



NATIONAL RESEARCH
UNIVERSITY

National Research University Higher School of Economics
Faculty of Computer Science
Master's Programme 'System and Software Engineering'

REFERENCE AND DATA SEMANTIC-BASED SIMULATOR OF PETRI NETS EXTENSION WITH THE USE OF RENEW TOOL

Student: Anton Rigin

(group mSSE19)

Supervisor: Sergey Shershakov
Senior Lecturer at School of Software Engineering,
Research Fellow at Laboratory of Process-Aware Information
Systems (PAIS Lab)

Moscow, 2020

DESCRIPTION OF SUBJECT AREA

- Concurrent software systems modelling and validation
- Process mining
 - The techniques of analysis of business processes based on event logs
- Petri nets and their modifications

BASIC TERMS, CONCEPTS, DEFINITIONS (1/8)

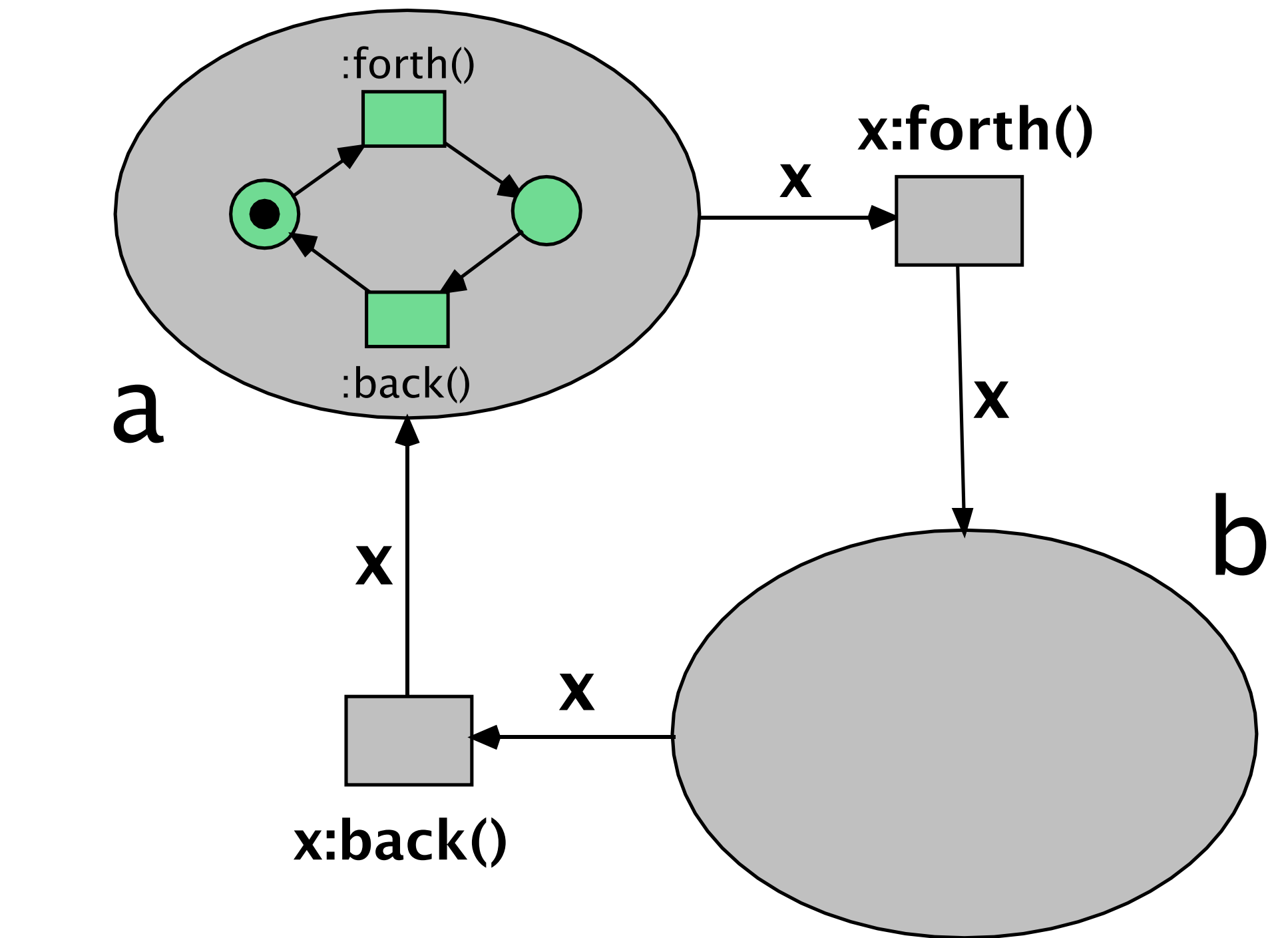
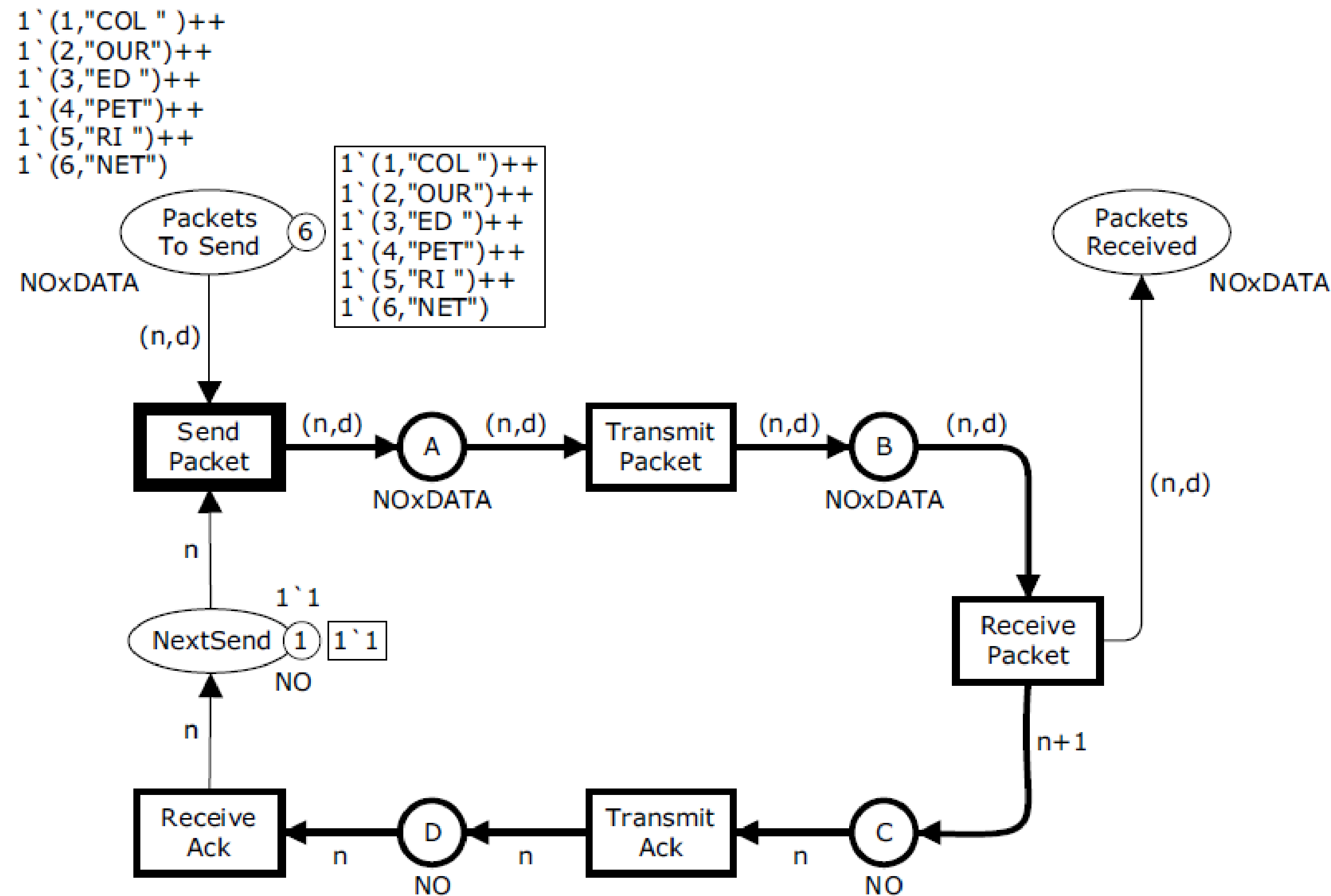
- **Petri nets**

- The powerful tool for modelling the processes and behaviour of concurrent software systems
- They can be represented as the **directed bipartite graph (bigraph)** where the vertices (nodes) are **places** and **transitions** and the edges are **arcs** connecting the places and transitions
- Petri nets support modelling behavior due to **tokens** which form **markings**
- Formally, the Petri net N is the triple $N = (P, T, F)$ [1], where:
 - P is the finite set of places of the net N
 - T is the finite set of transitions of the net N
 - $P \cap T = \emptyset$
 - $F \subseteq (P \times T) \cup (T \times P)$ is the set of arcs of the net N
- A marking M is a distribution of tokens across places: $M: P \rightarrow \mathbb{N}$

BASIC TERMS, CONCEPTS, DEFINITIONS (3/8)

- **Coloured Petri nets**
 - The Petri nets in which data types of tokens and arc expressions are defined and used [2] [3]
 - These data types are called **colours** [2] [3]
- **Reference Petri nets**
 - The Petri nets which have reference semantics rather than value semantics [4]
 - The tokens in such nets are references to objects (for example to other Petri nets – this concept called **hierarchical Petri nets, nets within nets** or **nested nets**) [4] [5]

BASIC TERMS, CONCEPTS, DEFINITIONS (4/8)



The example of hierarchical reference Petri net [6]

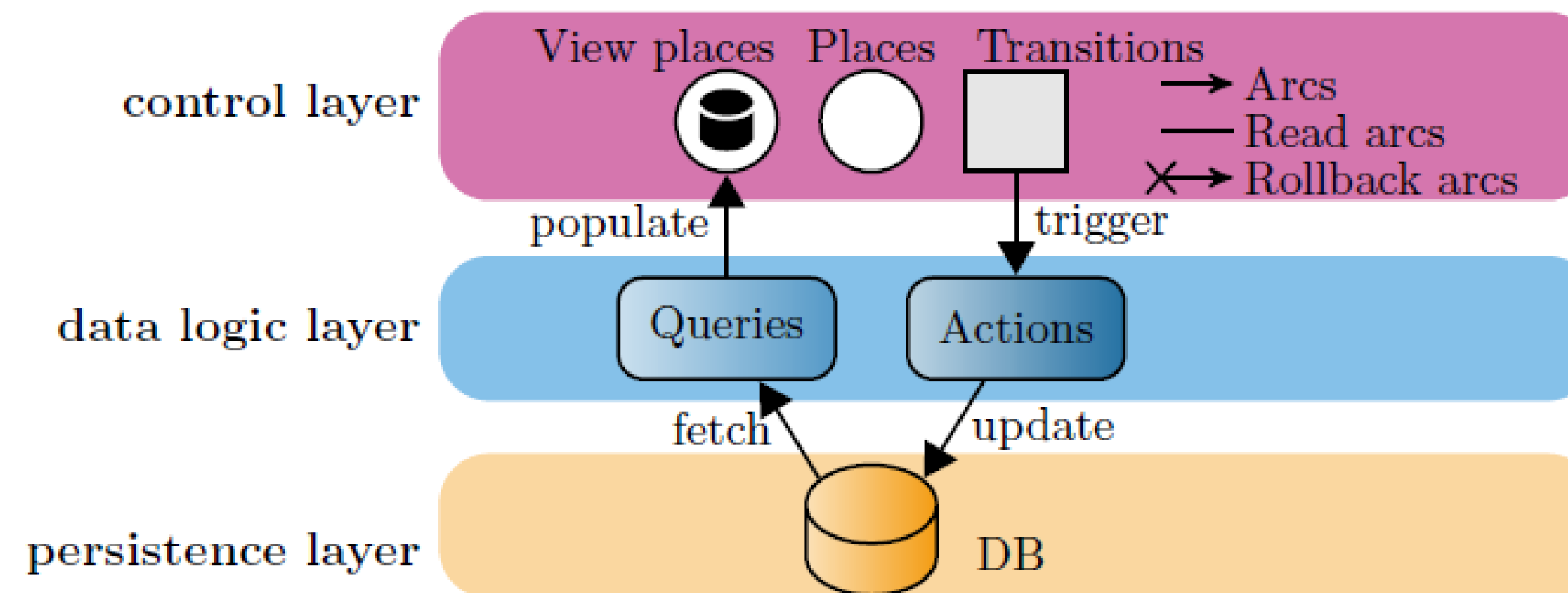
The example of coloured Petri net for modelling the simple transport protocol [3]

BASIC TERMS, CONCEPTS, DEFINITIONS (5/8)

- **DB-nets**

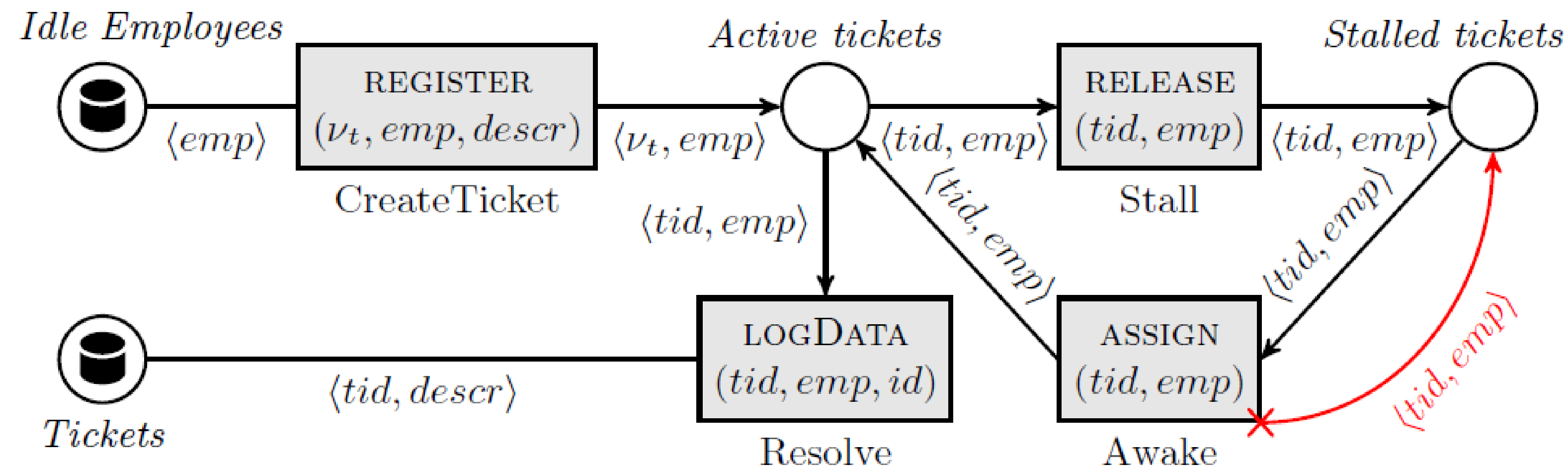
- Were introduced by Marco Montali and Andrey Rivkin in 2016 [7]
- DB-net is the model consisting of three layers [7]:
 - ✓ **Control layer** is the modified coloured Petri net
 - ✓ **Data logic layer** interconnects the control layer and persistence layer and allows to access and manipulate the persistent data using **queries** and **actions**
 - ✓ **Persistence layer** represents the relational database with **constraints**
- DB-nets allow to consider the process and the persistent data used by this process simultaneously [7]

BASIC TERMS, CONCEPTS, DEFINITIONS (6/8)



The DB-net structure [8]

BASIC TERMS, CONCEPTS, DEFINITIONS (7/8)



The example of the DB-net's control layer [7]

RELEVANCE AND MOTIVATION

- Petri nets are used as a popular technique for modelling the behavior of concurrent programs
- The Petri nets and their modifications implemented in many different software tools (**CPN Tools** for the coloured Petri nets [9], **Renew** for the reference Petri nets [5] and so on)
 - However, the standard Petri nets as well as coloured Petri nets and reference Petri nets cannot consider the state of persistent data
- The DB-nets solve this problem [7]
 - However, there is no published program implementations of the DB-nets
- **Renew** is a powerful software tool which implements reference Petri nets [5]
- Therefore, it could be useful to implement the DB-nets with the reference semantics support as the plugin for the **Renew**
- This implementation will allow to model the behavior of the programs with reference semantics (which are very common in such programming languages as Java) with considering the state of persistent data

OBJECTIVE AND TASKS

- **Objective**

- Implementing the software tool for simulating DB-nets with reference semantics support as the plugin for the Renew

- **Tasks**

- Learning the existing Renew's design and code
- Formulating the formal functional and non-functional requirements for the software tool (the DB-nets Renew plugin)
- Developing the design of the software tool
- Implementing the DB-nets concept in the form of Renew plugin
- Testing the developed software tool using some real enterprise data or artificially generated data
- Documenting the developed software tool

INITIAL FUNCTIONAL REQUIREMENTS (1/2)

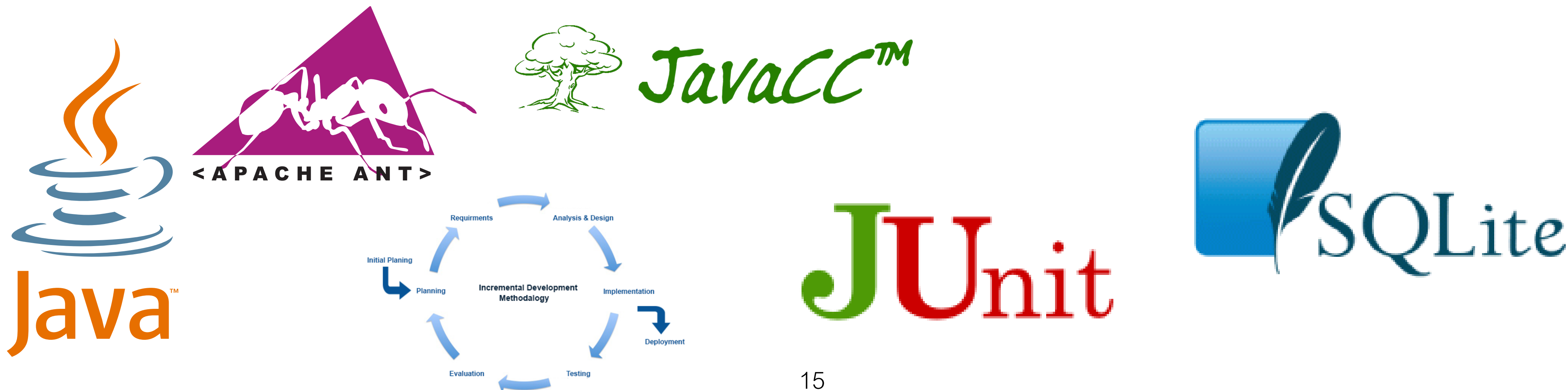
- The program shall allow user to model the db-net control layer using the Renew tool's graphical user interface (GUI) elements for net's interactive drawing
- The program shall allow user to create the database schema at the db-net persistence layer using the Renew tool's GUI based dialog windows
- The program shall allow user to declare queries and actions for the db-net data logic layer using the Renew tool's GUI based dialog windows
- The program shall allow user to simulate the modeled db-net's run using the Renew tool's GUI

INITIAL FUNCTIONAL REQUIREMENTS (2/2)

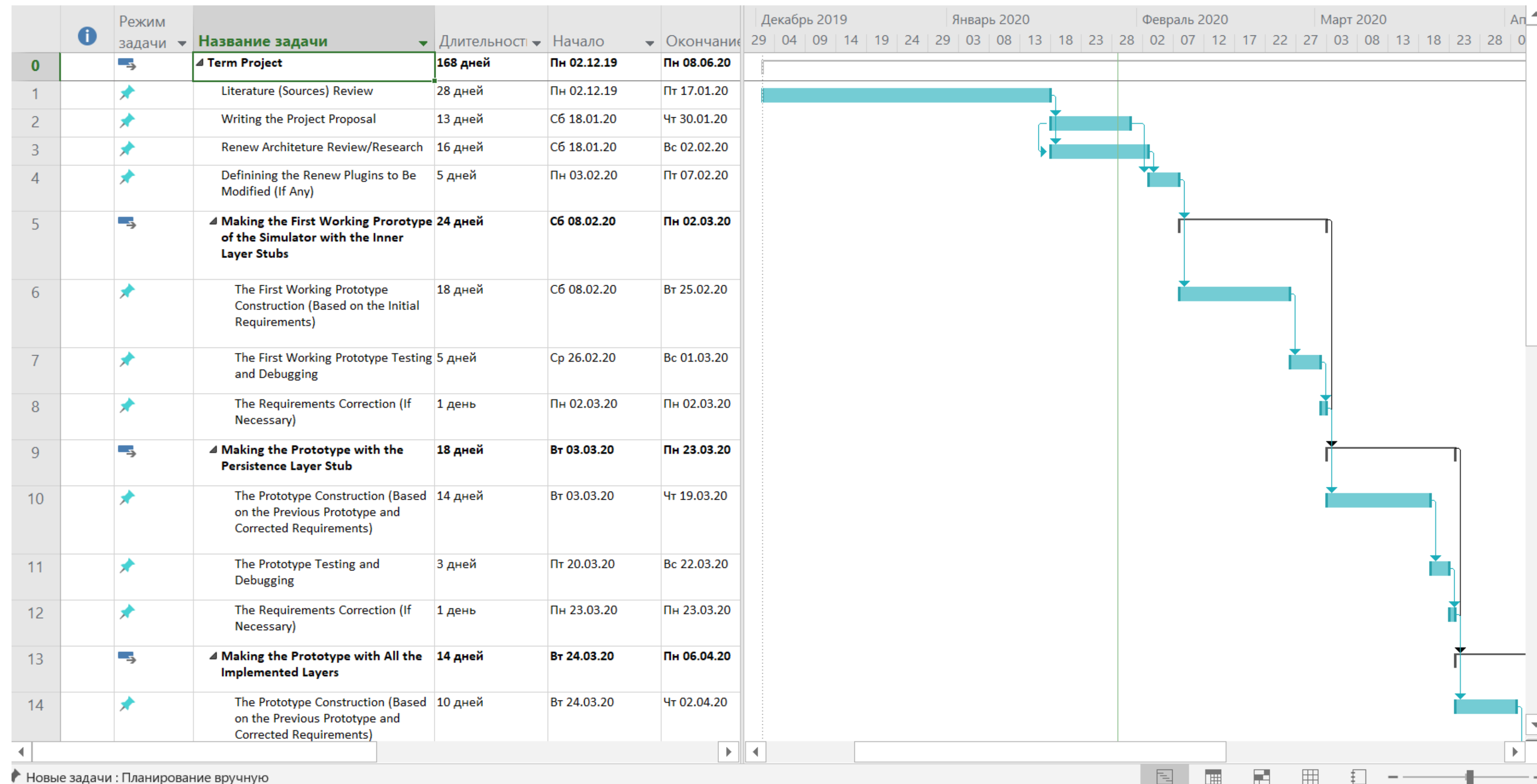
- The program shall allow user to input the external data for the db-net's run during the db-net's run simulation where it is necessary according to the db-net model using the Renew tool's GUI
- The program shall allow user to save the modeled db-net using the Renew tool's GUI
- The program shall allow user to open the previously saved modeled db-net using the Renew tool's GUI

METHODS, TOOLS AND TECHNOLOGIES TO BE USED

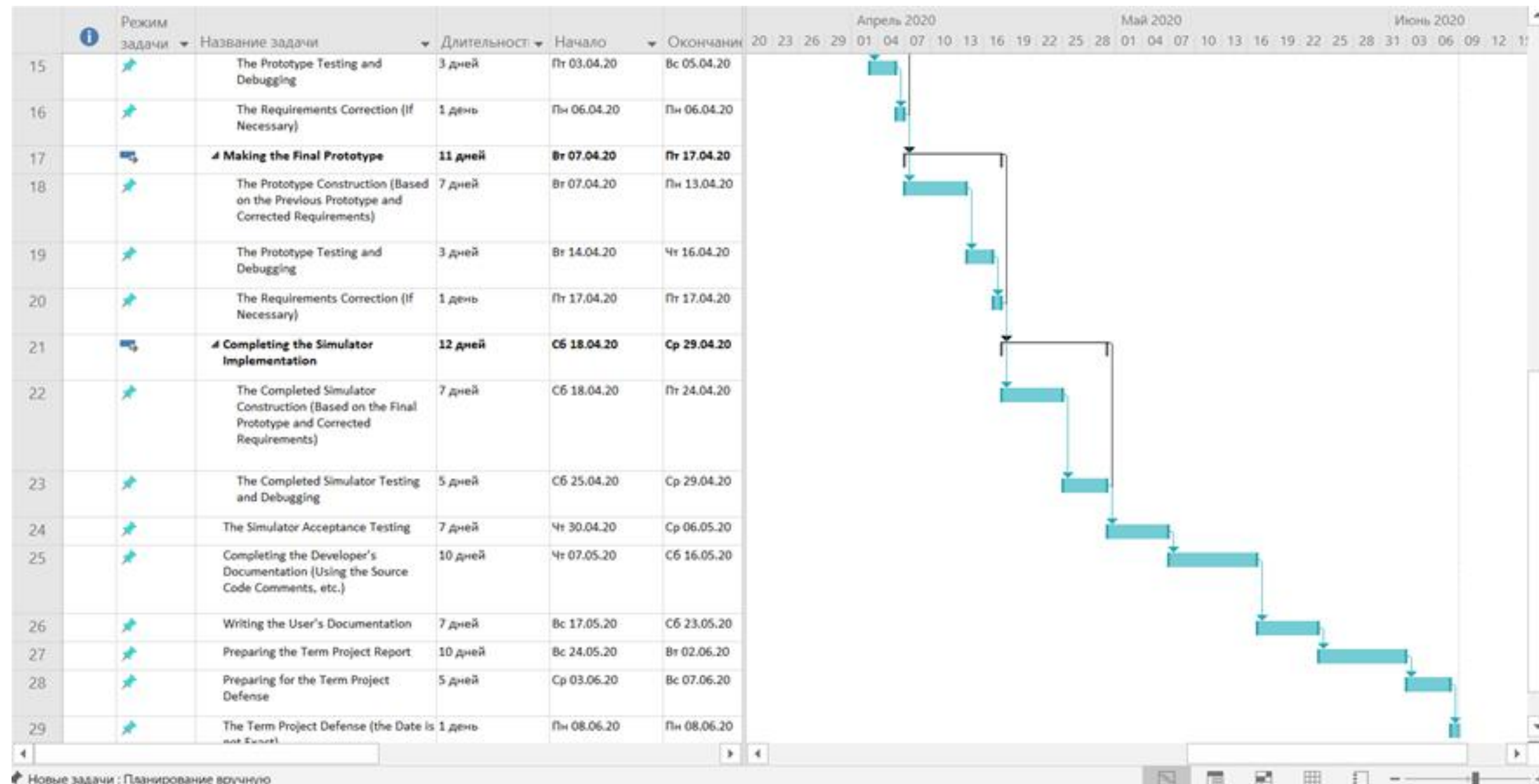
- The DB-net with the reference semantics support is a core model used in the project
- The methods of constructing DB-net and simulating its run are used
- Renew is a main software tool used in the project
- Technological stack: Java SE 8, Apache Ant, JavaCC, JUnit 4 and SQLite (the latter is planned to be used as the relational DBMS for the persistent layer of the DB-net)
- The elements of the incremental SLC model and prototyping approach are used in the project [10]



WORK PLAN – THE PROJECT'S GANTT CHART (1/2)



WORK PLAN – THE PROJECT'S GANTT CHART (2/2)



CURRENT RESULTS

- The existing Renew code and design is learnt
- The functional requirements are formulated
- The simple Renew plugin which writes its setup to log is created

EXPECTED RESULTS

- The developed, tested and documented software tool for simulating DB-nets with reference semantics support as the plugin for the Renew is expected
- The developed software tool is expected to be applicable for cases where it is necessary to consider both process and the persistent data
- The term project can be continued in the 2nd year as the master thesis which will possibly include development of some new features for the developed software tool and conducting a research with using a real enterprise data of some of the PAIS Lab's business partners (for example, the Exactpro Systems and London Stock Exchange)

REFERENCES (1/3)

- [1] W. Reisig. *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*. Berlin, Heidelberg, Germany: Springer, 2013.
- [2] J. Kurt. "A Brief Introduction to Coloured Petri Nets," in *Third Int. Workshop on TACAS*, Enschede, the Netherlands, Apr. 2 – 4, 1997, pp. 203 – 208.
- [3] J. Kurt and M. K. Lars. *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*. Berlin, Heidelberg, Germany: Springer, 2009.
- [4] B. Farwer. "LLPN – Linear Logic Petri Nets: What are Object Petri Nets?" Wayback Machine.
<https://web.archive.org/web/20051103131745/http://www.llpn.com/OPNs.html>
(accessed Dec. 1, 2019).

REFERENCES (2/3)

[5] “Renew – The Reference Net Workshop.” Renew.de. <http://www.renew.de/> (accessed Dec. 1, 2019)

[6] “File:Forthandback.svg.” Wikimedia Commons. <https://commons.wikimedia.org/wiki/File:Forthandback.svg> (accessed Dec. 5, 2019).

[7] M. Montali and A. Rivkin. “DB-Nets: On the Marriage of Colored Petri Nets and Relational Databases,” in *Transactions on Petri Nets and Other Models of Concurrency XII*, M. Koutny, J. Kleijn, W. Penczek, Eds., Berlin, Heidelberg, Germany: Springer, 2017, pp. 91 – 118.

REFERENCES (3/3)

[8] M. Montali and A. Rivkin. “DB-Nets: On the Marriage of Colored Petri Nets and Relational Databases.” ResearchGate.

https://www.researchgate.net/publication/310122815_DB-Nets_on_The_Marriage_of_Colored_Petri_Nets_and_Relational_Databases

(accessed Jan. 25, 2020).

[9] “CPN Tools – A tool for editing, simulating, and analyzing Colored Petri nets.” CPNTools.org. <http://cpntools.org/> (accessed Dec. 1, 2019).

[10] P. Bourque and R.E. Fairley, Eds., “Software Life Cycle Models,” in *Guide to the Software Engineering Body of Knowledge, Version 3.0*, USA: IEEE Computer Society, 2014, ch. 8, sec. 2.2, pp. 8-5–8-6.



NATIONAL RESEARCH
UNIVERSITY

Thank you for your attention!

amrigin@edu.hse.ru

anton19979@yandex.ru

HSE Address: 20 Myasnitskaya Ulitsa, Moscow 101000 Russia | 11 Pokrovsky Boulevard, Moscow 109028 Russia