Self-presentation

of achievements in scientific research

Andrzej Daniluk, Ph.D.

Department of Applied Computer Science, Institute of Computer Science,
Maria Curie-Skłodowska University

Lublin 2016

Scientific CV

Andrzej Daniluk

Education

Ph.D. of physical sciences in the field of physics with specialization in computational physics

2000
Maria Curie-Skłodowska University,
Faculty of Mathematics and Physics, Institute of Physics

Ph.D. dissertation *Growth models of epitaxy analysed with the help of RHEED theory (in Polish)*
Supervisor: Prof. dr habil. Paweł Mikołajczak, MCS University
Reviewers: Dr habil. Ryszard Taranko, MCS University
Prof. dr habil. Marian A. Herman, Institute of Physics, Polish Academy of Science, Warsaw

M.Sc. 1991
Maria Curie-Skłodowska University,
Faculty of Mathematics, Physics and Chemistry, Institute of Physics

Master thesis *Computer modeling of growth of thin epitaxial films (in Polish),*
Supervisor: Prof. dr habil. Mieczysław Jałochowski

Employment Department of Applied Computer Science, Institute of Computer Science,
MCS University, ul. Akademicka 9, 20-033 Lublin,
e-mail: andrzej.daniluk@poczta.umcs.lublin.pl

Professional career

2013 - : senior lecturer, Department of Applied Computer Science, Institute of Computer Science, MCS University
2005-2013: assistant professor, Department of Applied Computer Science, Institute of Computer Science, MCS University
2004-2005: assistant professor, Department of Applied Computer Science, Institute of Physics, MCS University
2002-2004: lecturer, the State School of Higher Education in Chełm.
2001-2004: assistant professor, Department of Experimental Physics, Institute of Physics, MCS University
1995-2000: assistant, Department of Experimental Physics, Institute of Physics, MCS University
1993-1995: assistant, Department of Biophysics, Institute of Physics, MCS University
1991-1993: physics and computer science teacher, High School No. 6 in Lublin

I. General characteristics of scientific interest

My first scientific works were related to the present-day research connected with using computer modeling methods in selected areas of material engineering. During my studies in the Physics Institute at Maria Curie-Skłodowska University, while working on my thesis, I prepared and implemented an original algorithm for the qualitative numerical modeling of empirical data on growth of ultrathin homoepitaxial layers crystallised using the method based on epitaxy from molecular beams. In 1993 I was employed as an assistant in the Institute of Physics MCS University. During that time I took part in research projects in the fields of physical chemistry and biophysics, and I worked out the spectroscope software for the Raman spectrum research. I am also the author of a program operating the apparatus used for obtaining the Langmuir-Blodgett monolayers – the ultrathin adsorptive monolayers on the surface of a liquid (the software has been used up to the present day in the Department of Biophysics in the Physics Institute MCS University). My scientific work resulted in the co-authorship of two publications.

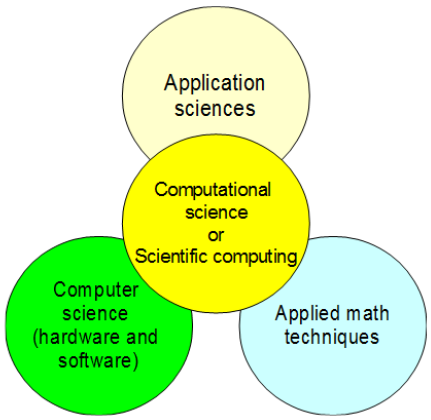
At that time I also began my cooperation with the group of psychologists in Lublin. I have prepared and implemented a computer program for the statistical compilation of the results of research carried out by the staff of the Psychology and Pedagogy Faculty MCS University.

Since 1996 I have carried on the scientific work on the material engineering of ultrathin epitaxial layers.

After the completion of my Ph.D. I continued my work on using different mathematical models and corresponding numerical algorithms allowing modeling of types of complex interfaces of crystalline heterostructures and temperature relationships of RHEED.

In 2005 I was employed in the Department of Applied Computer Science in the newly-opened Institute of Computer Science in the Faculty of Mathematics, Physics and Computer Science, MCS University. During this period I focused mainly on designing and implementing efficient algorithms and computer programs that would allow real-time modeling of high-energy electron diffraction RHEED.

Computational Science is a rapidly-emerging transdisciplinary field at the intersection of the natural sciences, computer science, and mathematics because now much scientific research involves computing as well as theory and experiment.



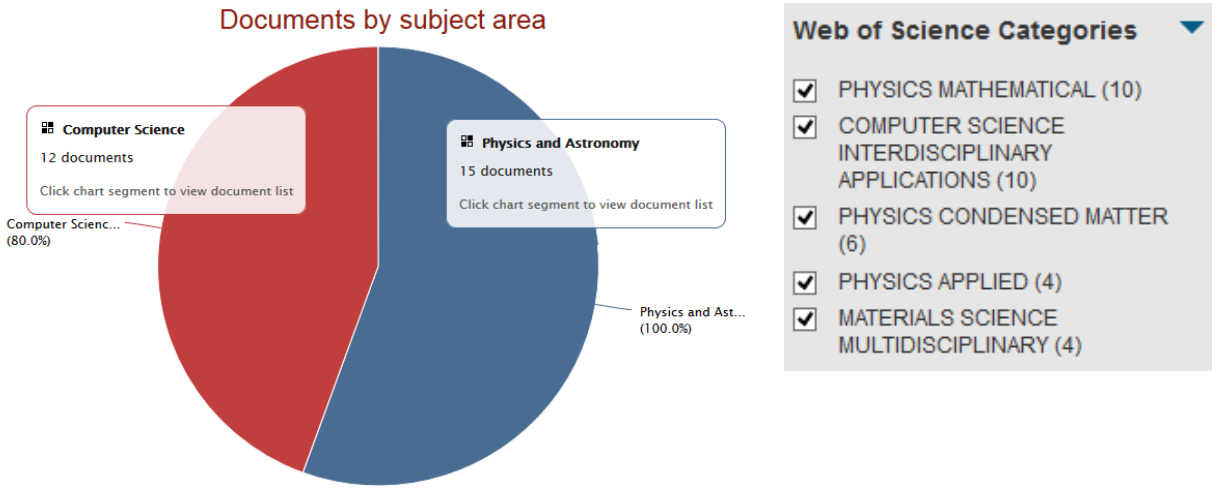
Overall, the area of my scientific interests can be characterized by the following formula:

Computational Science = problem domain (physics) +
computer science + applied math techniques

It is not possible to evaluate my scientific achievements in a fragmentary manner.

After obtaining the doctoral title the majority of papers (80%) have been published in Computer Physics Communications (CPC) (*An International Journal and The Library for Computational Physics and Physical Chemistry*). CPC is a scientific journal of high international reputation that represents a repository of computer programs used in physics and physical chemistry. It is rather focused on algorithms and methods than the physical results. Therefore, as described in the JCR database *ISI Web of Knowledge*, the papers published in it are publications in both physics and computer science (according to SCI, CPC is 100% Computer Science and 100% Physics). At present, the so-called 5-year Impact Factor of the journal is 3.508. The subject matter and research tools presented in papers [1-10] are in the interdisciplinary area of interest of modern science and technology, typical of computer and technical physics, as well as applied computer science.
All projects developed by me [1-3, 7-10] have been published in:
International CPC Program Library, School of Electronics, Electrical Engineering and Computer Science, Queen's University of Belfast *http://cpc.cs.qub.ac.uk/*, catalogues: ADUY, ADVL and AETW.

The vast majority of my interdisciplinary publications has been published in journals identified within the Journal Citation Reports (JCR). Below, I present the classification of my papers according to data from the Scopus and Web of Science databases, respectively:



Software developers struggle to get academic recognition for their work as authors. Moreover, software itself is currently not systematically treated as a full and equal academic citizen. According to many scientific authorities related to interdisciplinary research, software should be treated as an integral part of the scientific communication ecosystem in the 21st century, [F. Seinstra, D. Wallom, K. Keahey, SoftwareX 1–2 (2015), Editorial]: *"Yet, despite the impact scientific software instruments have on the daily practice of many scientists, and despite the great many new discoveries that can be attributed directly to the existence of these instruments, the design and implementation of software instruments is often not held in high regard in the traditional academic environment. In science, publications and their citations are by far the most essential prerequisites for pursuing a successful academic career. Since traditionally the design and development of software does not warrant a publication it is often denied the acknowledgement that citations provide. In this way, the current academic setting is hindering, even blocking, the career paths of the scientific software designers and implementers that a growing portion of science has become so much dependent on."*

The main scientific objective presented in the papers listed below was to build an abstract model for RHEED calculations and examine the possibility of applying this model to the quantitative and qualitative analysis of experimental data. Taken scientific issues associated with the phenomenon of diffraction of high-energy electrons RHEED used to monitor the state of the surface of epitaxial layer is an important domain of technical physics having a difficult to overestimate the impact of technological developments in many fields, including computer science. Application nature of the subject matter and originality of construction works (computer programs) performed in the context of modern methods and tools of computer physics and applied computer science is the main value submitted for evaluation of scientific achievements.

As a scientific achievement for the habilitation procedure according to the "*art. 16 pkt 1 ust. 2.1 i 2.2 ustawy z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. nr 65, poz. 595 ze zm.)*" I present thematically linked series of seven research articles and three implementation papers published in the journals from the Journal Citation Reports (JCR) database, with the common title:

Modeling of virtual scientific experiment

No	Bibliographic data	Comments	5-year Impact Factor
1	A. Daniluk, *Dynamical calculations for RHEED intensity oscillations*, Computer Physics Communications 166 (2005) 123-140.	The work contains a listing of the program suggested by the reviewers and editor.	3.508
2	A. Daniluk, *An extension of the computer program for dynamical calculations of RHEED intensity oscillations. Heterostructures*, Computer Physics Communications 176 (2007) 70-73.		3.508
3	A. Daniluk, *RHEED intensities from two-dimensional heteroepitaxial nanoscale systems*, Computer Physics Communications 185 (2014) 3001-3009.		3. 508
4	P. Mazurek, A. Daniluk, K. Paprocki, *Substrate temperature control from RHEED intensity measurements*, Vacuum 72 (2004) 363-367.		1.647
5	P. Mazurek, A. Daniluk, K. Paprocki, *Analysis of RHEED intensities during formations of the CaF_2/Si(111) and MgO/YSi_{2-x}/Si(100) interface*, Vacuum 57 (2000) 229-236.		1.647
6	P. Mazurek, A. Daniluk, K. Paprocki, *Forming the high quality $CoSi_2$ by solid phase epitaxy*, Optica Applicata 32 (2002) 389-395.		0.496
7	A. Daniluk, *Visual modeling for scientific software architecture design. A practical approach*, Computer Physics Communications 183 (2012) 213-230.	The size of work is larger. According to editor decision a part of diagrams is available in electronic form as the additional materials.	3. 508
According to "*art. 16 pkt 1 ust. 2.2 ustawy (Dz. U. nr 65, poz. 595 ze zm.)*" I declare the authorship of the original achievements in the form of construction of *GrowthXX* computing environments. The *Growth09* and *GrowthCP* computing environments are implemented based on three implementation works.			
8	A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part II*, Computer Physics Communications 181 (2010) 709-710.	The size of work as mentioned above. Represents the model implementation described in [7]. See Fig. 10 level M0. The class, sequence and use case diagrams are available in electronic form as the additional materials.	3. 508
9	A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part I*, Computer Physics Communications 181 (2010) 707-708.	The size of work as mentioned above. Represents a program used in [8]. Rules publishing *CPC Program Library* prevent publication of executable artifacts in one edition. The class, sequence and use case diagrams are available in electronic form as the additional materials	3. 508
10	A. Daniluk, *Cooperative and competitive concurrency in scientific computing. A full open-source upgrade of the program for dynamical calculations of RHEED intensity oscillation*s, Computer Physics Communications 182 (2011) 1389-1390.	The work includes a 36-page white paper available in electronic form (with the distribution files) in the *CPC Program Library* database.	3. 508

II. Problem domain

The problem domain is the Reflection High-Energy Electron Diffraction (RHEED) in dynamic terms. Reflection High-Energy Electron Diffraction, used for monitoring the process of thin films growth, is very sensitive to growth and form of the epitaxial layer growth front, and the intensity of reflected electron beams is a function of the degree of surface atoms arrangement and substrate temperature. Such changes in intensity result from fluctuations in surface roughness, and nature of these changes can reflect the way of formation and arrangement of surface atoms in the subsequent layers.

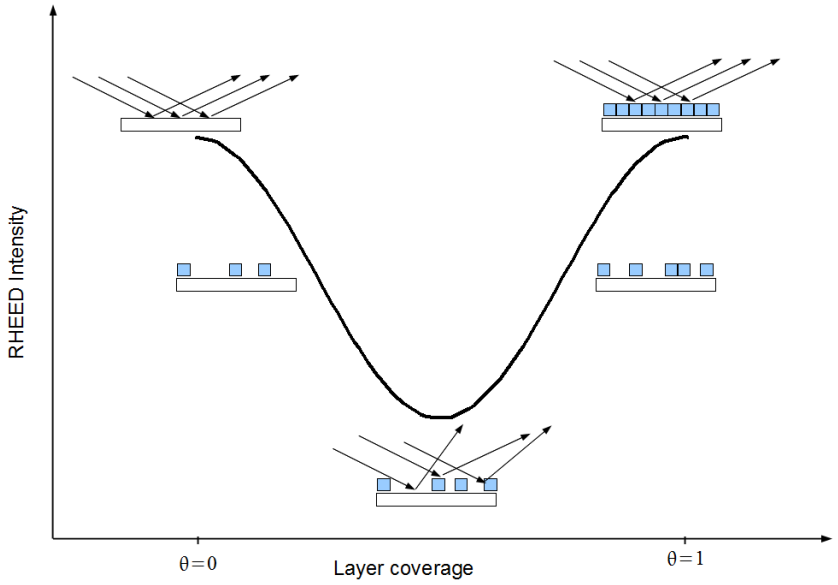


Fig. 1. A qualitative description of changes in RHEED intensity oscillations [3]

Oscillations of specular beam intensity were observed for the first time during molecular beam epitaxy (MBE) by Harris et al. [J.J. Harris, B.A. Joyce, P.J. Dobson, Surf. Sci. 108 (1981) L444]. Cohen et al. [P.I. Cohen, G.S. Petrich, P.R. Pukite, G.J. Whaley, A.S. Arrot, Surf. Sci. 216 (1989) 222] explained those oscillations by means of kinematic diffraction theory. In the kinematic approximation, the oscillating coverage of a growing layer is the structural parameter and the interference of electron beams reflected off terraces of different heights is reasoned to cause the RHEED intensity oscillations. Another approach treat diffraction as a multiple scattering process and considers the oscillation mainly as a consequence of contributions of the diffuse scattered potential components, which are varying periodically. Maksym and Beeby [P.A. Maksym, J.L. Beeby, Appl. Surf. Sci. 11–12 (1982) 663] developed an approach to RHEED theory in which the electrons are taken to be diffracted by a potential which is periodic in the two dimensions parallel to the surface but which need not be periodic in the dimension perpendicular to the surface. Peng and Whelan [L.M. Peng, M.J. Whelan, Surf. Sci. Lett. 238 (1990) L446] developed another

approach in which the electrons are taken to be diffracted by a potential which is periodic in the dimension perpendicular to the surface.

The origin of the RHEED intensity oscillations has been widely studied in the literature, however there are still issues on which no consensus has been reached in the interpretation of RHEED intensity oscillations such as the origin of oscillations and changes its phase.

Some authors origin of RHEED intensity oscillations explain using concept of interference between waves diffracted from two layers which a monolayer height difference at an off-Bragg condition [C.S. Lent, P.I. Cohen, Surf. Sci. 139 (1984) 121]. Another possible origin is diffuse scattering due to roughness of the surface [J.H. Neave, B.A. Joyce, P.J. Dobson, N. Norton, Appl. Phys. A31 (1983) 1]. Yet another possible origin is changes in the surface step density because the roughness increases as step density increases [S. Clarke, D.D. Vvedensky, Phys. Rev. B37 (1988) 6559]. Shin and Azis [B. Shin and M. J. Aziz, Phys. Rev. B76 (2007) 165408] proposed the model for the RHEED intensity oscillations in multilayer growth that includes the layer interference between terraces of different heights using the kinematic approximation and the diffuse scattering off steps on the surface. However, all of them failed to explain the fact that the phase of oscillations showed dependence on diffraction conditions [J. Zhang, J.H. Neave, P.J. Dobson, B.A. Joyce, Appl. Phys. A 42 (1987) 317]. Very recently Fuhr and P. Müller [J.D. Fuhr, P. Müller, Phys. Rev. B84 (2011) 195429] on the basis of computer simulations showed that the phenomenon of RHEED oscillations result from the angular dispersion of the incident beam and the periodic oscillation of the size of the growing islands.

In my scientific achievements of three papers describe the physical basis and represent a complete description of the problem domain:

[1] A. Daniluk, *Dynamical calculations for RHEED intensity oscill*ations, Computer Physics Communications 166 (2005) 123.

[2] A. Daniluk, *An extension of the computer program for dynamical calculations of RHEED intensity oscillations. Heterostructures*, Computer Physics Communications 176 (2007) 70.

[3] A. Daniluk, *RHEED intensities from two-dimensional heteroepitaxial nanoscale syste*ms, Computer Physics Communications 185 (2014) 3001.

The publication [1] describes the general model the problem domain. The algorithm implemented in [1] lets you iterate finding solutions of the Schrödinger equation (the energy and wave functions) as well as the periodic potential, which is required as an input to further calculations. The publication [2] implements the expanded calculation model for heterojunction structures of type diamond structure/FCC structure. In particular, this work implements an original method of self-consistent scattering potential for multilayer structures of lead layers deposited on a silicon substrate. However, at this stage it does not take into account the possibility of the occurrence of diffuse scattering. Publication [3] contains a description of the

problem domain (with an example implementation), taking into account the possible forms of diffuse scattering. Subsequently I applied described models [3] to growth of Ag on Si substrate. Growth of high-quality ultra-thin Ag film is of great interest from both scientific and technological viewpoints. Especially, ultra-thin metal films are model systems utilized to investigate quantum size effects (QSE). Moreover, ultra-thin Ag films can act as an excellent substrate for integrating various nano and low-dimensional structures.

Substrate temperature is an importance parameter controlling nucleation and growth dynamics during the thin film epitaxy in MBE system. Therefore, a very important factor to be taken into account during theoretical calculations of RHEED intensity oscillations is correct adoption of the model of lattice thermal vibrations. In the paper:

[4] P. Mazurek, A. Daniluk, K. Paprocki, *Substrate temperature control from RHEED intensity measurements*, Vacuum 72 (2004) 363,

which is a combination of experiment and calculations it was shown that the Debye's model should be consistently applied in performing advanced calculations of electron RHEED intensity changes:

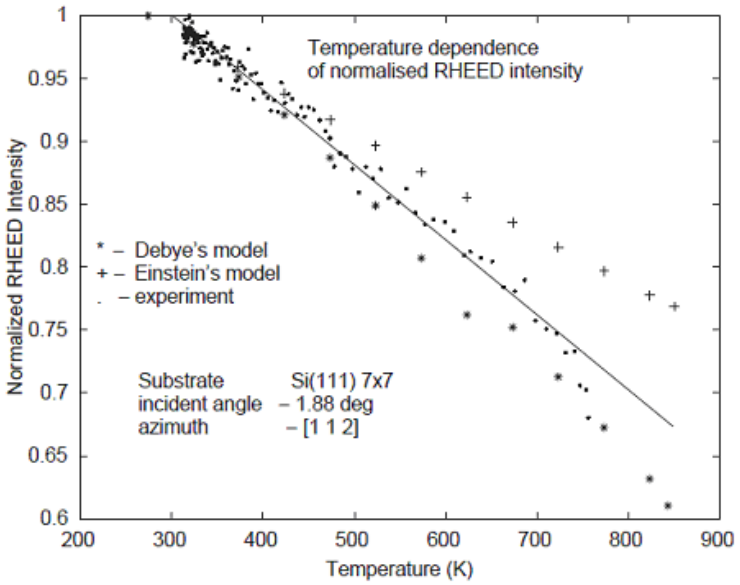


Fig. 2. Theoretical and measured temperature dependence of normalized RHEED intensity for Si(111). Experimental points are close to data calculated for Debye's model of mean square displacement of atoms in silicon Si(111) crystal lattice [4]

In this work it is also shown that it is possible to use RHEED intensity measurements to determine the crystalline substrate temperature from temperature dependence of normalised RHEED intensity as *in situ* technique.

Since the discovery of molecular beam epitaxy, which allows formation of layers (crystals) with a controlled composition of subsequent monolayers, the research attempts of many laboratories have been

focused on studying the dynamics of formation of both monolayers and specific heterojunction structures, that is the structures with monocrystalline interfaces created by materials of various chemical compositions (different physical and chemical properties). The fundamental scientific problem in such research is to specify both interface type and growth mechanism for subsequent layers. In order to model various processes occurring during the growth in crystalline heterostructures, as has been presented in the papers:

[5] P. Mazurek, A. Daniluk, K. Paprocki, *Analysis of RHEED intensities during formations of the CaF_2/Si(111) and MgO/YSi_{2-x}/Si(100) interface*, Vacuum 57 (2000) 229.

[6] P. Mazurek, A. Daniluk, K. Paprocki, *Forming the high quality $CoSi_2$ by solid phase epitaxy*, Optica Applicata 32 (2002) 389.

In [5-6] the results of experimental studies on the use of electron diffraction RHEED during formation of high-quality heterojunction structures of *CaF_2/Si(111)*, *MgO/YSi_{2-x}/Si(100)* and *$CoSi_2$/Si(111)* were presented. The experimental data were compared with the theoretical ones of my authorship. From dynamical RHEED measurements of growing CaF_2, the temperature range and flux intensity (when one type of interface was formed) have been found. It has been therefore confirmed that, for deposition at high temperature, the CaF_2 dissociates to give Si-Ca-F at the interface. From rocking curve measurements also surface of epitaxial CaF_2 layer was flat (height step up to 2 threelayer on large area) appropriate for M-I-S tunnelling structures. In the case of MgO epitaxy best results were obtained on Si(100) surface with thin 60-100 Å yttrium silicide buffer layer. This buffering silicon surface technique allows to rebuild Si(100) surface even when surface is not atomic clean, and opens the possibility of MgO growth on commercially large 2 inch diameter Si substrate in MBE.

From the application point of view epitaxial silicide materials form metal-semiconductor junctions which can be used as components of many microelectronics devices including ones based on metal-semiconductor (S-M-S) structures. $CoSi_2$ is an important metallic contact material for the semiconductor industry, it has a number of extremely useful properties. Paper [6] presents a practical experimental procedure for the preparation of high-quality thin epitaxial films of cobalt silicide. Growth of $CoSi_2$ on Si(111) substrate by solid phase epitaxy (SPE) technique is a complex process with a number of important steps taking place during the growth process. The employed growth conditions turned out to be optimal, and therefore we found it feasible to the RHEED oscillations.

Publications [4-6] formed the basis of the validation of the models created by me (Figs. 6-7 in further part of self-presentation).

II.1. Mathematical model of the problem domain

The basic mathematical model of the problem domain, and the applied mathematical framework has been published in:

[1] A. Daniluk, *Dynamical calculations for RHEED intensity oscillations*, Computer Physics Communications 166 (2005) 123.

Numerical calculations of the intensity of elastic scattering are based on the one-dimensional Schrödinger equation:

$$\left(\frac{d^2}{dz^2}+\frac{2m_0E}{\hbar^2}\right)\psi(z)=\frac{2m_0}{\hbar^2}U(z)\psi(z), \qquad (1)$$

where $\hbar$ is the Planck's constant, m_0 is the electron rest mass, E is the kinetic electron energy, $U(z)$ is the one-dimensional scattering potential of the crystal proportional to the layer coverage, and $\psi(z)$ is the electron wave function. A one-dimensional potential is found by averaging the three-dimensional potential $U(\bar{r})$ in the planes parallel to the film surface. Therefore the one-dimensional potential can be treated as the *0*th term of the Fourier series

$$U^n(z)=-\left(1+\frac{E}{m_0c^2}\right)\times\theta_n(1+\alpha i)\times\frac{8\pi}{S_0}\times\sum_{k=1}^{4}\sqrt{\frac{\pi}{b_k}}a_k\times\left[\exp\left(-\frac{4\pi^2(z-z_n)^2}{b_k}\right)\right], \qquad (2)$$

where a_k and b_k are the parameters of the analytic representation of the electron scattering factors, θ_n is the site occupation probability (sometimes also called the coverage ratio of the layer), i.e. the ratio of the number of places occupied by atoms to the total number of lattice sites (within the range 0-1), S_0 is the area of a two-dimensional unit cell parallel to the surface, z_n is the *n*th layer position along the axis perpendicular to the surface.

The dynamical diffraction calculations presented in [1-3] utilize the systematic reflection case in RHEED, in which the atomic potential in the planes parallel to the surface are projected on the surface normal, so that the results are insensitive to the atomic arrangement in the layers parallel to the surface. This model shows a systematic approximation in calculating dynamical RHEED intensities under the one-beam condition, in which the azimuth of the incident beam direction is set to several degrees off a certain crystallographic direction, fast electrons may be treated as being scattered by the averaged potential of atomic layers parallel to the surface. Under these conditions, the crystal potential can be treated as a one-dimensional potential in the surface normal direction.

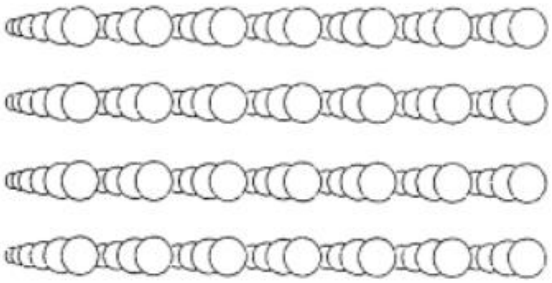


Fig. 3. Projection of a crystal lattice under one beam condition [A. Ichimiya, H. Nakahara, Y. Tanaka, *Surface structures during silicon growth on an Si(111) surface*, in: *Advances in the Understanding of Crystal Growth Mechanisms* edited by T. Nishinaga, K. Nishioka, J. Harada, A. Sasaki, H. Takei, 1997 Elsevier Science B.V.]

Currently, most authors dealing with RHEED calculations using the model of the scattering potential at one beam condition. The proportional model of the scattering potential described and implemented in [1] is a certain idealized form of reality. In real situations, the incident electron beam is scattered by the island edges. An interesting form of diffuse potential for crystalline *homoepitaxial* structures was previously proposed by Mitura and co-workers [Z. Mitura, S. L. Dudarev, L. M. Peng, G. Gladyszewski, M. J. Whelan, *The small terrace size approximation in the theory of RHEED oscillations*, J. Cryst. Growth 235 (2002) 79].

In the publication:

[3] A. Daniluk, *RHEED intensities from two-dimensional heteroepitaxial nanoscale systems*, Computer Physics Communications 185 (2014) 3001,

I have included the possibility of the occurrence of the diffuse scattering. This paper (along with the implementation) presents a new model of the scattering potential for the *heteroepitaxial* systems. This model takes into account three potential forms of diffuse scattering on the topmost monolayer. Each of the proposed forms of the model of the scattering potential is illustrated by the corresponding pseudo-code.

The proposed model of the scattering potential for heterostructures is as follows:

$$U_{Combined}(z)=\sum_{n}U^{substrate}(z)+\sum_{n}\left(U^{layer}(z)+U^{layer}_{add}(z)\right), \tag{3}$$

where the part responsible for the diffuse scattering on the topmost monolayer is:

$$U^{n}_{add}(z)=-\left(1+\frac{E}{m_0c^2}\right)\times W(\theta_n)i\beta\times\frac{8\pi}{S_0}\times\sum_{k=1}^{4}\frac{a_k}{100^{1/2}}\times\left[\exp\left(-\frac{4\pi^2(z-z_n)^2}{100}\right)\right], \tag{4}$$

where the $W(\theta_n)$ function defines the degree of diffuse scattering.

MODEL1

Assuming that during perfect layer-by-layer growth each site of the *n*th layer may be filled with θ_n probability or empty with $1-\theta_n$ probability, the $W(\theta_n)$ function that defines the degree of diffuse scattering takes the following form:

$$W(\theta_n)=(1-\theta_n)\theta_n, \tag{5}$$

and depends solely on the number of filled and empty places on the surface. This model corresponds to the perfect layer-by-layer growth mode.

MODEL2

Considering the occurrence of distribution growth mode, the probability that a deposited atom will occupy the top of the *n*th layer depends on the perimeter of this layer $d_n(\theta_n)$.

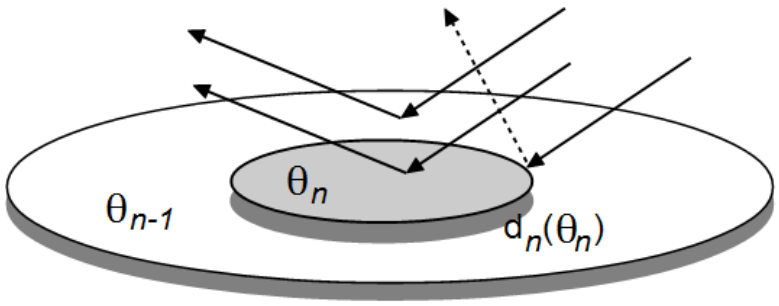


Fig. 4. Illustration of diffuse scattering on island edge. Incoming electrons arriving at island edge are nonspecularly scattered (dashed line) [3]

Assuming that in the case of the surface which is less than half filled, there is a definite number of islands and each of them has the same perimeter $d_n(\theta_n)$, and also that in the case of the surface which is more than half filled there is a definite number of empty sites with the same perimeter, the following can be written:

$$W(\theta_n)=d_n^{(1)}(\theta_n), \tag{6}$$

where:

$$d_n^{(1)}(\theta_n)=\begin{cases}\theta_n^{1/2} & for \quad \theta_n \le \theta_c \\ (1-\theta_n)^{1/2} & for \quad \theta_n > \theta_c\end{cases}, \tag{7}$$

with the assumptions: $\theta_c^{p1} = (1-\theta_c)^{p2}$, $p_1 = p_2 = 0.5$ and $\theta_c = 0.5$. In this model one can notice a considerable flexibility, which is dependent upon the way in which the perimeter of the film is linked to the coverage of this particular film.

MODEL3

If we assume that the growth takes place in small islands (small crystallites), the $W(\theta_n)$ function can be defined as follows:

$$W(\theta_n) = d_n^{(2)}(\theta_n), \tag{8}$$

where:

$$d_n^{(2)}(\theta_n) = \theta_n(1-\theta_n)^{1/2}, \tag{9}$$

which corresponds to a growth model in which both the number and size of the nucleation sites change during the film growth.

In [3], the layer growth kinetics has not been studied. The growth models are adopted in a qualitative way. The aim of this study was to demonstrate the applicability of the potential (3) and the consequences of such approach. Also demonstrated that developing more advanced models of the diffuse scattering, might be important in achieving a fully quantitative explanation of the experimental data, at least at off-symmetry azimuths.

Presented in [3] models assume that in the general case one can explain qualitatively the origin of RHEED oscillations as the superposition of effects due to: (1) periodic structure of the crystal lattice with *particular reference to periodic variations of the average scattering potential of the surface layer*, and (2) part of the electron beam is reflected diffusely by periodic oscillation of the size of the growing islands.

It should also be noted that the general assumptions of the model of the potential presented in [3] corresponds very well to the results published in parallel by Kawamura and Maksym [T. Kawamura, P.A. Maksym, *Origin of RHEED intensity oscillation during homoepitaxial growth on Si (001)*, Surface Science 630 (2014) 125], where the authors showed that are only two possible origins of the intensity variations. One is the interference between electron waves diffracted by the top and the subsequent underlying layers, and the other is the disturbance of the RHEED electron waves by step edges.

It has been shown that using the measurements of rocking-curve type, it is possible to experimental study of percolation transitions during the process of forming epitaxial heterostructures – which is very interesting from practical and scientific points of view. Also demonstrated that developing more advanced models of the diffuse scattering, might be important in achieving a fully quantitative explanation of the experimental data, at least at off-symmetry azimuths.

In my opinion, based on the publication [3] you can design and carry out a very interesting series of experiments.

Numerically the problem of calculating the changes of RHEED intensity oscillation from the growing layers is an NP problem. The time-computational complexity of the presented algorithm depends on the number of layers for both the substrate and the growing layers included in the calculations. The time-computational complexity of the presented solution is $O(n^2)$ for heteroepitaxial layers, and the memory complexity is $O(n)$, where n is the total number of layers used in the calculations.

III. Software design for scientific purposes

An important part of my work relates to the research on the design and implementation methods of software for scientific purposes based on the model-driven paradigm of architecture. Model-Driven Architecture (MDA) constitutes a modern and extremely efficient method of improving the process of generating software. It was created at the beginning of the twenty-first century by the Object Management Group (OMG) as an element of Model-Driven Development (MDD), a highly promoted trend in software engineering. MDA is the best-known MDD standard. MDA's goal is to develop standards that support a flexible approach to developing systems in the face of changing infrastructure and changing business requirements.

It should be noted that the term "Architecture" in Model-Driven Architecture does not refer to the architecture of the system being modeled, but to the architecture of various standards and model forms that serve as the technology basis for MDA.

The essence of MDA is to replace the 20th century classical programming approach, stating that "everything is an object" with the modern one – "everything is a model." It is worth noting that despite the fact that the standard model-driven software has been functioning in software engineering for over a decade, the concept of its use on a wider scale in applied sciences has not been fully appreciated. The authors of most computational programs confine themselves to fully documenting the mathematical model on which the present software is based, neglecting other (from a practical point of view no less important) elements of the generative cycle of the project. The result is that the majority of simulation programs in social circulation are very difficult to analyse and modify or adapt to new and changing user requirements.

The description of the applied information technology with integral implementation is presented in three papers which can not evaluated separately. Individual treatment of these papers causes disruption of the relationship shown in Figures 6-7, which is a fundamental methodological error.

[7] A. Daniluk, *Visual modeling for scientific software architecture design. A practical approach*, Computer Physics Communications 183 (2012) 213.

[8] A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part I*, Computer Physics Communications 181 (2010) 707.

[9] A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part II*, Computer Physics Communications 181 (2010) 709.

The MDA paradigm is not a radical aberrance from the performing-system design, but an evolutionary step that combines various technologies for a more effective software development process. In practice, the use of MDA consists in:

- shifting of the burden of system development to a higher level of abstraction and attributing the central role to modeling;
- strict separation of the system layers;
- automatic generation of a code directly from the logical model;
- introduction of mechanisms for automatic verification and validation of code.

As can be seen in Figure 5, MDA defines four viewpoints of a system, which are designed with the specific models:

- The *Computation-Independent Model* (CIM), which is a representation of a system from the computation-independent viewpoint. In the OMG terms, this kind of model is also commonly known as a domain model or business model, where the technical details of the structure of the system are hidden or as yet not determined;
- The *Platform-Independent Model* (PIM), which is used by software architects and designers, and is focused on the operational capabilities of the system outside the context of a specific platform;
- The *Platform-Specific Models* (PSMs), which are used by software developers and programmers, and include details related to the system for a specific platform. The PSM may use a General Purpose Language like C++, Java, C#, Object Pascal, etc., or different Domain Specific Languages (DSLs);
- *Source Codes*.

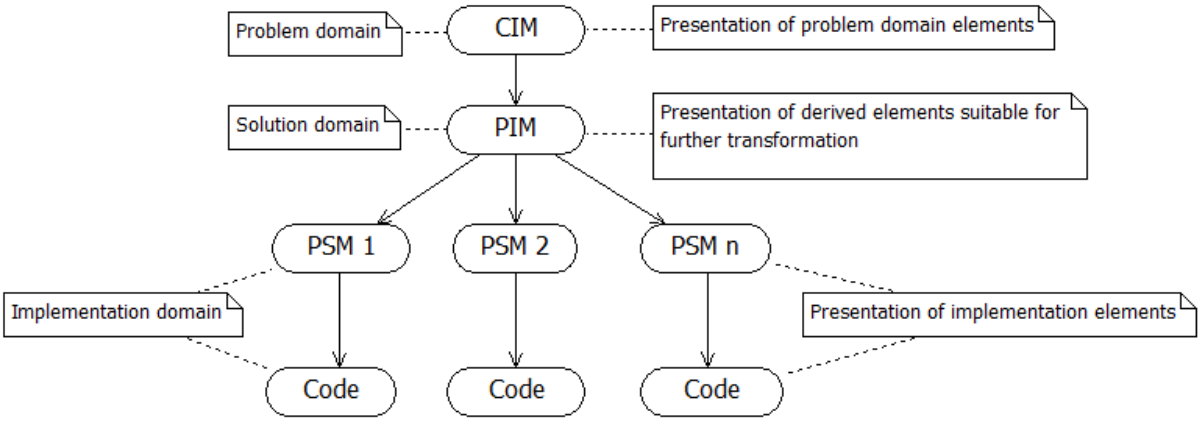


Fig. 5. The standard relationship between the layers of MDA [7]

III.1. Software development life cycle for scientific purposes

In general sense, a computer simulation is a program that is run on a computer and that uses step-by-step methods to explore the approximate behavior of a mathematical model of a real-world system. Such a computer program is a computer simulation model. A typical scientific software development process is schematically illustrated in Figure 6. This process includes choosing a mathematical model, finding a way of implementing that model in a form that can be run on a computer, calculating the results of the algorithm, and visualising and studying the resultant data.

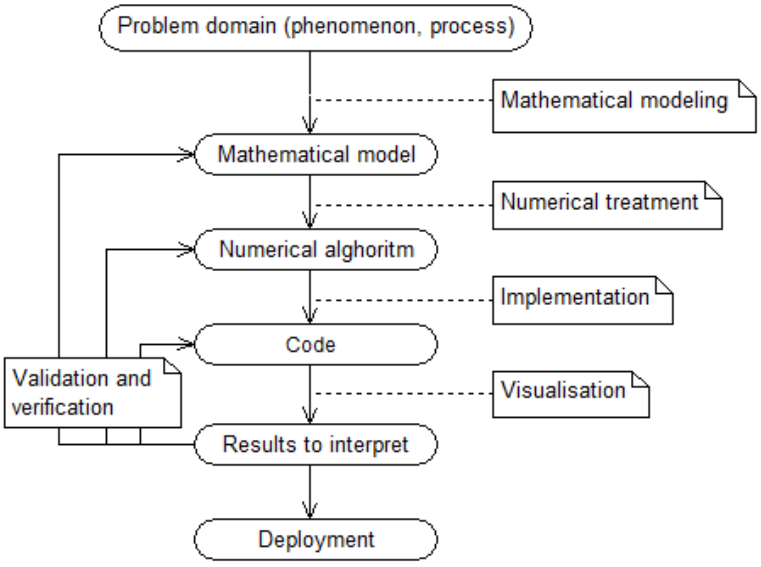


Fig. 6. A traditional scientific software development life cycle [7]

A significant drawback of the cycle shown in Figure 6 is that it does not take account of all aspects of modeling the problem domain objects, and thus prevents the modeling of complex scientific experiments.

According to [P.A. Fishwick, *Simulation Model Design and Execution: Building Digital Worlds*, (1995) Upper Saddle River, Prentice Hall]: *"A system is a set of interacting or interdependent entities, real or abstract, forming an integrated whole. In general, a system is a construct or collection of different elements that together can produce results not obtainable by the elements alone"*.

System modeling scientific experiment should allow the active modification of the parameters of phenomenon that is the subject of research in the runtime simulation, allowing the user to understand the causal relationships between the components of the course or the conditions of the studied phenomenon in real time. The problem can be summarized as follows: standard computer simulation models the phenomenon of forming the subject of the study, while virtual scientific experiment additionally takes into account the existence of infrastructure of this phenomenon. Wherein the term *infrastructure* mean relationships of elements (parts) of an entire experiment. In this context, the existence of the mathematical model is not enough, you need to build a model taking into account all the problem domain objects.

As an interesting result of investigations on the projects [7-9] is the software life cycle for scientific purposes presented by me in [7]. An important aspect of the use of this cycle is possibility of automatic creation of detailed models of the product. It should be noted that the author treats the designed program as the software for the specific industry of knowledge. The practical significance of the various levels of modeling with Fig. 7 is accurately described in [7].

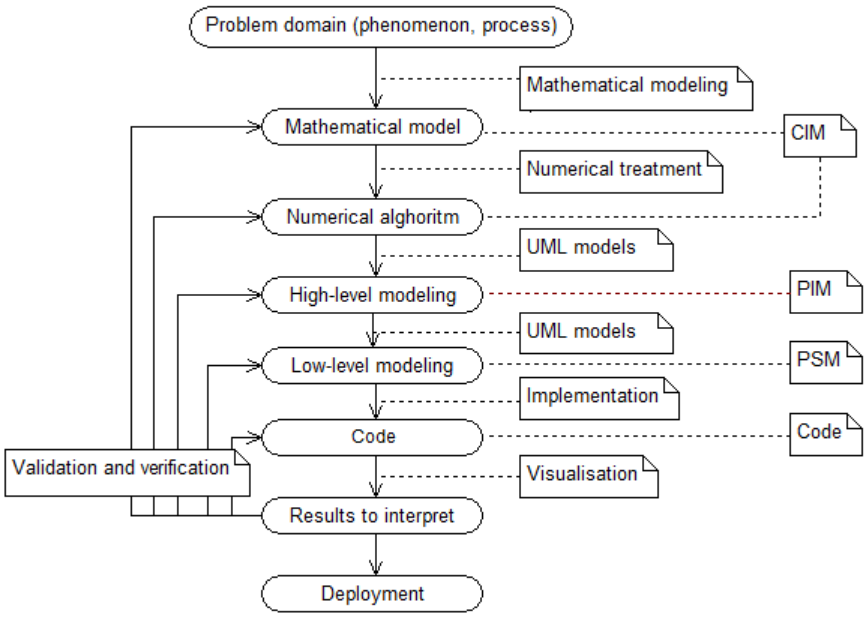


Fig. 7. The software life cycle for scientific purposes [7]

Important elements of the process shown in Figures 6 and 7 are verification and validation of the model elements. Verification is the process of determining whether the output of the simulation approximates the true solutions to the equations of the original model. In this context verification deals

with mathematics and addresses the correctness of the numerical solution to a given model. Validation is the process of determining whether the chosen model is a good enough representation of the problem domain. Validation deals with physics and addresses the appropriateness of the model in reproducing experimental data. According to Roy [S. Roy, *Recent advances in numerical methods for fluid dynamics and heat transfer*, Journal of Fluid Engineering, 127(4) (2005) 629]: "*Verification can be thought of as solving the chosen equations correctly, while validation is choosing the correct equations in the first place*".

On the other hand, taking into account the existence of the UML models of problem domain, validation or consistency checking, is also used for evaluating models in respect of semantic and syntactic quality criteria. It should be noted that not all requirements of a generated code are checked during the validation process. Some elements are verified later by the compiler. Please note that, in accordance with the assumptions of MDA, one PIM model can be a source of many PSM models (see Fig. 5).

According to the definition introduced by Johansson and co-workers [H.J. Johansson, P. McHugh, A.J. Pendlebury, W.A. Wheeler, *Business Process Reengineering: Breakpoint Strategies for Market Dominance*. John Wiley & Sons, 1993], the business process is "*a set of linked activities that take an input and transform it to create an output. Ideally, the transformation that occurs in the process should add value to the input and create an output that is more useful and effective to the recipient either upstream or downstream*"

It can be easily shown that the proposed process meets all the criteria of the standard business process, i.e.: definability, order, existence of customer of the process, value-adding during the process, embeddedness, and multi-functionality.

III.2. Metamodel

Recognizing the model as the abstraction of a reality area, the metamodel is an abstraction showing the properties of this model. Because there is a wealth of modeling languages (notations), complex metamodels are invariably problematic for researchers and developers who need to understand them - to use them effectively. These languages provide good support for the modeling process, facilitate the communication process in many application areas. Today most developers use the notation based on UML. The author used only UML with profiles provided by the OMG infrastructure in the publications.

UML has a graphical syntax; therefore it requires a visual mechanism for defining its syntax (language concepts and relations between them). Although the syntax of a visual language can be described by complex graph grammar, the easiest way is to use a metamodel. A metamodel is a model that defines the language for expressing other models. Figure 8 shows a sample metamodeling stack. Models represent different phenomena from the real world, and are instances of their metamodels. In this

terminology the languages are called metamodels, and the artifacts written in those languages (including executable programs) are called models.

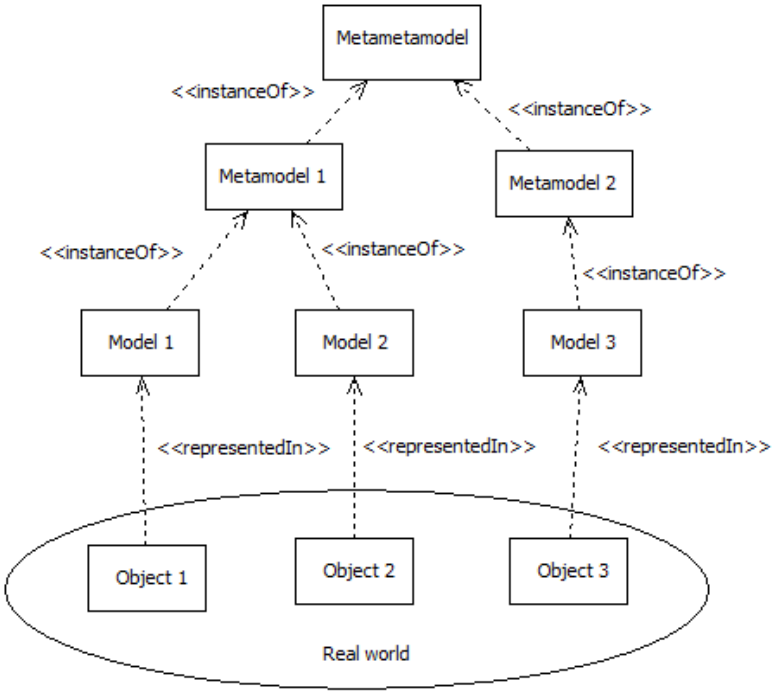


Fig. 8. Metamodeling stack

The UML metamodel defines the structure of UML models. A metametamodel (or a metalanguage) is a model that defines the language for expressing a metamodel. UML is defined in terms of a metametamodel, called the MOF (Meta-Object Facility). The MOF has repository mechanisms suitable for storing and accessing models in various languages. UML users do not need to learn the MOF to build UML models because the UML concepts are fully sufficient for the UML use. The MOF may be useful to the developers writing tools to exchange models. OMG defines a four-layered architecture for UML. Figure 9 illustrates the model-instance relationships, the mappings and the levels on which the models and their instances occur.

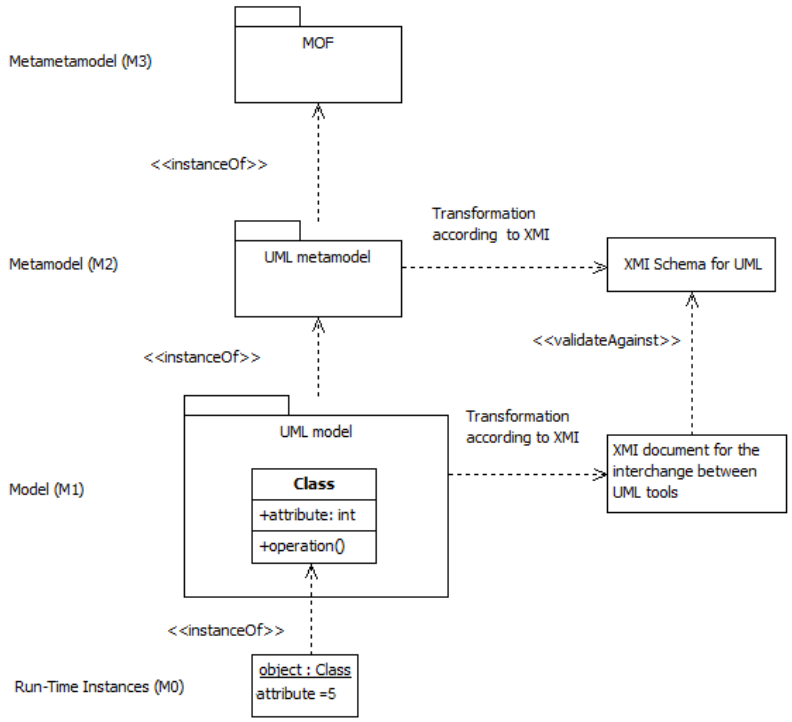


Fig. 9. OMG modeling infrastructure

The most specific layer, M0, is called the run-time instance layer. This layer only comes into play when a model results in the creation of a code. The next layer, M1, is called the model layer. The UML models created by us are particularly in this layer. XMI (XML Metadata Interchange) is an external format designed to facilitate easy interchange of metadata between modeling tools. A great number of MDA/MDD tools like: Sparx Enterprise Architect, Together, Eclipse, etc, are XMI compatible. The author used the tools of Together and Enterprise Architect in his papers.

III.3. Model of RHEED experiment

"Models represent reality, simulation imitates it."

Russell L. Ackoff, Maurice W. Sasieni, Fundamentals of Operations Research, 1968

Design of problem domain usually takes into account the domain model created during the analysis phase and aims to produce a generic architecture to which all systems within the domain can conform. The objective of problem domain design is to satisfy many requirements as possible while retaining the flexibility offered by the developed model.

Figure 10 shows how to interpret the issues discussed in [7-10] in the context of the problem domain. Level M1 depicts a beautiful relationship between physics and computer science. It turns out that

the models of physical phenomena can be represented by the relationships between classes where class (being the basic unit of knowledge) represents real-world entities with their attributes, properties and operations, and the relationships between the classes (generalization, associations, aggregations, and dependencies).

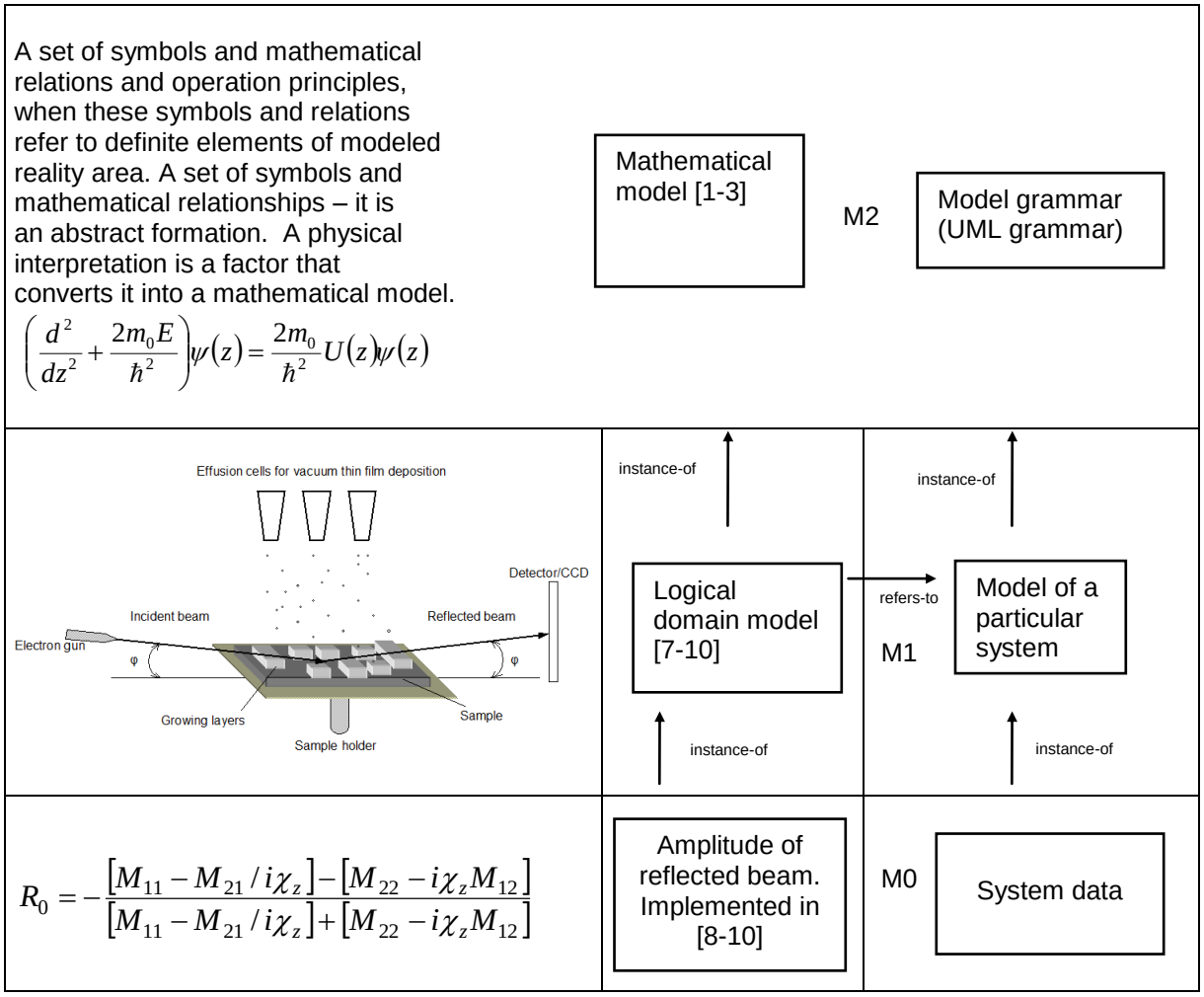


Fig. 10. Specifications significance *vs* model specification of RHEED experiment. As a model specification language used UML class diagrams, use cases and sequence diagrams

As follows from Figure 10 during the analysis of the papers the relationship between them can not be broken. The basic and extended mathematical model of the problem domain was described in [1] and [3], [4-6] contain basic test data, full specification of RHEED experiment logic model in [7], and implementation in [8-9] and [10] (together with the presentation layer).

III.3.1. The specification of the model of the RHEED experiment

A crucial element in the analysis, design and implementation of software are use cases. The use case model captures the requirements of a system, and is a means of communicating with users and other stakeholders what the system is intended to do. Use cases not only guide the analysis, but also help to

build object-oriented model of the problem domain and are especially important during software testing. Figures 11 and 12 show use cases for the *Growth09* and *RHEED1DProcess* subsystems respectively. The *Growth09* calculation model is based on the simplest kinematic diffraction theory, where it is assumed single act of scattering of electrons on the surface atoms of the crystal. This model works well on the assumption that the angle of incidence of the electron beam is small (~ 1 °) and the electron beam does not penetrate into the crystal, and the intensity of the specularly reflected beam is strongly dependent on the form of the growth front surface [A. Daniluk, *Kinematical calculations of RHEED intensity oscillations during the growth of epitaxial thin films*, Computer Physics Communications 170 (2005) 265]. The *RHEED1Process* calculation model is based on a dynamical theory of electron diffraction RHEED taking account of multiple scattering electron wave on crystal potentials.

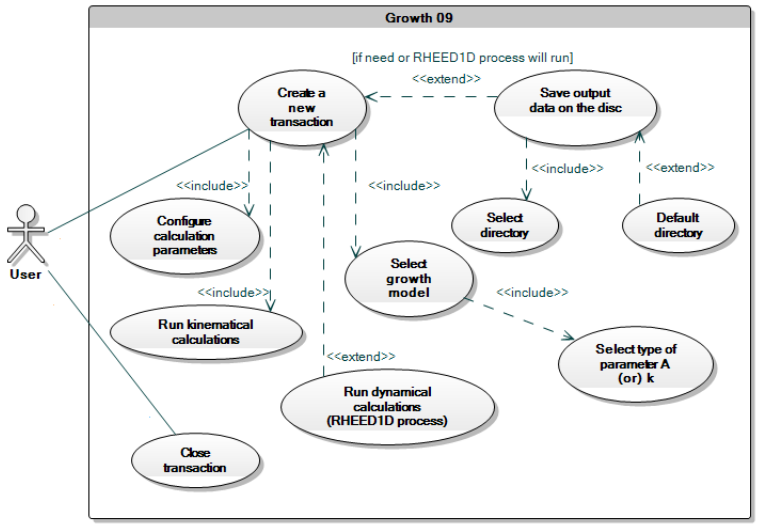


Fig. 11. Use cases for *Growth09* subsystem. Diagram is available in electronic version [8]

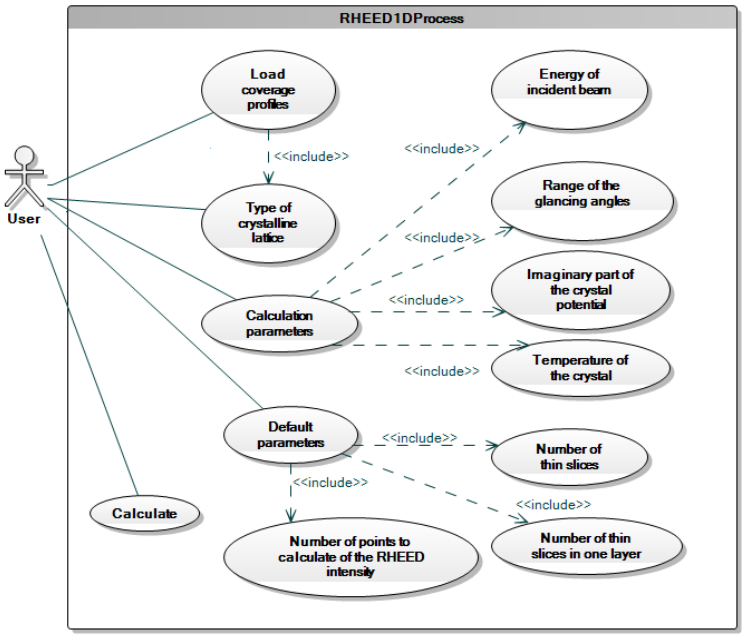


Fig. 12. Use cases for *RHEED1Process* subsystem. Diagram is available in electronic version [9]

Figures 13 and 14 show respectively the logical structure of classes for the computational environment [7-9], and the dynamic aspects of the system in the form of a suitable sequence diagram.

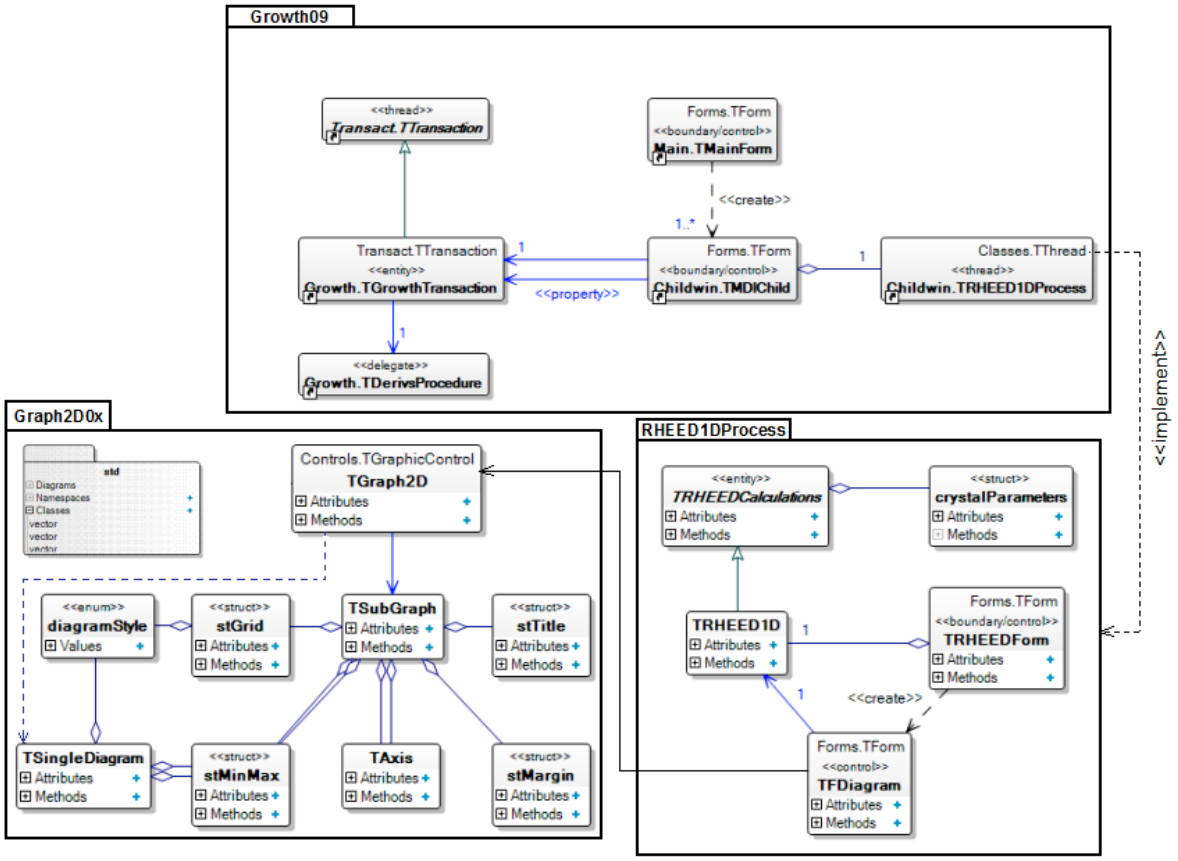


Fig.13. A logical structure of the computing environment of RHEED experiment [7]. The *RHEED1Process* package implements a design pattern for the RHEED problem domain, *Graph2D0x* package implements a design pattern for the layer presentation of RHEED and *Growth09* package - a general model for the transactional simulations of the growth process.

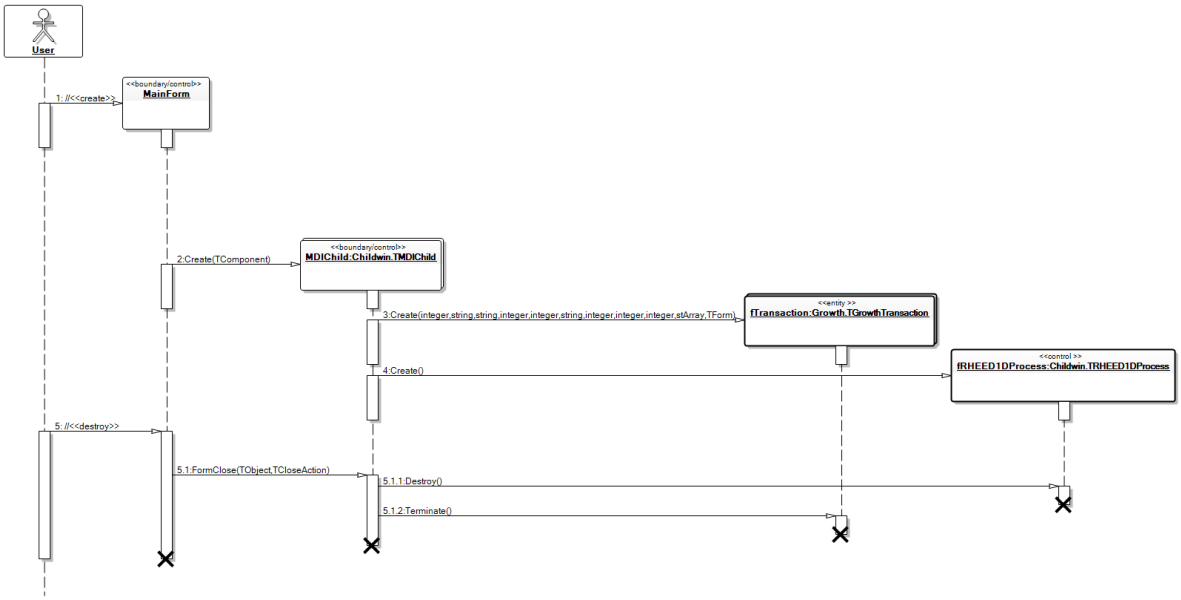


Fig.14. Sequence diagram showing the dynamic aspects of the logical model of the environment from Fig. 13 [7]

While some designers still use model specification language merely for sketching designs to share with co-workers, however model specification diagrams are more than just nice pictures. Modern modeling tool can capture the meaning of diagram elements and their relationships in machine-readable form, and use this to reason about the design, perform consistency checks, and even automatically generate parts of the application code. Creating, storing and transforming machine-readable models in this way puts modeling at the heart of the software production process, and forms the basis of Model Driven Architecture.

III.4. Reuse system artifacts

Analyzing the process shown in Fig. 7 one should recognize the potential benefit of the approach to software development for scientific purposes proposed in [7]. The advantage is the ability to reuse nearly all elements of the system – from the idioms describing individual functionality, through complex data structures as components, up to the architecture as a whole. It is important that through artifacts containing complete metadata describing the system it is possible to reproduce accurately the system in the most object-oriented languages.

The example of such approach is the paper:

[10] A. Daniluk, *Cooperative and competitive concurrency in scientific computing. A full open-source upgrade of the program for dynamical calculations of RHEED intensity oscillation*s, Computer Physics Communications 182 (2011) 1389,

which presents a general form of implementation [8-9]. It should be noted that an integral part of this project is 36-page implementation document provided by the publisher in the electronic form. Both models [8-9] and [10] are the basis for construction of a full virtual, real-time computational environment of physical experiment. This environment is distributed in the form of open source software and has developed and documented advanced graphic user interface. The openness of the environment means that all of the elements of the software can be freely used, changed, and shared (in a modified or unmodified form) by anyone. All of the elements of the environment [8-10] can be replaced by completely different solutions, e.g. the modules responsible for implementing the layer growth models and scattering potential can be easily modified or replaced by different solutions, as shown in diagram 13. It should be noted that the specific growth models (diffusive and distribution) have been used in [8-10] for the purpose of implementing the pattern, and to present the basic factors affecting the calculation of the RHEED depending on the surface model. However, the delegate from Fig. 13 (element labeled by <<delegate>> stereotype) may indicate any user-defined functions that implement e.g. the algorithms based on MC simulations (Monte Carlo) or MD (molecular dynamics) [S.-P. Kim, S.-Ch. Lee K.-R. Lee and Y.-Ch. Chung, *Molecular Dynamics Simulation at the Early Stage of Thin-Film Deposition: Al or Co on Co (111)*, Jpn. J. Appl. Phys., Vol. 43, No. 6B (2004) 3818].

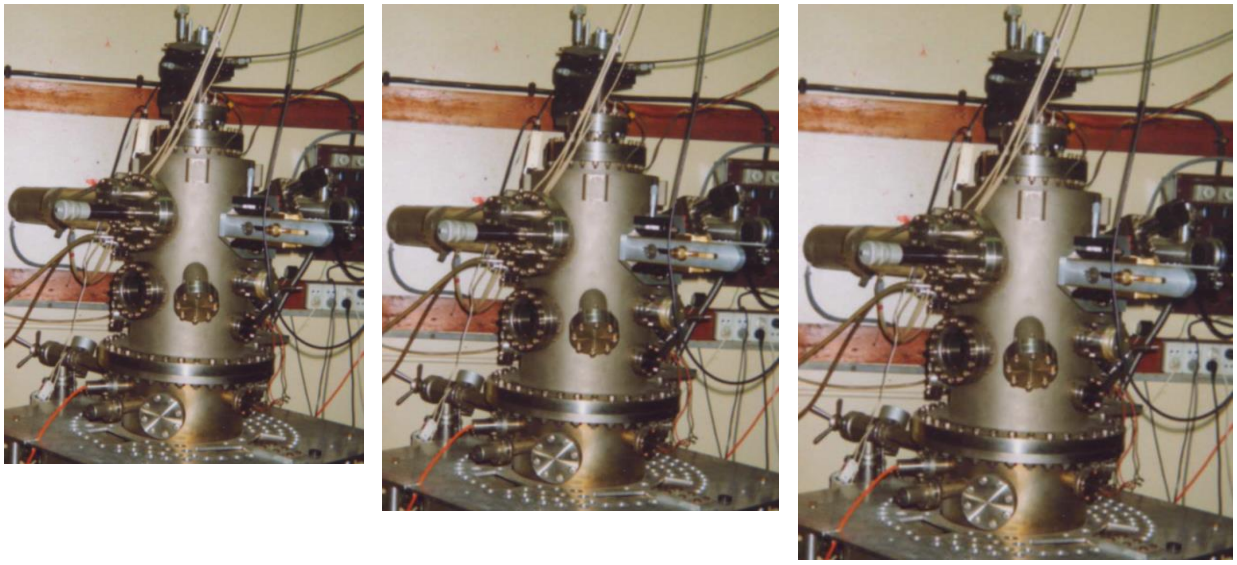
The same effect can be achieved by adding to the existing project the *.dll* (or *.lib*) library that implements a new, currently desired functionality (e.g. new models of the scattering potential). It should be also noted that the project [10] provides options for external data assimilation (e.g. experimental) available through file operations.

III.5. Multithreaded transactions

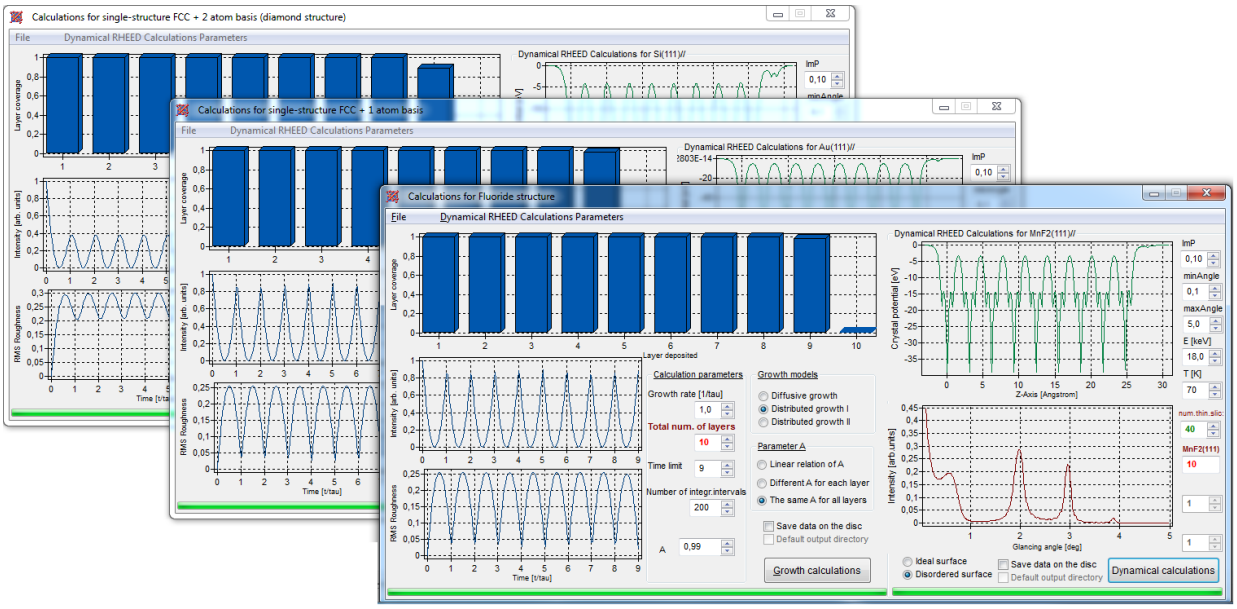
Sequential development of multithreaded computational applications is a complex procedure. Above all, knowledge about potential thread race, synchronisation and shared-variables issues is required. The method I proposed, based on a transaction-computing model in simulation programs significantly simplifies the idea of using threads while designing such software. The term "transaction" acquired a very specific meaning in computer science – it denotes any operation that exhibits the following four ACID features:

- A – atomicity – no intermediate stages in executing a transaction are observable. It means that a transaction is either fully executed or not executed at all. A transaction can be accepted (terminate on a normal basis) or cancelled (rejected).
- C – consistency – observable status changes do not influence system constants. Consistency is connected with atomicity – the difference lies in the fact that the application developer is responsible for consistency, while the responsibility for indivisibility lies with the implementation of the transaction system.
- I – isolation – multiple transactions can be executed simultaneously without mutual interference. Their execution resembles operations executed in a sequential manner.
- D – durability – the result of program operation is stored after its termination (e.g. in the form of data collection on the disk).

Except for complex systems of a database nature, all four ACID features are not always required for the normal operation of the simulation program. In the applications [8-10] I have designed the possibility of working in the so-called "light transaction" mode. It is characterised by all ACID features but the program may as well function correctly without the need to comply with the durability rule. A transaction can be terminated at any time without causing a conflict within the computed calculations.
Implementations [8-10] are based on the use of the template method supplemented with the concept of delegates. The logical model of the RHEED experiment used in [10] allows you to perform multiple virtual experiments, as shown in Fig. 15.



(a)



(b)

Fig.15 (a) Hypothetical example of the laboratory with much of the RHEED equipment where at the same time you can perform a variety of experiments comparing their results. (b) A virtual, multithreaded equivalent of different RHEED experiments for example of running application [10]

The design of the environment [10] based on the concept of multithreaded transactions makes it possible to perform calculations with regard to the possibility of manipulation of selected parameters of virtual experiment in real-time. In particular, the implementation of model [10] allows you to:

- performance of "sample transfer" (sample data for selected homo and heteroepitaxial structures are stored in its own application database;
- manipulation of variables (factors) describing: selected basic types of layer growth, the temperature of substrate layer, the energy of the incident beam, the range of angle of incident beam, and many others;
- controlling of disturbance level of the scattering potential of the crystal;
- measurements of the intensity of the diffracted beam.

III.6. Possibilities of wider applications

A pattern is a general recurrent solution to a commonly occurring problem in the software design. According to Alexander, Ishikawa and Silverstein [Ch. Alexander, S. Ishikawa and M. Silverstein, *A Pattern Language, Buildings, Construction*, Oxford University Press, 1977]: "*Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to the problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice*".

The design pattern is a category of patterns that deals with the object oriented software. They represent solutions of problems that arise when developing software within a particular context. The design pattern depicts the static and dynamic structure and collaboration among key participants in software designs.

Efficient implementation of algorithms (e.g. for solving sets of non-linear differential equations and Schrödinger equations) is a problem often appearing in applied sciences. In the *GrowthCP* project an uncomplicated design pattern based on the theory of multithreaded transactions was used. Exploiting this pattern significantly facilitates implementation of different kinds of algorithms. Implementation of the pattern was tested in uncomplicated syntactically and semantically modern programming language, so is simple for understanding, modification, and implementation in other object-oriented languages. Moreover, the *GrowthCP* project presents a possible way of implementing multithreaded transactions in modern visual event-driven programming environments.

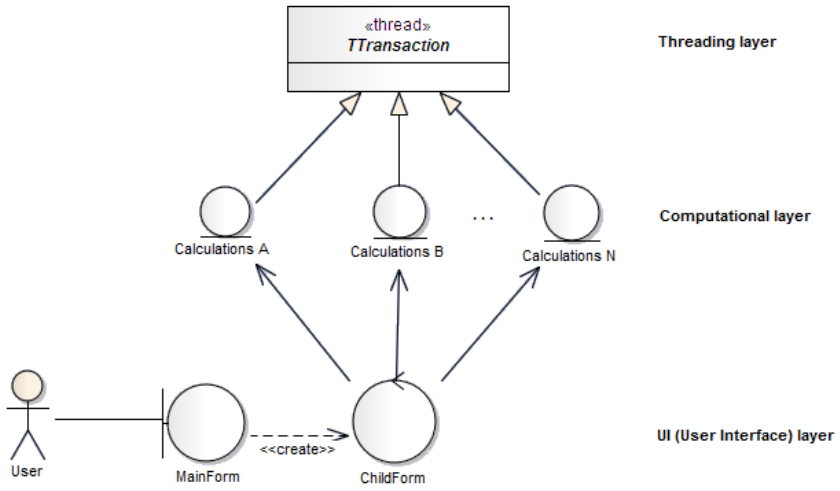


Fig. 16. A general pattern for the multithreaded calculations in the MVC implementation [10]

Generally, this pattern can be useful in situations where the need of the multithreaded work is occurring with algorithms of different types, and the presentation of data is performed according to the MVC standard (as showed on Figure 16).

IV. Summary of the main achievements

1. Participation in the development of practical experimental procedures for obtaining very good quality heterostructures CaF_2/Si(111), MgO/YSi_{2-x}/Si(100), and $CoSi_2$/Si (111) prepared by MBE and SPE methods. Materials of this type are technologically very attractive with a high potential for successful commercial applications.
2. Development (and implementation) of algorithm for the calculation of dynamic high-electron diffraction RHEED and its testing in relation to the experimental measurements carried-out in the Institute of Physics Maria Curie-Skłodowska University and data taken from the literature, e.g. [Y. Fukaya, Y. Shigeta, *Fast measurement of rocking curve of reflection high-energy electron diffraction by using quasi-1D convergent beam*, Surface Science 530 (2003) 175], [Y. Shigeta, Y. Fukaya, *Structural phase transition and thermal vibration of surface atoms studied by reflection high-energy electron diffraction*, Applied Surface Science 237 (2004) 21].
3. Formal publication and implementation of appropriate computer program [1]. At the time of publication (the paper sent for publication in 2004) was one of the three open programs of this type available in the world literature. The other projects were published at about the same time by the groups from Cambridge and Oxford: [A. Ichimiya, P.I. Cohen, *Reflection High-Energy Electron Diffraction*, Cambridge University Press, Cambridge, 2004] and [L.-M. Peng, S.L. Dudarev, M.J. Whelan, *High Energy Electron Diffraction and Microscopy*, Oxforf University Press, Oxford, 2004], but based on a different calculation formalism.
4. Proposal of a new model of the crystal potential, taking the diffuse scattering into account [3]. Using such approach, it was shown that it is possible to obtain valuable information for the analysis of experimental data. Also demonstrated that developing more advanced models of the diffuse scattering, might be important in achieving a fully quantitative explanation of the experimental data, at least at off-symmetry azimuths. Presented in [3] models assume that in the general case one can explain qualitatively the origin of RHEED oscillations as the superposition of effects due to: (1) periodic structure of the crystal lattice with particular reference to periodic variations of the average scattering potential of the surface layer, and (2) part of the electron beam is reflected diffusely by periodic oscillation of the size of the growing islands.
5. A computational approach enables the computer time scaling of a conventional RHEED calculation to be reduced to n^2, where n is the number of layers used in the calculation.
6. Development of an object-oriented computational model for the RHEED problem domain. According to the author's knowledge this is currently the only such *interdisciplinary development* available in the world literature [7-9]. Papers [7-9] constitute *de facto* the record of an innovative, general pattern for the method of solving the complex RHEED problem in the full generative cycle and the evolution of

Model-Driven Architecture for application in construction of simulation programs for scientific purposes with the advanced graphic user interface.

7. Development and testing the applicability of the software development process for scientific purposes. The scientific method presented in [7] treats the process of computational software modeling as the one consisting in construction of a sequence of models, ranging from a general, describing the problem domain, to a detailed model of specific system implementation. Transitions between the models involve the transformations of high-level abstraction models into detailed ones, specific for the final programming platform. The final elements in the transformation are the models at the level of the system code.
8. The construction of open, virtual, operating in real time transactional and multithreaded computing environment acts as a physical experiment of RHEED [10]. According to the author's knowledge this is currently the only such formally published, documented and available as an open-source environment. Implementation [10] does not take into account the possibility of occurrence of diffuse scattering [3]. However, the method of presentation of models [3] allows for the quick implementation as the element(s) of any pattern (in particular, the pattern described in [10]).

31.05.2016

V. Published original papers and books available in social circulation

After obtaining the Doctor title

40. A. Daniluk, *RHEED intensities from two-dimensional heteroepitaxial nanoscale systems of GaN on a Si surface*, Computer Physics Communications (2016) accepted for publication.

39. A. Daniluk, *RHEED intensities from two-dimensional heteroepitaxial nanoscale systems*, Computer Physics Communications 185 (2014) 3001.

38. A. Daniluk, *GRAPPLE for Measurement Systems*, Journal of Information Technology & Software Engineering 3:1 (2013) – Editorial.

37. A. Daniluk, *Modeling in the Process of Constructing the Software for Scientific Purposes*, Journal of Information Technology & Software Engineering 2:3 (2012) - Editorial.

36. A. Daniluk, *Visual modeling for scientific software architecture design. A practical approach*, Computer Physics Communications 183 (2012) 213.

35. A. Daniluk, *Cooperative and competitive concurrency in scientific computing. A full open-source upgrade of the program for dynamical calculations of RHEED intensity oscillation*s, Computer Physics Communications 182 (2011) 1389.

34. A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part II*, Computer Physics Communications 181 (2010) 709.

33. A. Daniluk, *Model-Driven Development for scientific computing. Computations of RHEED intensities for a disordered surface. Part I*, Computer Physics Communications 181 (2010) 707.

32. A. Daniluk, *Model-Driven Development for scientific computing*, 7th Conference on Computer Methods and Systems CMS'09, Kraków, (2009), 339.

31. A. Daniluk, *Model-Driven Development for scientific computing. An upgrade of the RHEEDGr program*, Computer Physics Communications 180 (2009) 2394.

30. A. Daniluk, *Multithreaded transactions in scientific computing. The Growth06_v2 program*, Computer Physics Communications 180 (2009) 1219.

29. A. Daniluk, *An extension of the computer program for dynamical calculations of RHEED intensity oscillations. Heterostructures*, Computer Physics Communications 176 (2007) 70.

28. M. Brzuszek, A. Daniluk, *Multithreaded transactions in scientific computing: New versions of a computer program for kinematical calculations of RHEED intensity oscillations*, Computer Physics Communications 175 (2006) 678.

27. A. Daniluk, K. Skrobas, *A new version of a computer program for dynamical calculations of RHEED intensity oscillations*, Computer Physics Communications 174 (2006) 83.

26. A. Daniluk, *Dynamical calculations for RHEED intensity oscillations*, Computer Physics Communications 166 (2005) 123.

25. A. Daniluk, *Kinematical calculations of RHEED intensity oscillations during the growth of thin epitaxial films*, Computer Physics Communications 170 (2005) 265.

24. P. Mazurek, A. Daniluk, K. Paprocki, *Substrate temperature control from RHEED intensity measurements*, Vacuum 72 (2004) 363.

23. A. Daniluk, *Transakcje*, Materiały IV Krajowej Konferencji Metody i Systemy Komputerowe w Badaniach Naukowych i Projektowaniu Inżynierskim, Krakowskie Centrum Informatyki Stosowanej, Kraków (2003) 691.

22. A. Daniluk, *Concurrent programming and futures*, Annales UMCS Informatica AI, 1 (2003) 303.

21. P. Mazurek, A. Daniluk, K. Paprocki, *Forming the high quality $CoSi_2$ by solid phase epitaxy*, Optica Applicata 32 (2002) 389.

20. P. Mazurek, A. Daniluk, K. Paprocki, *Analysis of RHEED intensities during formations of the CaF_2/Si(111) and MgO/YSi_{2-x}/Si(100) interface*, Vacuum 57 (2000) 229.

Before obtaining the Doctor title

19. A. Daniluk, *Komputerowy system rejestracji i analizy obrazów z kamery CCD*, Materiały II Krajowej Konferencji Metody i Systemy Komputerowe w Badaniach Naukowych i Projektowaniu Inżynierskim, Krakowskie Centrum Informatyki Stosowanej, Kraków (1999) 515.

18. A. Daniluk, *Symulacje komputerowe w czasie rzeczywistym wzrostu warstw epitaksjalnych*, Materiały II Krajowej Konferencji Metody i Systemy Komputerowe w Badaniach Naukowych i Projektowaniu Inżynierskim, Krakowskie Centrum Informatyki Stosowanej, Kraków (1999) 307.

17. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *Initial growth stage of CaF_2 on Si(111) studied by RHEED*, Electron Technology (1998), 31(3-4) 338.

16. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *RHEED intensity oscillations observed during the growth of CaF_2 on Si(111)*, Physical Review B57 (1998) 12443.

15. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *RHEED intensity oscillations observed during the growth of YSi_{2-X} on Si (111) substrates*, Surface Science 391 (1997) 226.

14. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *Monte-Carlo simulation of Ge on Si(111) MBE growth: Analysis of percolative structure*, Thin Solid Films 306 (1997) 220.

13. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *Distributed growth model used for the interpretation of RHEED intensity oscillations observed during the growth of Pb on Si(111)*, substrates, Thin Solid Films 306 (1997) 228.

12. A. Daniluk, and P. Mikołajczak, *Perkolacyjne modelowanie wzrostu Stranski-Krastanov dla Ge/Si(111)*, Abstrakt [w:] IV Seminarium Powierzchnia i Struktury Cienkowarstwowe, 18-21 IX Kazimierz Dolny (1996).

11. A. Daniluk, P. Mazurek, P. Mikołajczak, *RHEED intensity oscillations observed during growth of Ge on Si(111) substrates*, Surface Science 369 (1996) 91.

10. W. I. Gruszecki, M. Matuła, A. Daniluk, Z. Krupa, *Increased heat emission in photosynthetic apparatus of rye subjected to light stress*, Journal of Photochemistry and Photobiology B: Biology 32 (1996) 67.

9. S. Krawczyk, A. Daniluk, *Solvent effects and vibrational dependence in electrochromic spectra of carotenoids*, Chemical Physics Letters 236 (1995) 431.

8. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *Epitaxial growth of YSi_{2-x} films on Si(111) substrates*, IF UMCS Scientific Report 1995, ed. A. Baran, Institute of Physics, Maria Curie-Skłodowska University, 49, Lublin 1995.

7. A. Daniluk, P. Mazurek, K. Paprocki, P. Mikołajczak, *RHEED intensity oscillations observed for growing films Ge/Si(111) at room temperature*, IF UMCS Scientific Report 1995, ed. A. Baran, Institute of Physics, Maria Curie-Skłodowska University, 47, Lublin 1995.

6. A. Daniluk, and P. Mikołajczak, *Growth of Ultrathin Films Composed of Si/Si and Pb/Si Analysed by the Distributed Growth Model*, Abstract of the Workshop on MBE Growth Physics and Technology, December 12-14 Warsaw (1994).

5. C. W. Domański, A. Daniluk, *Z badań nad preferencjami i symboliką barw*, Annales UMCS Sect. J. Vol. 7 (1994) 37.

4. A. Daniluk, *Statystyczny program obliczeniowy. Test Psychologiczno-Pedagogiczny*, Annales UMCS Sect. J. Vol. 6, (1993) 213.

3. Z. Mitura, A. Daniluk, *Studies on RHEED oscillations at low glancing angles*, Surface Science 277 (1992) 229.

2. Z. Mitura, A.Daniluk, M. Stróżak, M. Jałochowski, A. Smal and M. Subotowicz, *Analysis of Shapes of RHEED Intensity Oscillations Observed for Growing Films*, Proceedings of XX Intemationaf School of Semiconducting Compounds, Jaszowiec, 1991, Acta Physica Polonica A80 (1991) 365.

1. Z. Mitura, A. Daniluk, M. Stróżak, M. Jałochowski, A. Smal and M. Subotowicz, *Growth of Ultrathin Films Composed of Pb and/or In Analysed by the Distributed Growth Model*, Abstract of the 15-th International Seminar on Surface Physics in Przesieka, May 20-25, A-45, (1991).

Detailed design-phase works (computer programs) published in worldwide science databases after obtaining the Doctor title: International CPC Program Library, School of Electronics, Electrical Engineering and Computer Science, Queen's University of Belfast, http://cpc.cs.qub.ac.uk/

3. Program: RHEED_DIFFx, *Catalogue Id*: AETW
2. Programs: RHEEDxxx, *Catalogue Id*: ADUY
1. Programs: GROWTHxxx, *Catalogue Id*: ADVL

Published workbooks, books and academic scripts (in Polish)

16. A. Daniluk, *USB. Praktyczne programowanie z Windows API w C++. Wydanie II*, Helion 2013.

15. A. Daniluk, *Bluetooth. Praktyczne programowanie*, UMCS 2012 – *skrypt akademicki*.

14. A. Daniluk, *Pracownia programowania obiektowego*, UMCS 2011 – *skrypt akademicki*.

13. A. Daniluk, *USB. Praktyczne programowanie z Windows API w C++*, Helion 2009.

12. A. Daniluk, *RS 232C - praktyczne programowanie. Od Pascala i C++ do Delphi i Buildera.*

Wydanie III, Helion 2007.

11. A. Daniluk, *C++Builder Borland Developer Studio 2006. Kompendium programisty*, Helion 2006.
10. A. Daniluk, *C++Builder 2006. Ćwiczenia praktyczne*, Helion 2006.
9. A. Daniluk, *C++Builder. Kompendium programisty*, Helion 2003.
8. A. Daniluk, *C++Builder 6. Ćwiczenia zaawansowane*, Helion 2003.
7. A. Daniluk, *C++Builder 6. Ćwiczenia praktyczne*, Helion 2003.
6. A. Daniluk, *ABC Delphi 7*, Helion 2003.
5. A. Daniluk, *ABC Delphi 6*, Helion 2002.
4. A. Daniluk, *Delphi 6. Nowe narzędzia obliczeniowe*, Helion 2002.
3. A. Daniluk, *RS 232C - praktyczne programowanie. Od Pascala i C++ do Delphi i Buildera.* Wydanie II, Helion 2002.
2. A. Daniluk, *RS 232C - praktyczne programowanie. Od Pascala i C++ do Delphi i Buildera*, Helion 2001.
1. A. Daniluk, *C++ Builder 5. Ćwiczenia praktyczne*, Helion 2001.

31.05.2016