turpis. Sed scelerisque sem eget ante interdum, vel volutpat arcu volutpat. Aliquam cursus, dolor a luctus.

[1] https://www.docker.com/

Figure 1. Good writing is bad writing that was rewritten several times. Don't worry, start somewhere.

2. How To Use This Template

Here will go several sections describing **your work**. From theoretical background (Section 2), through your own methodology (Section 3), experiments and implementation (Section 4 and possibly 5), to conclusions (Section 6). Instead of such technical content, here in this template we give a few hints how to write the paper.

Here is a list of actions to do first when you want to write an Excel@FIT paper:

1. Download all the template files (Sec. 2.1) into a directory. Maybe setup a GIT sync for backup, sharing, and for use from multiple computers.
2. Rename *2019-ExcelFIT-ShortName.tex* – replace ShortName with something that identifies your work and is short enough. For example: *VehicleBoxes*, *VanishingPoints*, *FastShadows*, *NewProbeTesting*, *CheapDynamicDNS*, ... This ensures that the filename already gives a hint what is in there (*mypaper.pdf* is really stupid).
3. Decide the language of your paper. English is recommended, as it is the language of science and technology. However, if you want to write in Czech or Slovak, you may. Use the correct option to the *\documentclass* command – the very first line of the template. The option may be either *[czech]* or *[slovak]*.
4. Insert meta information: **your name**, **e-mail**, **paper title**. Make sure the year in the top right corner of the document is correct. Do not hesitate to use ěščřžýáíé in your name – the LaTeX template is configured to eat UTF8 Unicode.

5. Insert teaser images ("image abstract"). Use as many *TeaserImage* commands as suitable – three or four will usually be fine for a one-line teaser. If you absolutely don't have any image showing your work (what kind of work could that be, anyway?!), remove the *Teaser* command.
6. Insert references to supplementary material. That will typically be clickable links to a youtube / vimeo video and to downloadable code, hyperlink to an online demo, or a github repo. If you have anything else relevant, put it in. If there is no supplementary material (really?!), remove or comment out the *Supplementary* command.
7. Keep calm and start writing (Figure 1). Some suggestions how to do this are in Section 3.
8. When your paper is accepted to Excel@FIT, uncomment *ExcelFinalCopy* at the beginning of this file. The line numbers will disappear from the sides of the text and your paper is ready for final publication.

Jean-Luc Lebrun [**?**] offers excellent recommendations for the canonical sections of scientific/technical papers. That is why Abstract, Introduction, and Conclusions in this template are already structured (remove the **[Bold labels]** in the Introduction and Conclusions, they are there just for your information and should not remain in the paper). This structure is no more than a recommendation, but divert from it only in cases when you exactly know what you are doing. The "phony" texts (typeset in gray color) roughly indicate the lengths of individual parts of these sections. Replace them with reasonable amounts of text.

2.1 What Files are Here and Why

The template package for Excel@FIT papers contains these files:

2019-ExcelFIT-ShortName.tex This is the template for the main LaTeX file – this is your paper. Do yourself a favor and replace *ShortName* in the filename with something meaningful.

2019-ExcelFIT-ShortName-bib.bib You can delete the contents of this file completely and start adding BibTeX references. It is much easier to use a small editing tool (Section 4, JabRef) than to format *.bib* file manually. Rename the file so that *ShortName* is consistent with the previous file (and update the filename in the *.tex* file).

ExcelAtFIT.cls LaTeX class file based on the *Stylish Article*[2] document class. Do not modify this file.

ExcelAtFIT-logo.pdf This is the logo on the title page.

VUT-FIT-logo.pdf Another logo on the title page.

images/placeholder.pdf Placeholder image; include it, scale it as needed, then replace it with real content.

images/keep-calm.png You don't need this file; it is only used in this template to show how to include a *.png* file (Figure 1).

3. How To Write the Paper — A Few Hints

A reasonable way to start writing is sketching the **abstract** [**?**]. Writing the abstract helps focus on what is important in the paper, what is the contribution, the meaning for the community. This exercise might take some 20 minutes and it pays back by clearing the key points of the text. In 99 % cases it is very reasonable to stick to the abstract structure [**?**] which is provided in this template.

Once you have the abstract, it should be very clear what is the message of the paper, what is the newly introduced knowledge, what are the proofs of its contribution, etc. This is the right time to start constructing the *skeleton* of the paper: it's **comics edition** [**?**]. This thing is composed of mainly four items:

1. **Sections and subsections.**
2. **Figures and tables.** At this phase, knowing that "once there will be a figure about this and that" is just fine. That is why we have the *placeholder.pdf* image – see Figure 2. If this totally generic image can be replaced by some temporary image which still needs more work, but which is closer to the target version, go ahead. A hand-drawing photographed by a cellphone is perfect at this stage.
3. **Todo's.** In the early comics version, every section is filled by one or more `\todo` commands and nothing else. A todo in the text might look like: **[[you should do something]]**. Unlike some elaborated todo packages, this simple solution (defined in the template) does not break the page formatting and it is perfectly sufficient.
4. **Phony placeholder texts.** These help you estimate the proportions of individual sections and subsections and to better aim at the correct paper length. Use `\blind{3}` to get three paragraphs of beautiful grey phony text.

[2] http://www.latextemplates.com/template/stylish-article

One hour is usually enough for creating a nice comics edition of the paper. No reason to wait, make a copy of the template and start butchering it.

Having the comics edition usually lubricates the whole writing process. Now, the paper contains 20 or so todo's – why not take the easiest one of them and replace it with a few lines of text within 15 minutes or even less. Writing is no more a scary complex work.

3.1 Images and Tables

Visuals (figures, tables, good equations, section headings) make the skeleton of a properly written paper. A time-stressed reader should be able to get the idea from only browsing them. Therefore:

1. **Make them perfect.** Cheap and ugly images – cheap and ugly paper. Imperfect or shorter text – who cares?
2. **Make them self-contained.** Be not afraid to have a ten-lines-long caption under an image. The image plus its caption must make perfect sense by themselves, without reading the text.
3. **Make them many.** EVERY technical idea is better explained by an image. Two images per page are a moderate start.

LaTeX lets you easily insert both vector and raster graphics. It is reasonable to use three formats:

.pdf Perfect for vector graphics. All graphs **must** be in vector and therefore in .pdf. Gnuplot, pyplot, Matlab – they all produce vector graphs in .pdf easily. Diagrams, system structures, sketches – all vector graphics. It's 2019, not 1980 anymore...

.jpg Suitable for photos. **Never** for plots or screenshots.

.png Good for precise raster graphics. Screenshots, raster plots, raster outputs of programs. Not for diagrams and plots – unless it is a one-in-ten-years exception.

Caption of a table goes **before** the table (e.g. Table 1), just the opposite way than with figures. There is no logic behind, that's just how it is.

3.2 Sections and Subsections

It is usually wrong to have subsections in the Introduction; it is always wrong to have them in Conclusions. In this kind of paper, it is very likely to be wrong to have any subsubsections.

Section headings are the skeleton of the paper make them accurate and descriptive. One-word section titles (apart from Introduction and Conclusions) are typically wrong, because they are not descriptive. "Proposed Method for Running X by Using Y" is better than "The Method". "Implemented Application for PQR Communication" is better than "Application". The outline of all section titles should contain all the keywords relevant for the work. Just by seeing them, the reader should be able to tell precisely the topic of the paper. If not, the section headers are wrong (usually too short and generic).

3.3 Keywords

Keywords are specified at the top of the document.

1. When making the list of keywords, ask yourself this: "What should one write to google, so that the right answer would be my paper?"
2. Very generic terms ("IT", "Graphics", "Hardware") are useless. Narrow terms are fine ("Matrix Code Recognition", "Appearance-Based Vehicle Segmentation", ...)

4. Some Useful Tools

This list is not a list and it is by no means complete. If you prefer other tools – cool, stick with them. If you are just beginning, consider these.

Overleaf Online LaTeX editing – if you don't want to install and learn many tools, Overleaf is a great solution: works online and allows sharing your text with your supervisor. Unless there are very good reasons for not doing so, stick to Overleaf.

MikTeX Problem-free LaTeX for Windows; a distribution with perfect automation of package download. Single setup, no more worries.

TeXstudio Portable and opensource GUI for LaTeX writing. Ctrl+click jumps from pdf to latex and back. Integrated spellchecker, syntax highlighting, multifile projects, etc. First, install MikTeX, then TeXstudio. Ten minutes and you are a LaTeX master.

JabRef Nice and simple Java program for managing *.bib* files with references. Not much to learn – one window, a straightforward form for editing the entries.

InkScape Opensource and portable editor of vector files (SVG and – conveniently – PDF). The proper tool for making great drawings for papers – not the easiest to learn, though.

GIT Great for team collaboration on LaTeX projects, but also helpful to a single author – for versioning, backup, multi-computer, ...

5. Frequently Used LaTeX Fragments

Here goes an example of a table:

Table 1. Table of Grades

Name		
First name	Last Name	Grade
John	Doe	7.5
Richard	Miles	2

Figure 2 shows a wide figure, Figure 1 is a single-column figure with width specified relatively to the column. Some mathematics $\cos\pi = -1$ and α in the text[3].

Now, this is an equation:

$$\cos^3\theta = \frac{1}{4}\cos\theta + \frac{3}{4}\cos 3\theta \tag{1}$$

and here is a bunch of equations aligned horizontally:

$$3x = 6y + 12 \tag{2}$$

$$x = 2y + 4 \tag{3}$$

In programming, longer and more descriptive identifiers are better:

```
volume = width * height * length
if volume > volume_max:
  print "That's too much material!"
```

but the same is **wrong** in mathematical writing and in papers and single-letter identifiers are to be used:

$$V = w \times h \times l, \tag{4}$$

$$\delta(V) = V > \tau_V \tag{5}$$

identifiers composed of more than one letters are meaningful only in rare cases such as $V_{\max}$ or t_{start}. Oftentimes it makes sense to define one's own reasonable notation by using accents:

$$\bar{x} = \frac{\sum\limits_{x_i \in X} x_i}{|X|}. \tag{6}$$

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

[3] And some mathematics $\cos\pi = -1$ and α in a footnote.

6. Conclusions

[Paper Summary] What was the paper about, then? What the reader needs to remember about it? Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin vitae aliquet metus. Sed pharetra vehicula sem ut varius. Aliquam molestie nulla et mauris suscipit, ut commodo nunc mollis.

[Highlights of Results] Exact numbers. Remind the reader that the paper matters. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed tempus fermentum ipsum at venenatis. Curabitur ultricies, mauris eu ullamcorper mattis, ligula purus dapibus mi, vel dapibus odio nulla et ex. Sed viverra cursus mattis. Suspendisse ornare semper condimentum. Interdum et malesuada fames ac ante ipsum.

[Paper Contributions] What is the original contribution of this work? Two or three thoughts that one should definitely take home. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent posuere mattis ante at imperdiet. Cras id tincidunt purus. Aliquam erat volutpat. Morbi non gravida nisi, non iaculis tortor. Quisque at fringilla neque.

[Future Work] How can other researchers / developers make use of the results of this work? Do you have further plans with this work? Or anybody else? Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse sollicitudin posuere massa, non convallis purus ultricies sit amet. Duis at nisl tincidunt, maximus risus a, aliquet massa. Vestibulum libero odio, condimentum ut ex non, eleifend.

Acknowledgements

I would like to thank my supervisor X. Y. for his help.

Figure 2. Wide Picture. The whole figure can be composed of several smaller images. If you want to address individual images in the caption or from the text, use the *subcaption* package.