

Пловдивски Университет „П. Хилендарски“
Факултет по Математика и Информатика
Катедра „Компютърни системи“

доц. д-р Александър Пенев

***Компютърна Графика в
Хетерогенна Среда***

Публична Академична Лекция
/29.11.2018/

Хабилитационна процедура в
област на висше образование 4. Природни науки, математика и информатика,
професионално направление 4.6. Информатика и компютърни науки,
Информатика - Компютърна графика



Компютърната графика (КГ) е много сложна и разнородна научна област, развивала се десетки години, с множество приложения в ИТ индустрията.

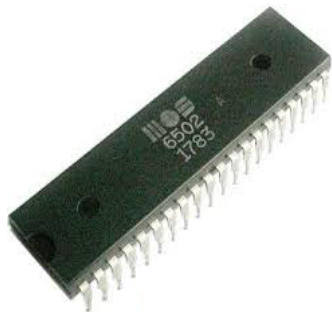
В тази лекция ще се спра на развитието на КГ (и взаимовръзката и със специализирания графичен и друг хардуер, използван за постигане на нейните цели) в периода след 1990 г. Ще се спра на кратко и на някои от научните ми интереси и резултати свързани с темата.

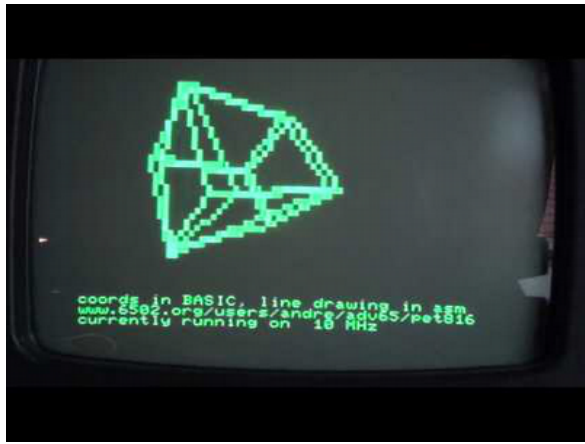
Разбира се голяма част от развитието и резултатите на КГ вече са били постигнати до този период.



Началото

Дипломна работа (1994-1996)



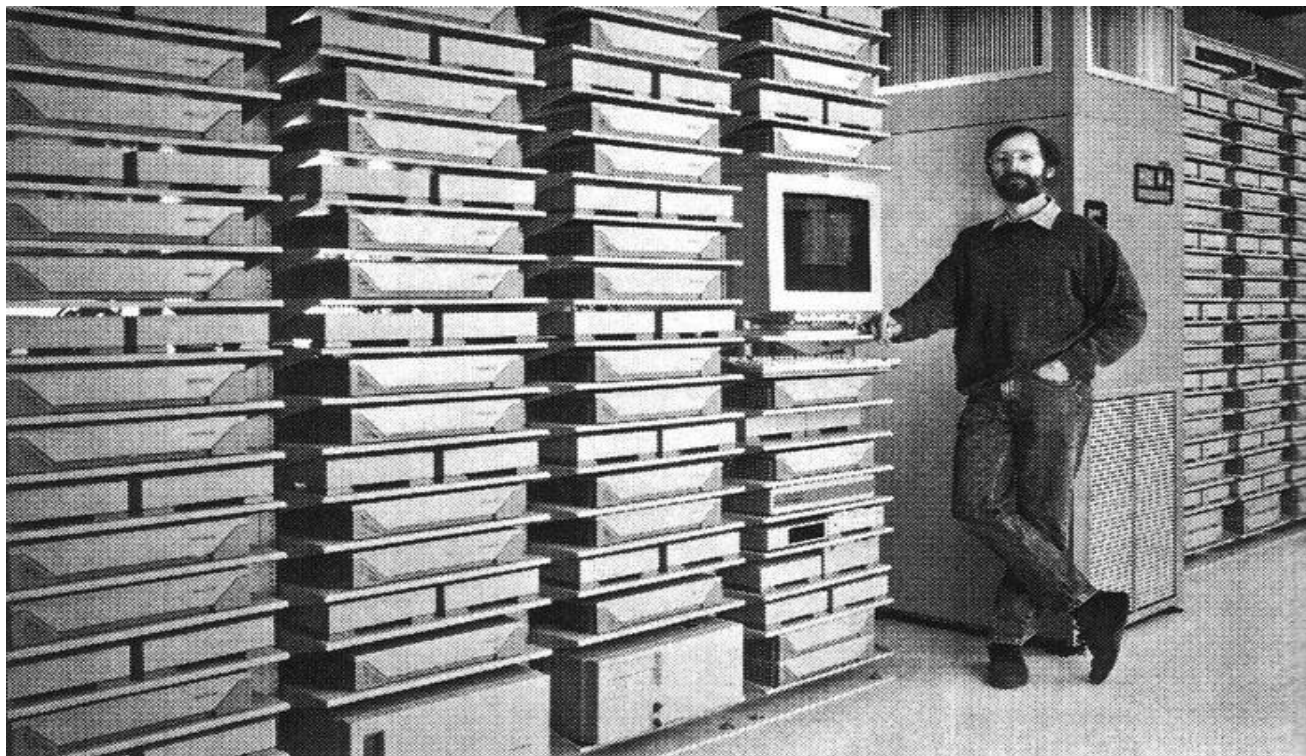


Персоналните компютри са в разцвет



SGI властва в
сегмента на графичните
станции и сървъри





Ed Catmull в рендеринг „фермата“ на Pixar, 1995



Клъстер от 117 (87 дву процесорни и 30 четири процесорни, 100 MHz)
SPARCstation 20s със от 192 до 384 MB RAM,
Всеки с 4-5GB дисково пространство.

Те работят със Solaris и софтуер Pixar's "Renderman" и
SparcServer 1000e за разпределение на задачите.



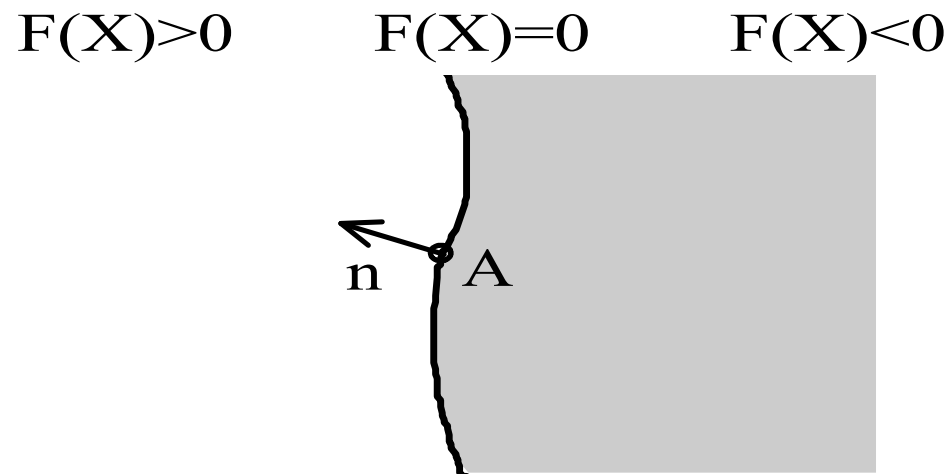
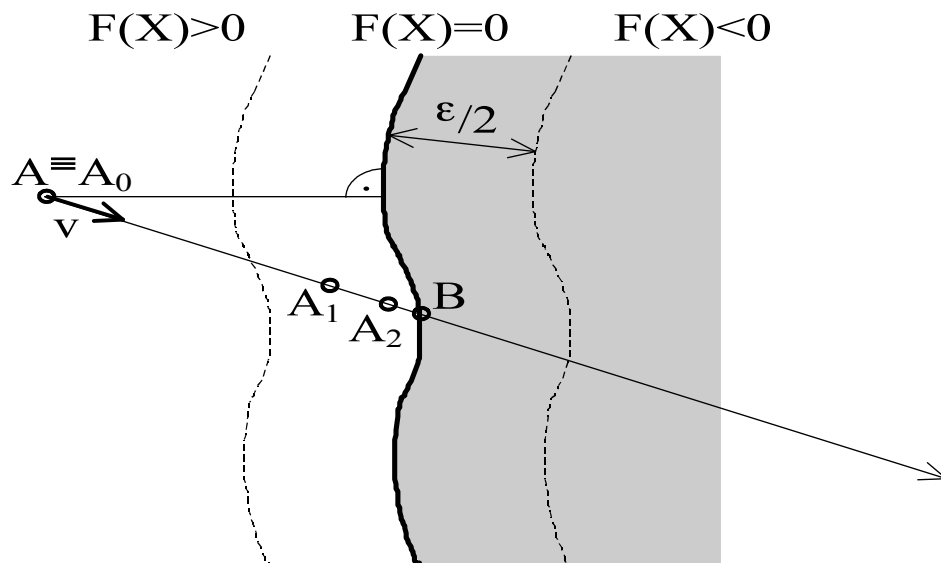
Дипломна работа

„Един подход за описание на геометрична информация“

- Базирано на Signed Distance Functions (SDF) представяне
(вариант на представяне с неявни повърхности, което се различава коренно от най-разпространения вид представяне - Граничното известно още като B-Rep)
- Алгоритъм за сечение на лъч с геометрията аналогичен на т.нар. „Sphere Tracing“ е дефиниран и се използва при визуализацията
- SDL - собствен специализиран език за описание на сцената
- GVM - графична виртуална машина



Визуализация на SDF



$$A_0 = A$$

$$A_{i+1} = A_i + v \cdot |F(A_i)|, \text{ докато } |F(A_i)| > \epsilon/2.$$



Пример

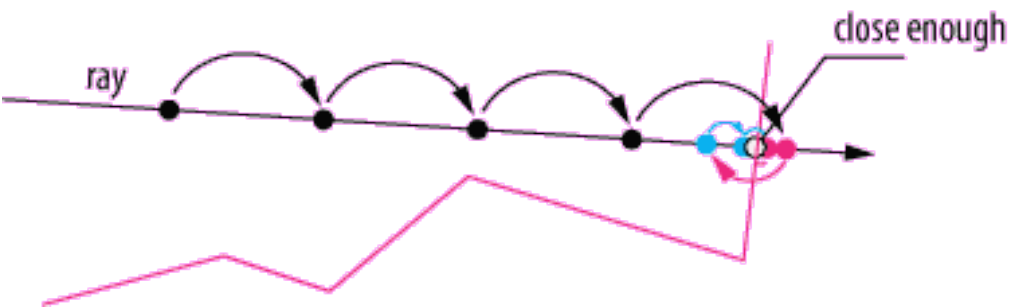
```
new LS(point,pos,vx,vy,vz,num)=[ ((point+pos)*vx.num) ,  
                                  ((point+pos)*vy.num) ,  
                                  ((point+pos)*vz.num) ] ;  
  
...  
...  
...  
new HalfSpace1()=  
[  
  property RR(point)=[0.6,0.6,0.6] ,  
  property BR(point)=[0.7,0.5,0.8] ,  
  property Res()=[1,1,1] ,  
  property KD(point)=[0.2,0.2,0.5] ,  
  property Tn(point)=5 ,  
  property LtR()=[1.3,1.3,1.3]  
] & HalfSpace([~2,0,0],[1,1,0],[1,~1,~1],[~1,0,1]) ;  
new Light1()=PointLight([0,3,~2],[90,100,80]) ;  
new LightD()=DiffuseLight([0,0,0],3) ;  
  
Show(320,200,LightD(),Light1(),Sphere1(),Checker1(),HalfSpace1()) ;
```



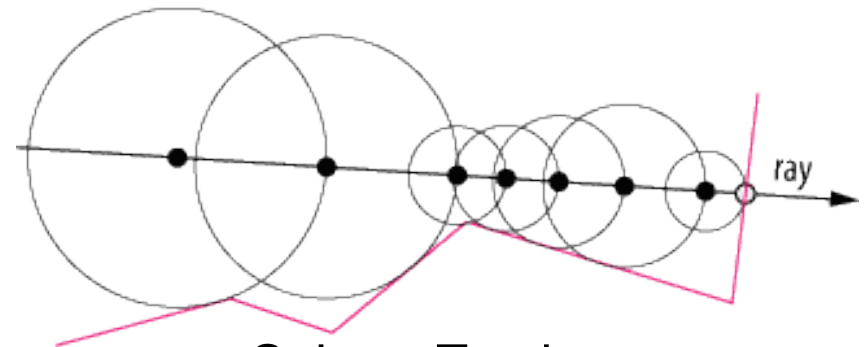
По същото време J. Hart

J. Hart, *Sphere tracing: a geometric method for the antialiased ray tracing of implicit surfaces*, The Visual Computer (1996) 12: 527.

<https://doi.org/10.1007/s003710050084>



Ray Marching



Sphere Tracing



Публикации и други

1. А. Пенев, Д. Димов, *"Един подход при работа с геометрична информация"*, XXI International Conference "Information Technologies and Programming", Пловдив, Юни, 1996
2. А. Пенев, *"Един подход за описание на геометрична информация"*, Дипломна работа, ПУ, Пловдив, Юли 1996



Дисертация

Отворени хибридни системи за
геометрично моделиране (2003-2013)



Появяват се първите съвременни GPU



nVidia GeForce 256 – един от първите съвременни GPU
(1999)

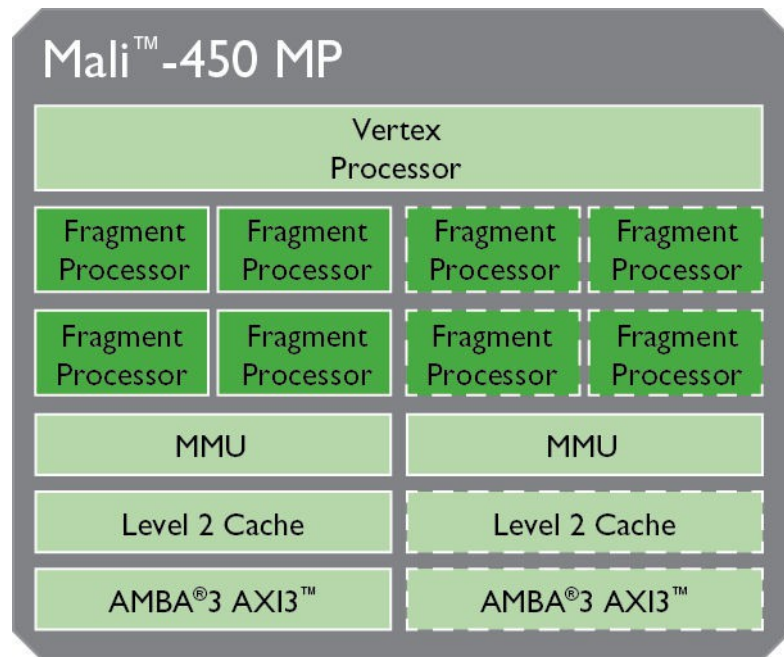
Много-ядрени Процесори



Макар голяма част от основните алгоритми в Компютърната графика да са силно паралелни, задачата за оптималната им реализация на ново-появяващият се хардуер не е никак тривиална.

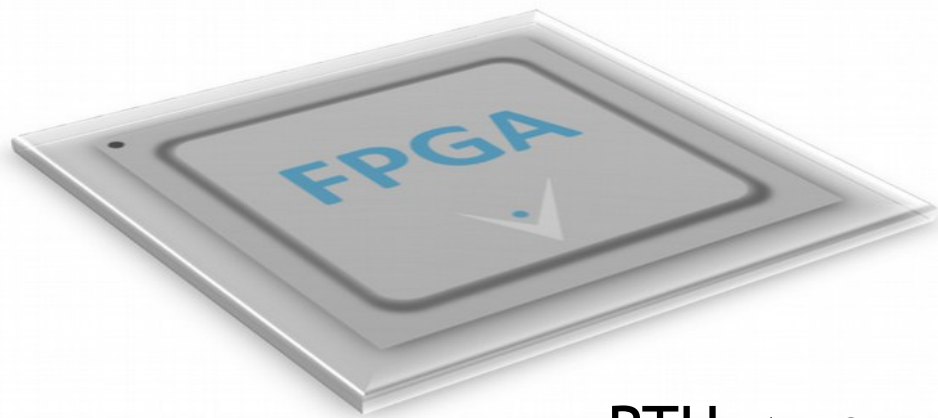


Появяват се различни Мобилни Платформи



Мобилните приложения поставят нови ограничения пред реализацията на алгоритмите в КГ прилагани в тях. Тук основен ресурс за оптимизация е енергията, консумирана от съответния алгоритъм, а не толкова класическите, време и памет.





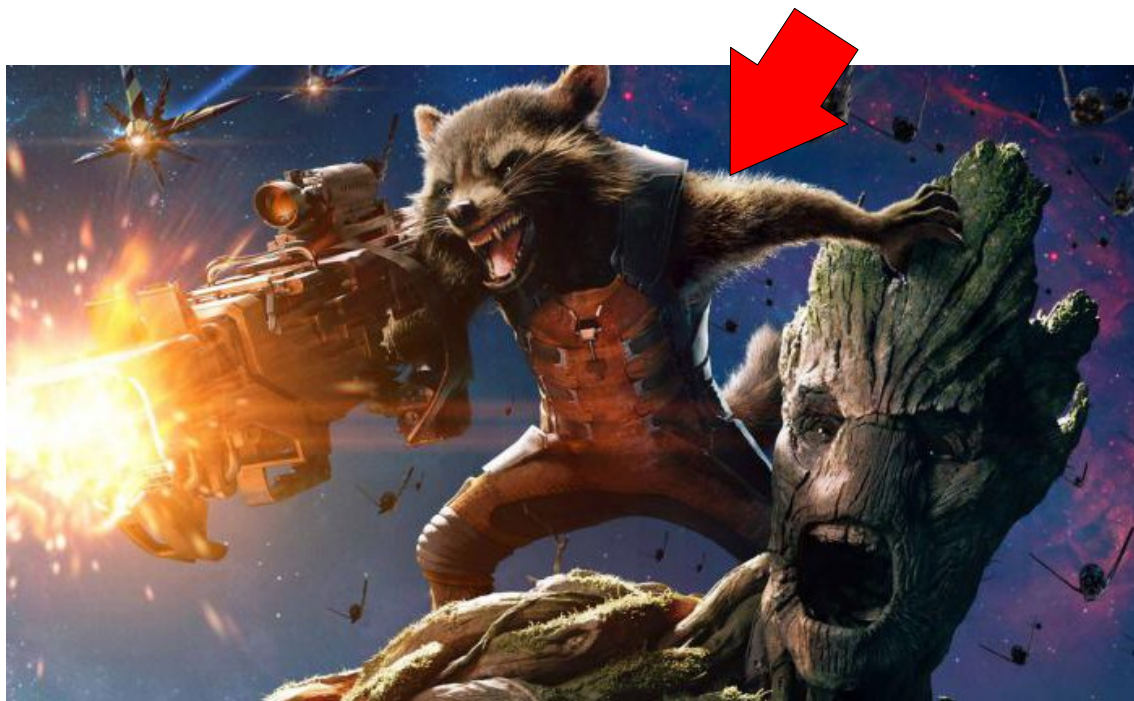
RTU и други



Philipp Slusallek RTRT on FPGA 15/63

Казахме че един от първите изцяло
Генерирани анимационни филми е
„Играта на играчките“





А защо не „Играта на пухкавите мечета“ или „Играта на желираните прасенца“... ?



Отговор

Защото са необходими много по-сложни модели за описание на сцените.

За да се визуализират комплексни сцени (съставени не само от базова геометрия, но и специални ефекти, описания на коса, козина, системи частици и др.) е необходима много по-голяма изчислителна мощ, която на онзи етап на развитие не е била налична.

Boundary Representation (B-Rep) не е достатъчен.



Основна цел на докторската дисертация

*Да се създаде модел на
отворена хибридна система
за геометрично моделиране (ОХСГМ),
базирана на
отворена хибридна (нехомогенизирана)
представяща схема (ОХПС)*



Публикации свързани с темата и други

3. А. Пенев, *"Отворени хибридни системи за геометрично моделиране"*, Дисертационен труд, ПУ, Пловдив, Август 2013
4. А. Penev, *"Computer Graphics and Geometric Modelling - a Hybrid Approach"*, International Journal of Pure and Applied Mathematics, Academic Publications, vol. 85, No. 4, 2013 [SCOPUS]
5. А. Penev, V. Vassilev, T. Petrov, *"SolidOpt - Innovative Multiple Model Software Optimization Framework"*, Second International Conference on Creativity and Innovation in Software Engineering (CISE-2009), Ravda (Nessebar), Bulgaria, 2009
6. А. Penev, D. Dimov, V. Vassilev, *"Applying Model-View-Controller in geometric modelling methodology"*, International conference "Infomatics in scientific knowledge", Varna, 2008



Публикации свързани с темата и други

6. А. Пенев, Д. Димов, *"Приложения на OpenF в обучението по Компютърна Графика"*, Международна научна конференция "Науката, образованието и времето като грижа", Смолян, 2007
7. A. Penev, D. Dimov D., D. Kralchev, *"Acceleration of structured and heterogeneous configuration of the applications"*, 36th conference of the Union of Bulgarian Mathematicians, Varna, 2007
8. A. Penev, D. Dimov, D. Kralchev, *"Functional representation in Computer graphics"*, Proceedings of the 35th Spring Conference of the Union of Bulgarian Mathematicians, Borovets, Bulgaria, 5-8 April 2006
9. A. Penev, D. Dimov, D. Kralchev, *"Open hybrid system for geometrical modeling"*, 17th International Conference on Systems for Automation of Engineering and Research, Varna, Bulgaria, 19-21 Sep, 2003

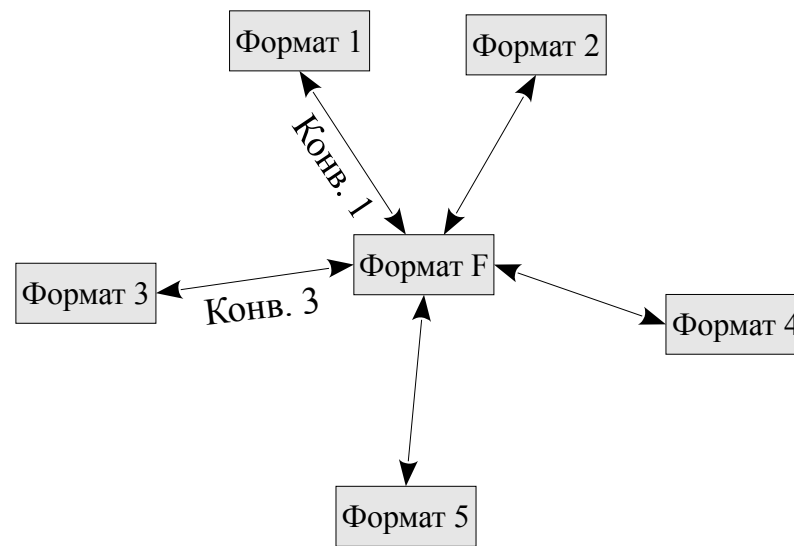
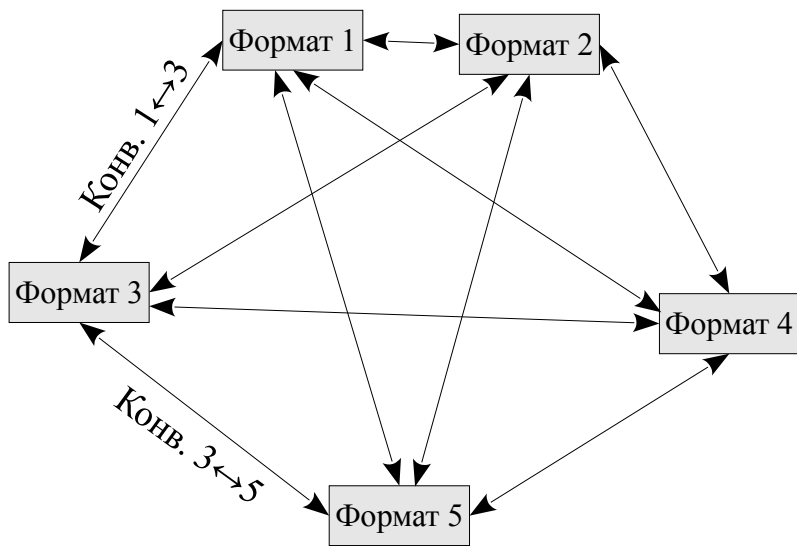


OpenF - фреймуърк реализация на идеята

- Анализирани са и са сравнени възможностите, предимствата и недостатъците на (фиксираните) хибридните представящи схеми за геометрично моделиране
- Предложен е модел (концепция) на отворена хибридна система за геометрично моделиране (ОХСГМ), базирана на отворена хибридна представяща схема (ОХПС)
- Предложена е архитектура на софтуерна рамка за създаване на ОХСГМ, базирани на ОХПС
- Разработен е прототип на софтуерна рамка (framework) OpenF за създаване на ОХСГМ, базирани на ОХПС
- Разработени са приложения-примери за използване на софтуерната рамка OpenF. Изследван е интереса и приложимостта на OpenF за целите на обучението



Хибридность относно Представянето

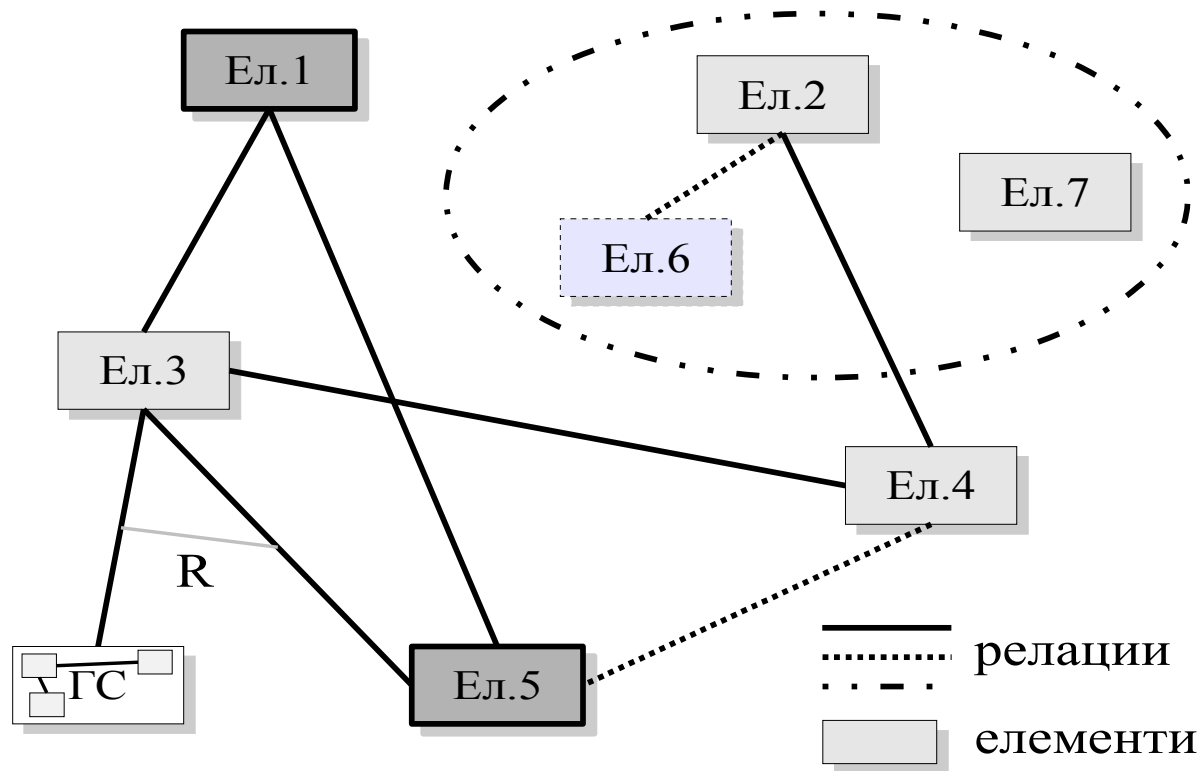


Хибридна система:

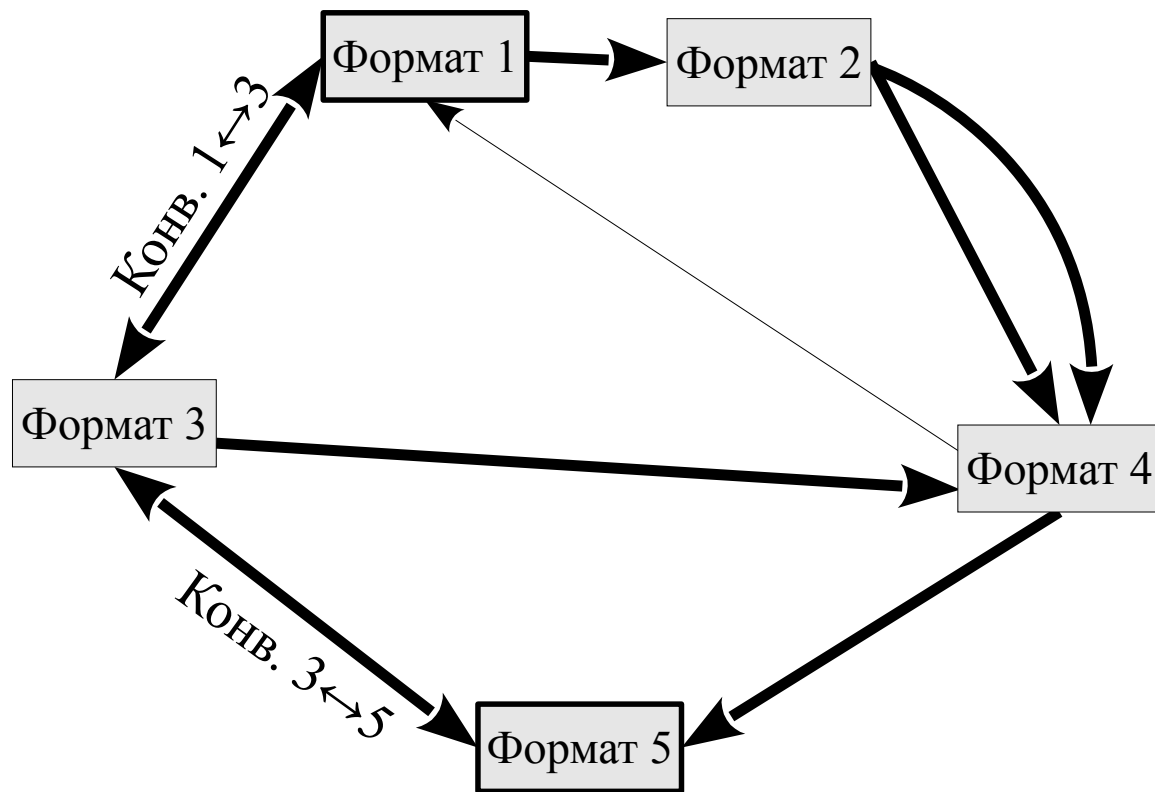
а) с пълно множество конвертори; б) с междинен формат



Отворен хибриден граф на сцената



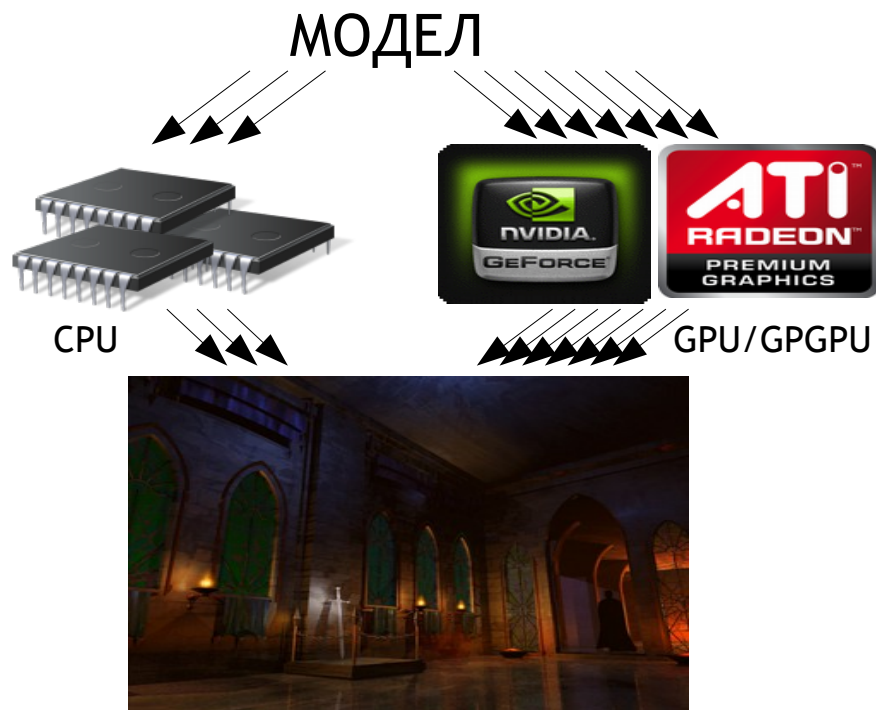
Търсене на пътища за конвертиране



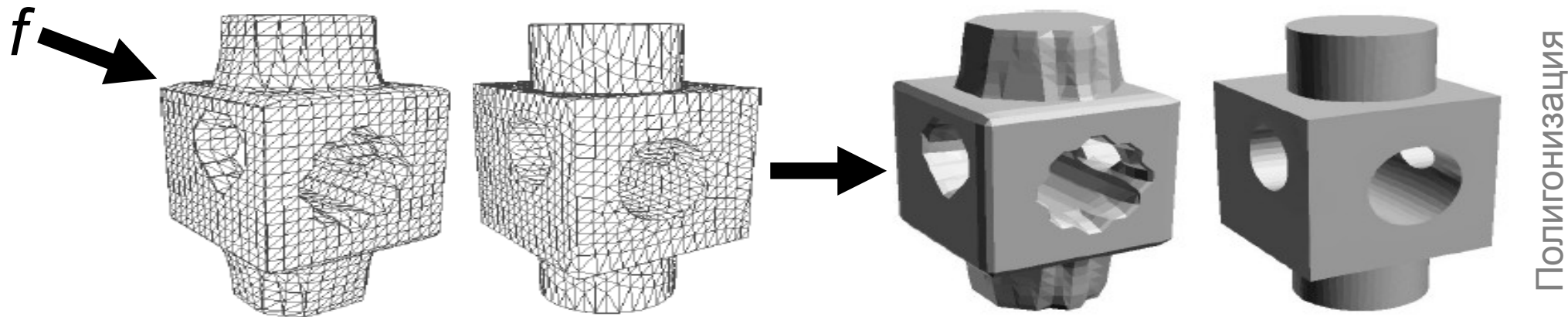
Хибридность относено Реализацията

(използване на различни видове хардуер за реализиране на оп.)

1. CPU / GPU / GPGPU
2. OpenGL;
3. OpenCL;
4. OpenRT;
5. и др.



Хибридность относно Визуализацията



f

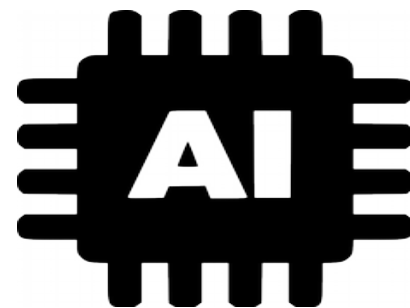
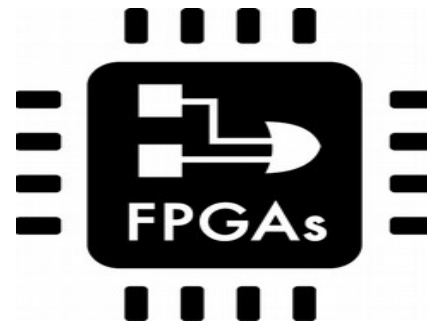
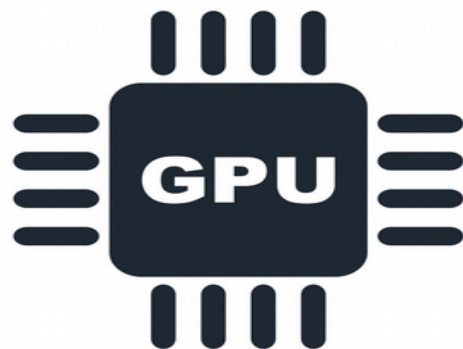
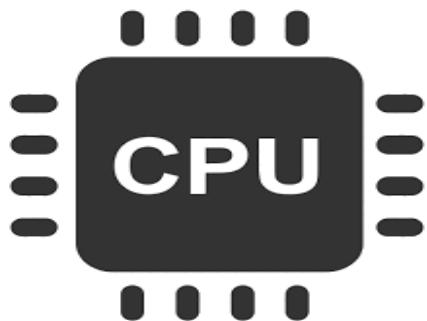


Ray tracing



Хабилитация

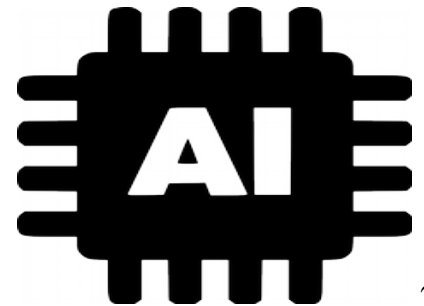
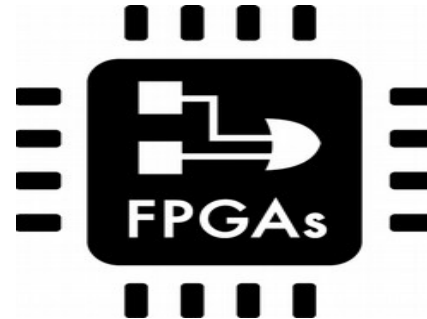
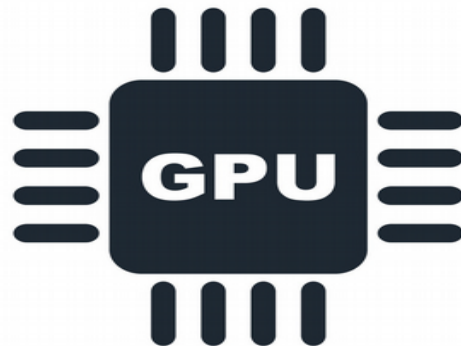
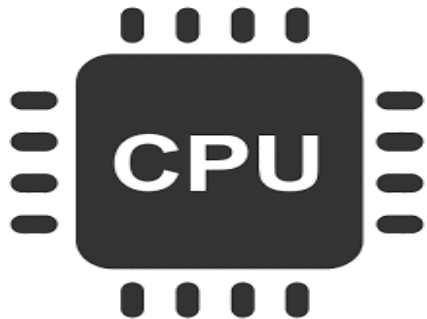
Компютърна графика (2013-2018)





Хабилитация

Компютърна графика (2013-2018)



Развитие на GPGPU (2006-до сега)



Intel Phi



Nvidia Tesla

Учебно помагало

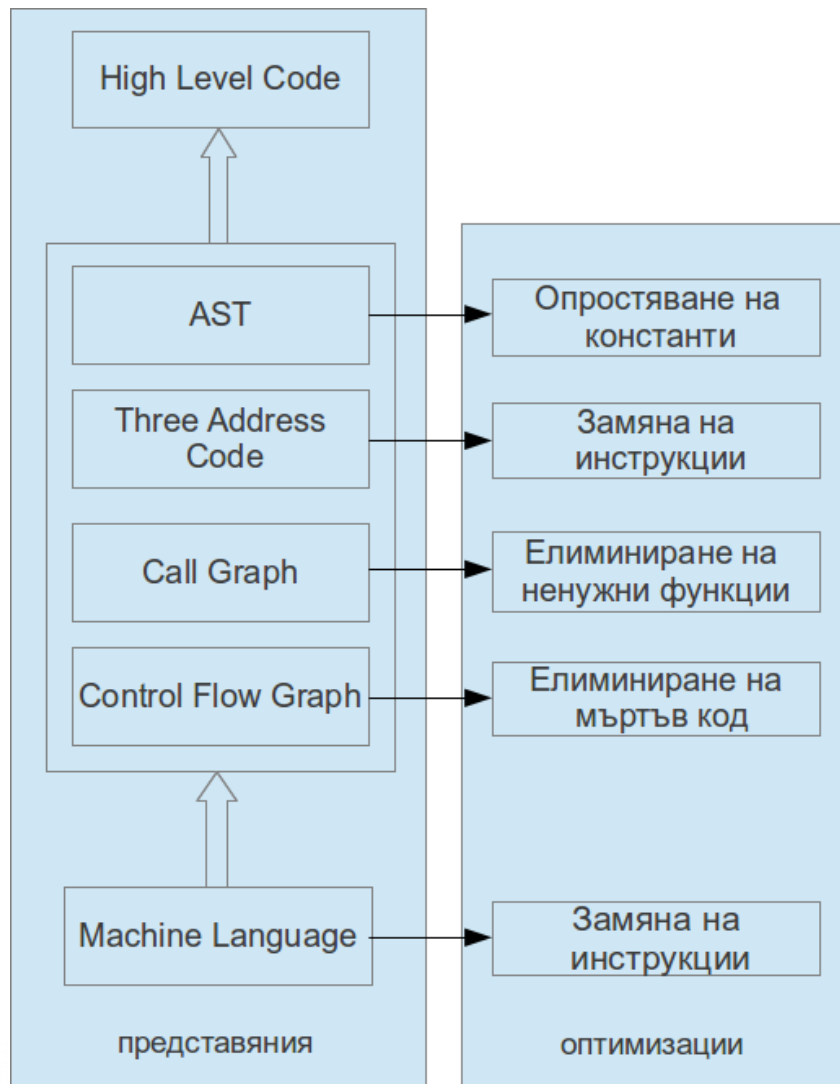
- Дефинира функционална спецификация на проект със средна сложност, който позволява на обучаваните да се сблъскат с различни реализационни аспекти на изучаваната теория
- Може лесно да се използва и адаптира при практическото обучение по „Компютърна графика“ в повече от една специалност
- Ръководството не е обвързано с нито един език за програмиране, операционна система, хардуерна платформа и други (дори и избора на графичната библиотека не фиксиран). Това дава голяма свобода на обучаваните в избор на подходящи средства за реализиране на проекта
- Ръководството (и последователността от задачи в него) е организирано така, че почти във всеки един момент студентите имат работещо приложение, което те могат да предадат след изтичане на крайния срок за предаване на проекта. Започва се от проста диалогова (двумерна) система, прост модел, прост алгоритъм на визуализация; постепенно тази система се развива с усложняване на тези елементи за постигане на изискваните функционалности; накрая се достига до сложна (тримерна) система за геометрично моделиране, сложен модел (граф на сцената), различни алгоритми на визуализация, сложен графичен интерфейс за взаимодействие с потребителя



Проекти SolidOpt и SolidV

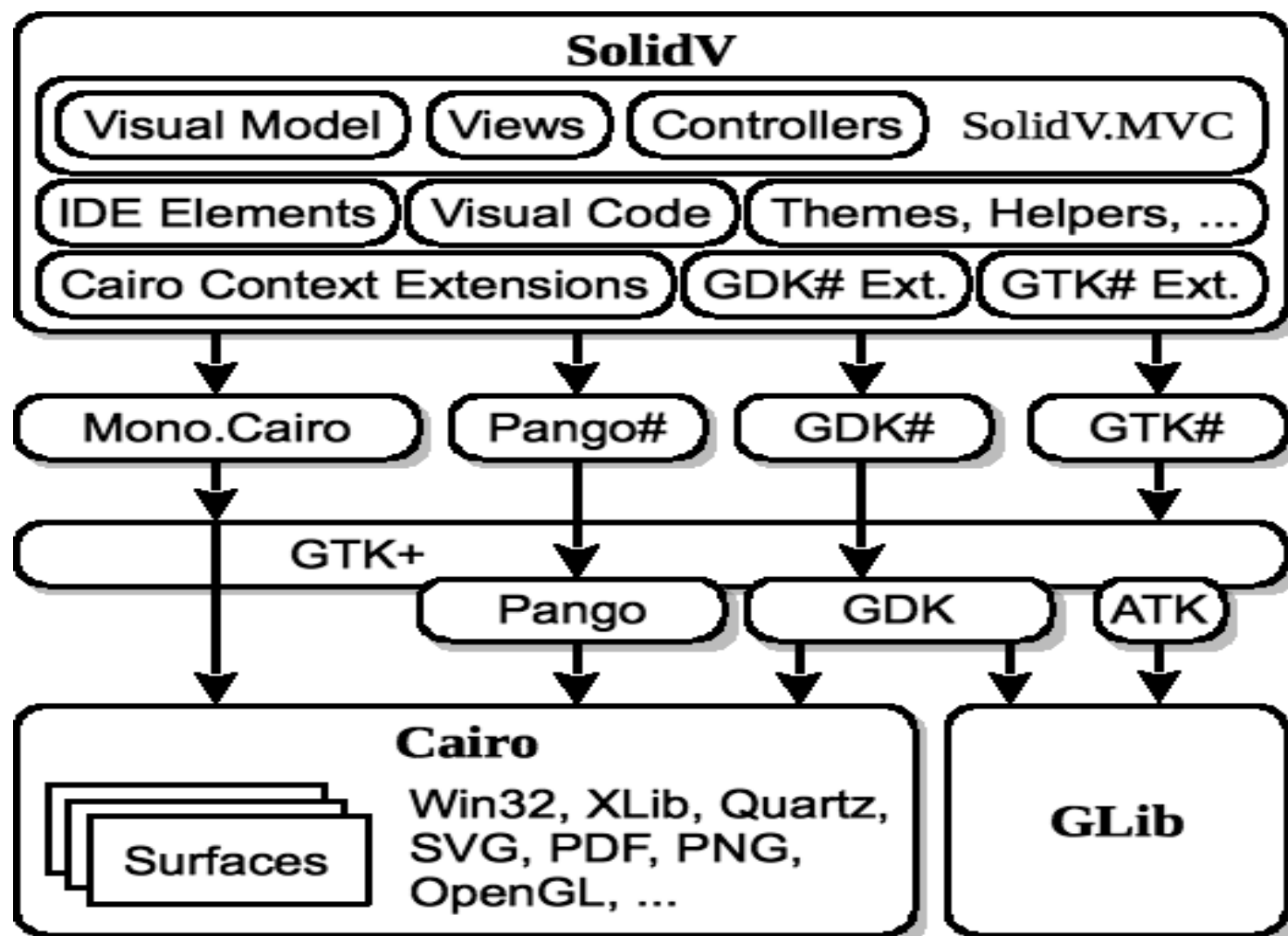
- Доразвит е моделът на система за оптимизация на софтуер, базиран на подхода „софтуера-като-модел“ - включващ йерархия от модели и различни трансформации над тях
- SolidOpt предоставя средства за декомпиляция, анализ и трансформации над .NET IL програми, като по този начин се допълват оптимизациите от компилатора с Domain и Application специфични оптимизации
- Доразвита е концепцията за еволюционен подход за оптимизиране на софтуерните системи в зависимост от поведението ѝ в околната среда
- SolidV надграждане на SolidOpt с визуални модели с цел лесно създаване на домейн-специфични Визуални езици и Среди за програмиране

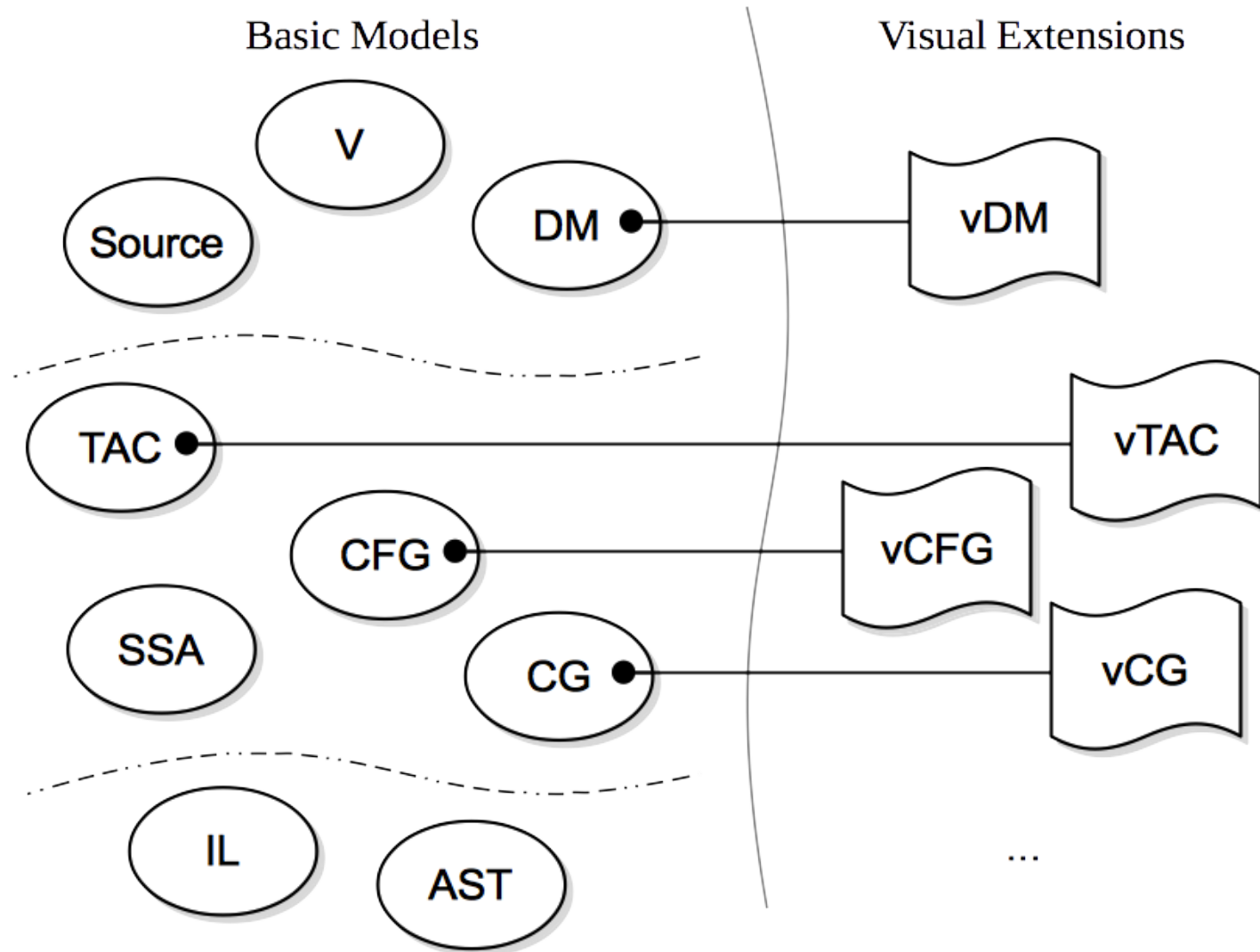




- SolidOpt работи на много нива на абстракция
- Използва се подход на декомпиляция на изпълнимия код готовата система до различни междинни представяния
- На всяко ниво на абстракция се прилагат общи и домейн-специфични оптимизации за постигане на желаните резултати







SolidReflector

SolidReflector
File Plugins Help

Assembly

- TestProject
 - TestProject.exe
 - <Module>
 - TestClassA
 - MethodA
 - MethodB
 - MethodC
 - MethodD
 - MethodE
 - Main

CFG Visualizer

CFG Text

CFG Visualizer

Block Name: 0
IL_0000: ldc.i4.4
IL_0001: stloc.0
IL_0002: ldloc.0
IL_0003: ldc.i4.3
IL_0004: bne.un IL_0019

Block Name: 1
IL_0009: ldstr "a == 3"
IL_000e: call System.Void
IL_0013: ldstr "a == 3"
IL_0018: ret

Block Name: 2
IL_0019: ldloc.0
IL_001a: ldc.i4.4
IL_001b: bne.un IL_0030

Block Name: 3
IL_0030: ldstr "a is unk"
IL_0035: call System.Void
IL_003a: ldstr "a is unk"
IL_003f: ret

Block Name: 4
IL_0020: ldstr "a == 4"
IL_0025: call System.Void
IL_002a: ldstr "a == 4"
IL_002f: ret

Call Graph Visualizer

CG Text

CG Visualizer

MethodA

WriteLine

WriteLine

WriteLine

WriteLine

WriteLine

WriteLine

Simulation Visualizer

Simulate

New CFG:

Method output:

TAC Visualizer

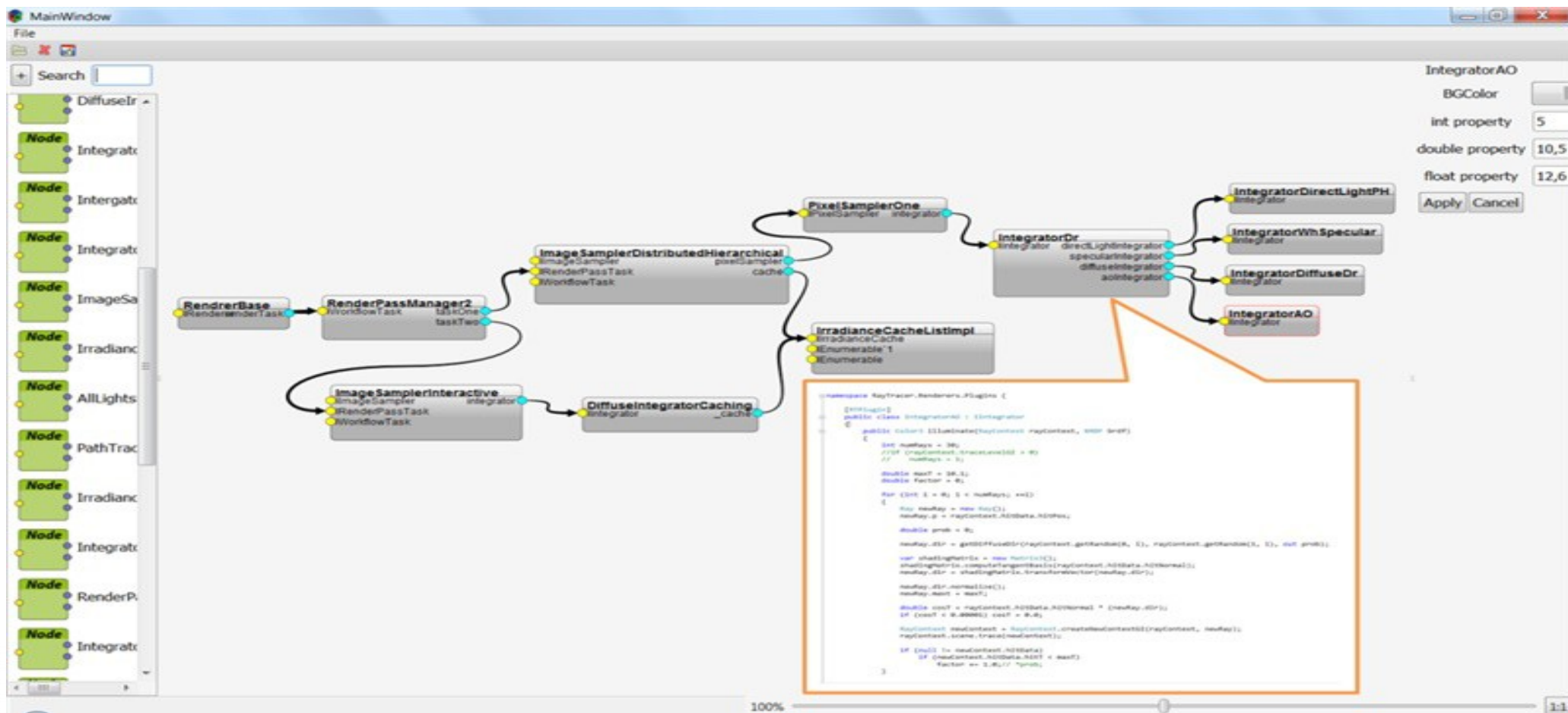
System.String TestProject.TestClassA::MethodA() {
L0: V_0 = 4
L1: T_0 = V_0 == 3
L2: iffalse T_0 goto L6
L3: pushparam "a == 3"
L4: call System.Void System.Console::WriteLine(System.String)
L5: return "a == 3"
L6: T_1 = V_0 == 4
L7: iffalse T_1 goto L11
L8: pushparam "a == 4"
L9: call System.Void System.Console::WriteLine(System.String)
L10: return "a == 4"
L11: pushparam "a is unknown"
L12: call System.Void System.Console::WriteLine(System.String)
L13: return "a is unknown"
}

IL Visualizer

IL Text

.method public hidebysig static String MethodA () cil managed
{
.locals init (int32 V_0)
IL_0000: ldc.i4.4
IL_0001: stloc.0
IL_0002: ldloc.0
IL_0003: ldc.i4.3
IL_0004: bne.un IL_0019
IL_0009: ldstr "a == 3"
IL_000e: call System.Void System.Console::WriteLine(System.String)
IL_0013: ldstr "a == 3"
IL_0018: ret
IL_0019: ldloc.0
IL_001a: ldc.i4.4
IL_001b: bne.un IL_0030

RayTracer

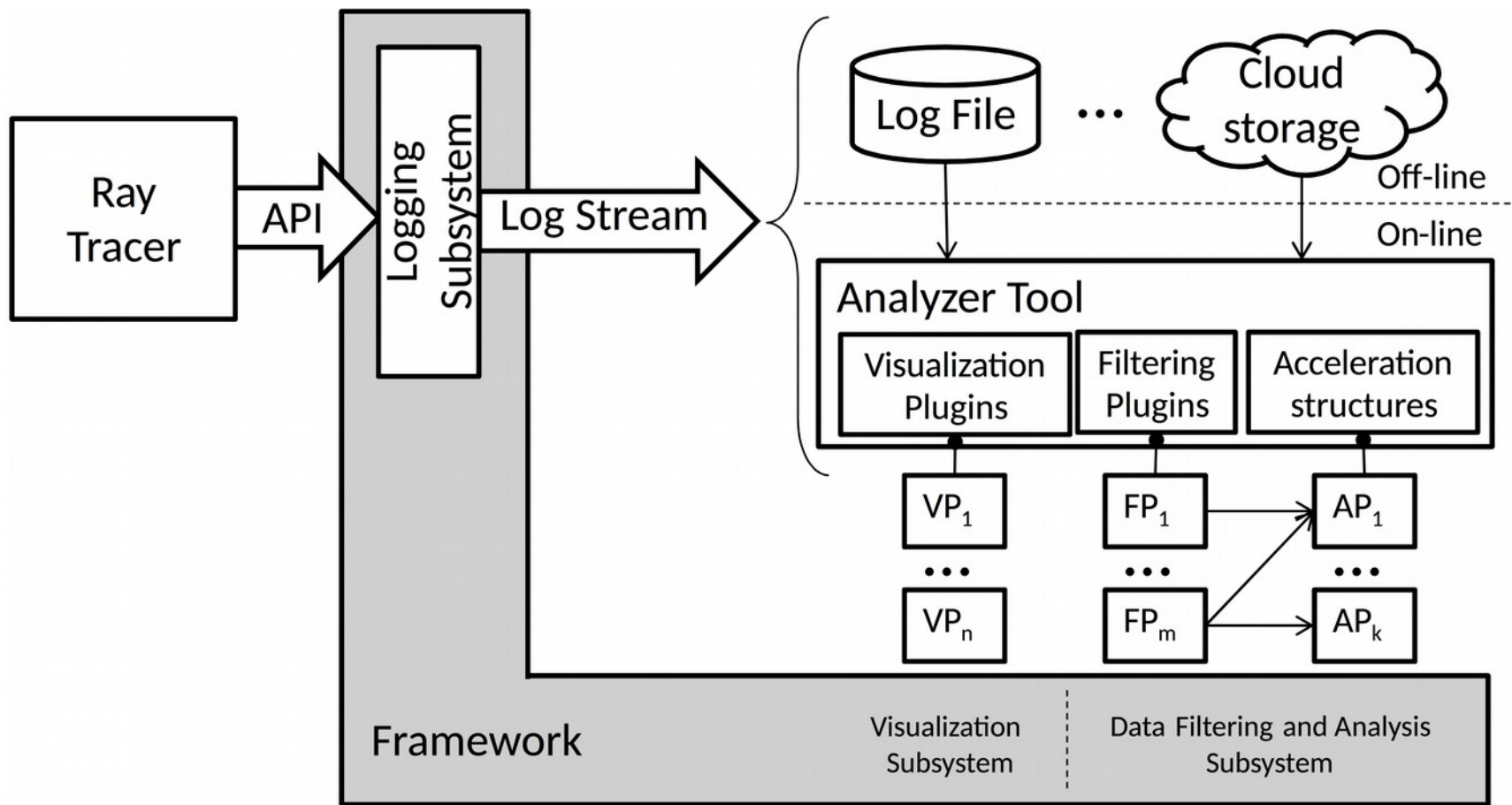


Визуален Анализ на Ray tracing алгоритми

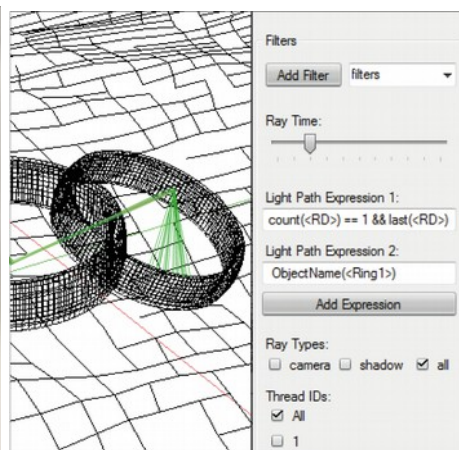
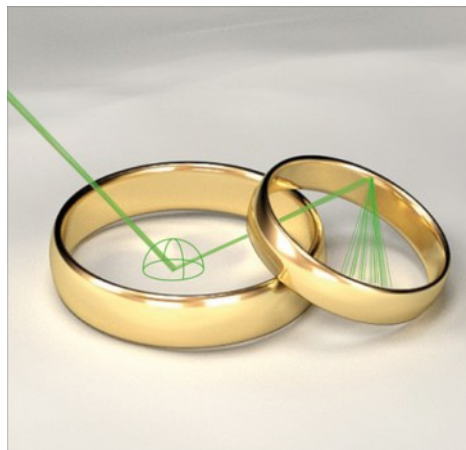
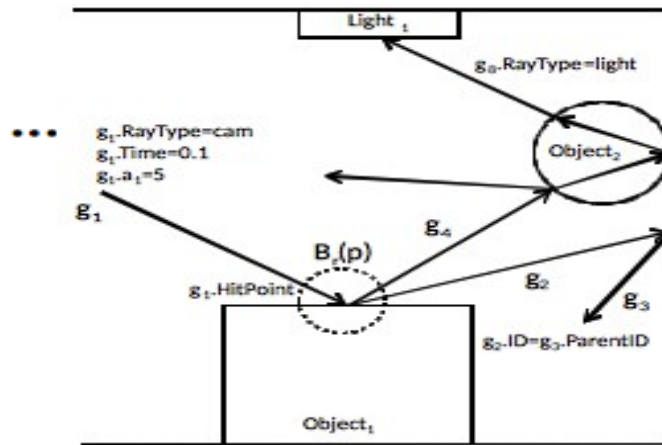
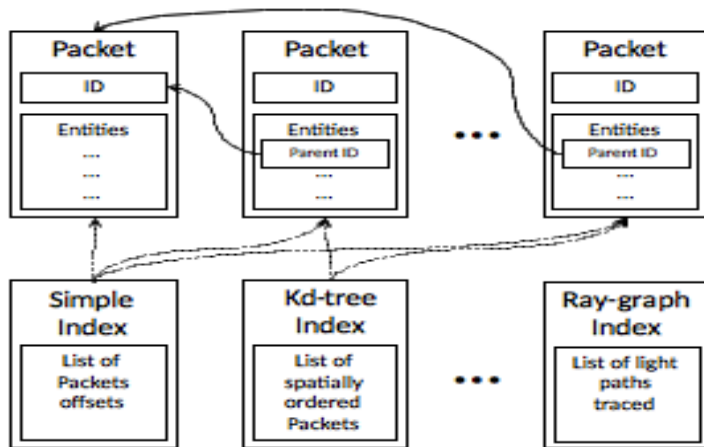
- Предложена е концепция за визуален динамичен анализ на алгоритми за визуализация, базирани на Ray tracing парадигмата
- Домейн-специфичния и визуален подход дава предимство на концепцията в сравнение с класическия подход за дебъгиране на алгоритми
- Широкия набор от филтри (част от които специфични за областта - Light Path Expression филтри, пространствени филтри, времеви филтри, филтри на многонишковото изпълнение, диференциални филтри и др.) дава възможност за лесно ориентиране за евентуалните проблеми (или специфики на реализацията, ако не става въпрос за откриване на грешки, а за разбиране на алгоритъма) въпреки специфичния за Ray tracing алгоритмите голям обем от данни



Визуален Анализ на Ray tracing алгоритми



Визуален Анализ на Ray tracing алгоритми

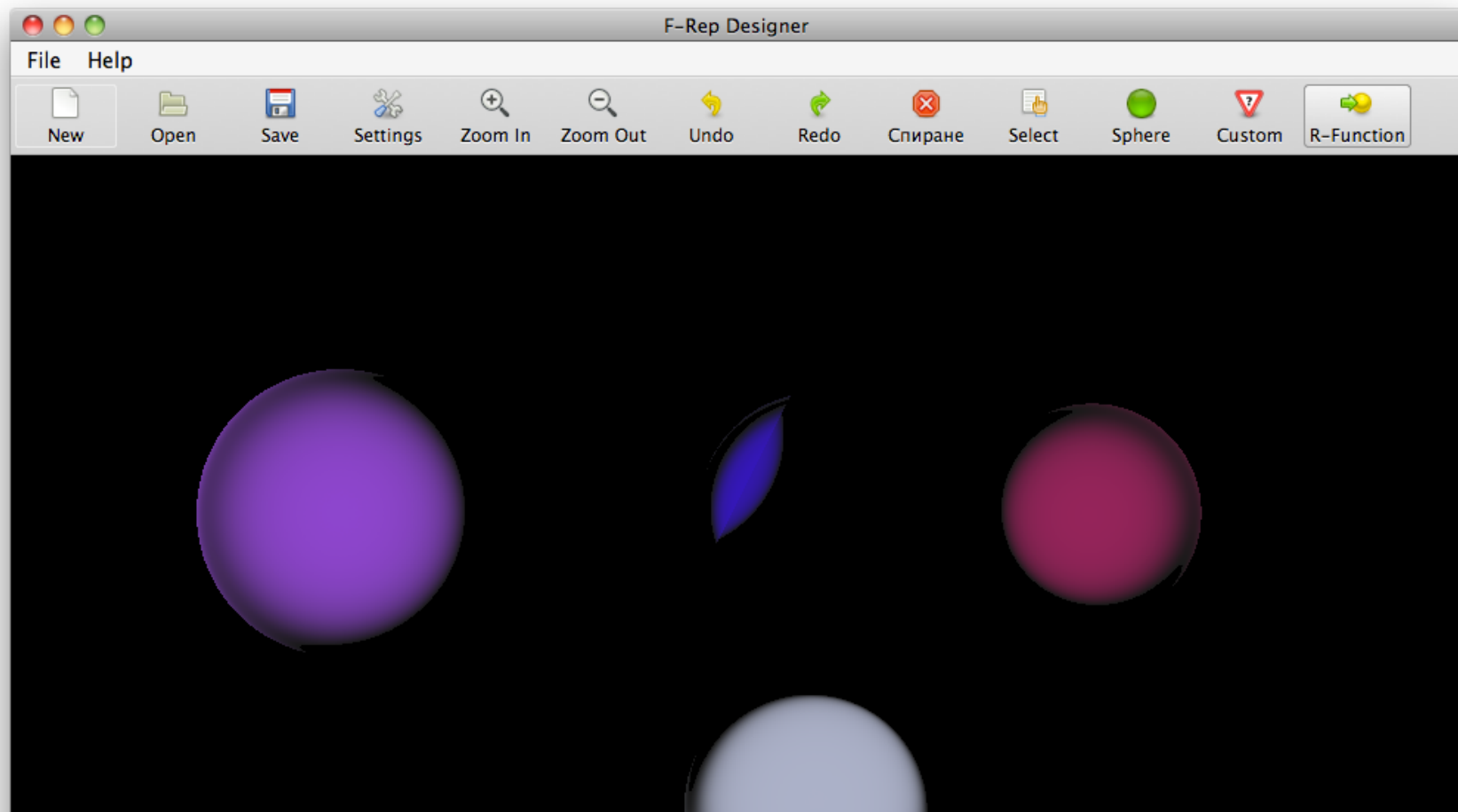


F-Rep Designer

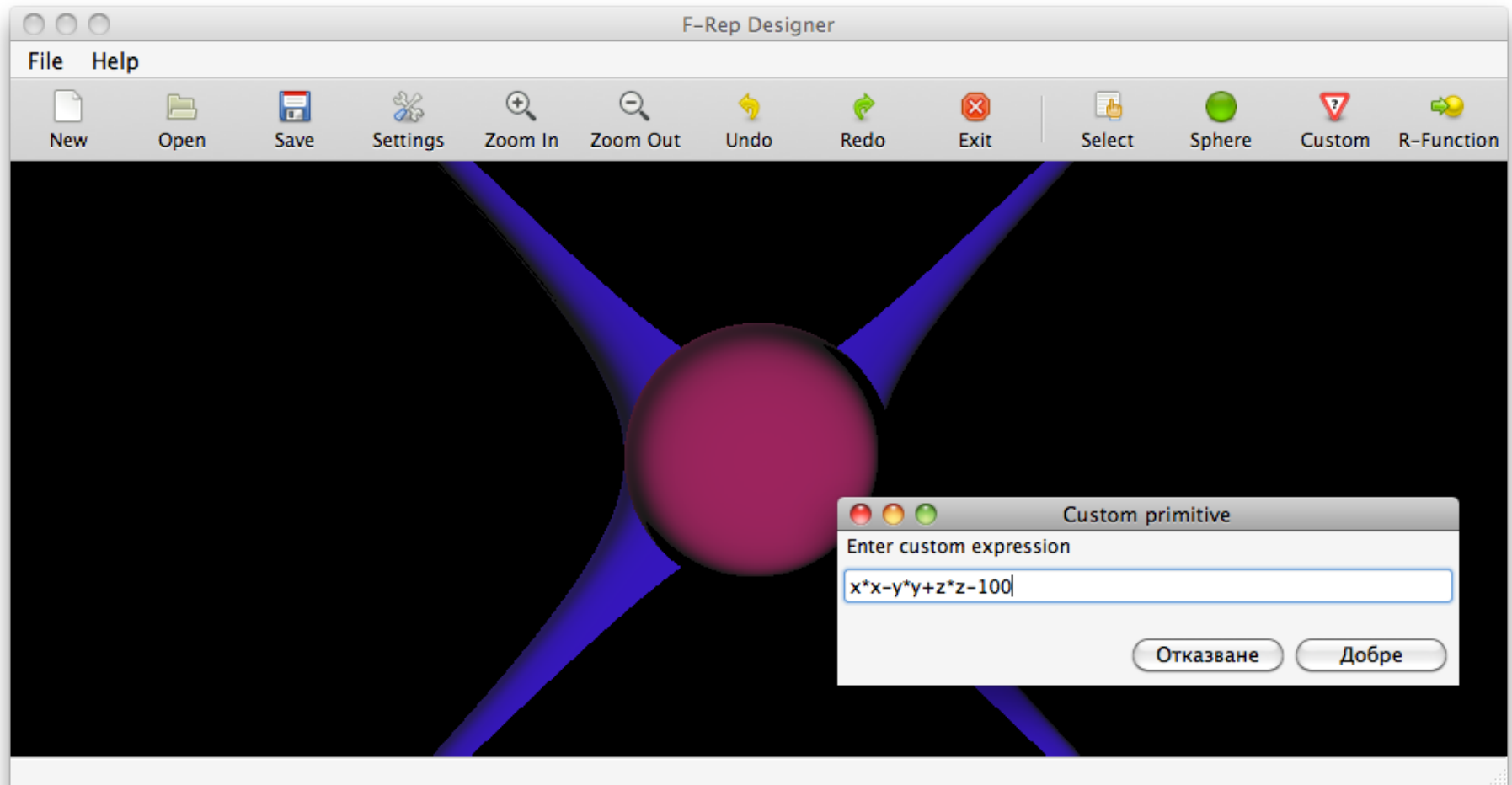
- Предложена е и е реализирана концепция за система за тримерно моделиране изцяло базирана на F-Rep представяща схема
- Използван е иновативен подход (за подобен род системи) при който се комбинират модела (F-Rep, геометрията на моделираната сцена), функциите за изчисляване на нормалните вектори, модела на осветяване и алгоритъма за визуализация (Ray Tracing) и те се компилират (в C# IL или в OpenCL) до изпълним за CPU/GPGPU код. Този код се изпълнява на съответния изпълнител, което генерира крайното изображение



F-Rep Designer



F-Rep Designer



Проект Clad

- Показан е подход за Автоматично диференциране (AD) на C++ програми, при който AD се извършва от допълнителен модул в процеса на компилация, извършващ трансформация на избрани функции и методи
- Предложен е подход за „разтоварване“ на изчисления на специализиран графичен хардуер (GPGPU)
- С различни примери е показана приложимостта на прототипната реализация (системата Clad) в Компютърната графика, Физиката на високите енергии и др.
- Системата се развива главно от студенти по Google Summer of Code с менторска организация CERN SFT

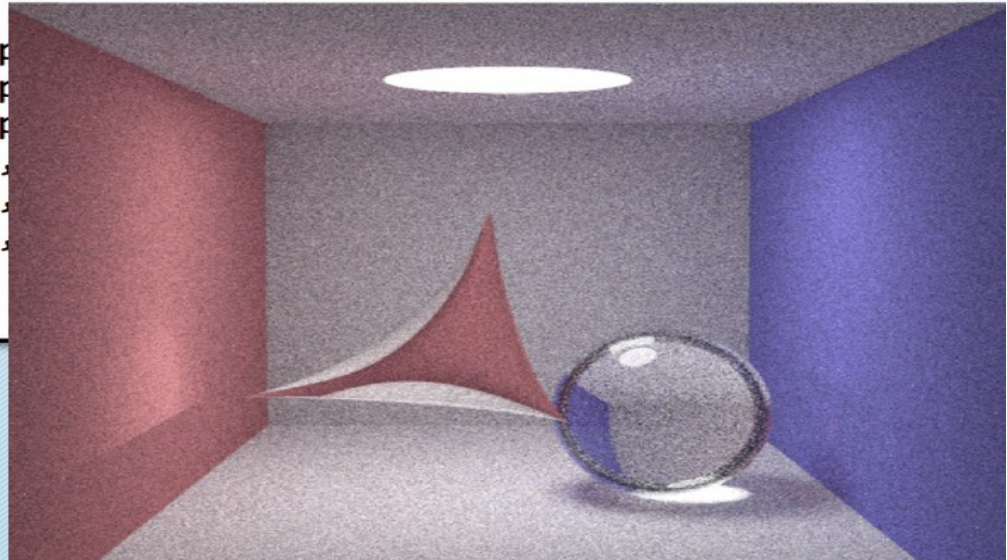


clad in Computer Graphics

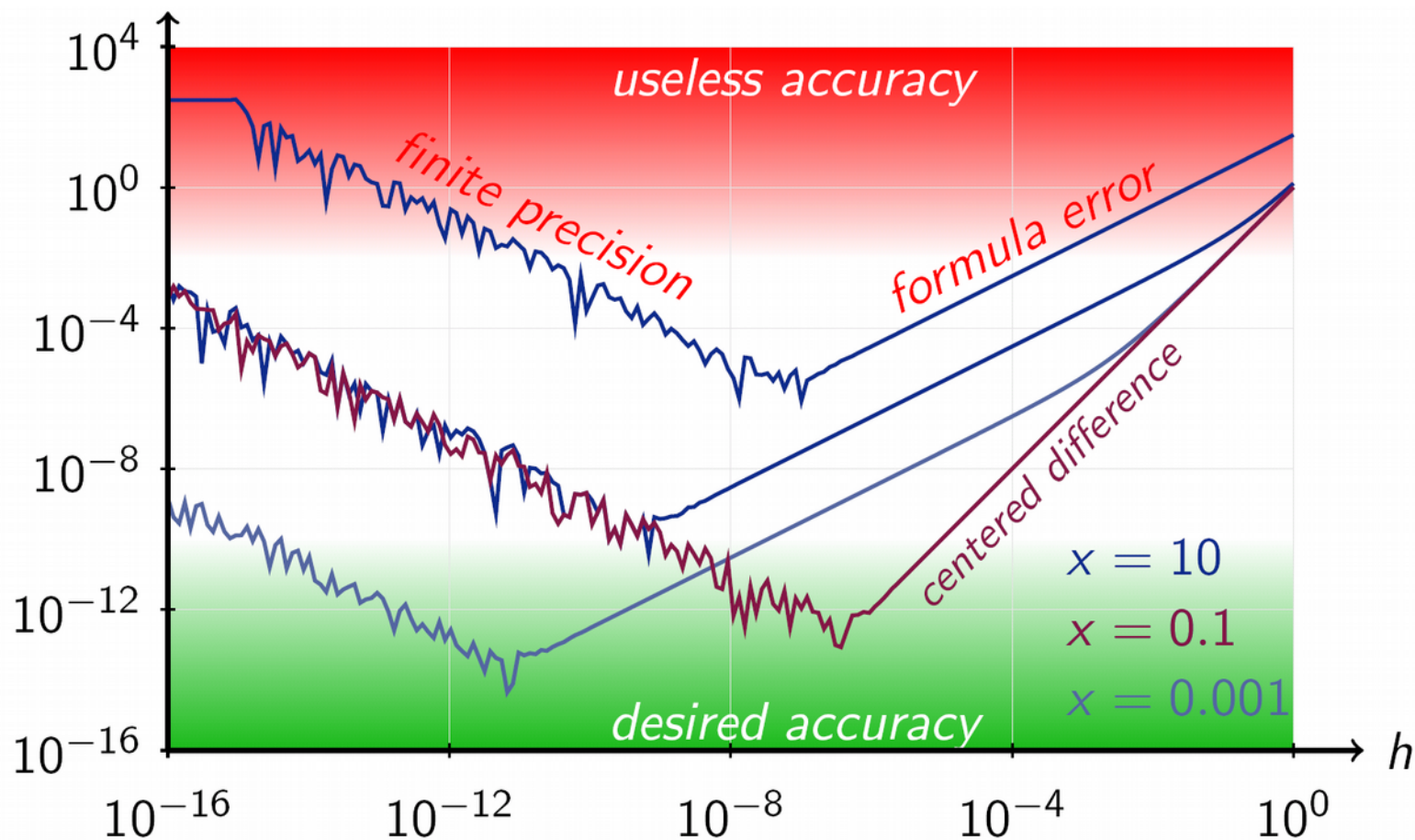
15

```
// SmallPT.cpp
float hyperbolic_solid_func(float x, float y, float z, const Vec &p, float r) {
#define sin_a 0.965925826289068
#define cos_a 0.258819045102521
    return pow((x-p.x)*cos_a+(z-p.z)*sin_a, 2./3.) + pow(y-p.y, 2./3.) + pow((x-
p.x)*-sin_a+(z-p.z)*cos_a, 2./3.) - pow(r, 2./3.);
}
```

```
Vec normal(const Vec &pt) const override {
    auto hs_func_dx = clad::differentiate(hyperbolic_solid_func, 1);
    auto hs_func_dy = clad::differentiate(hyperbolic_solid_func, 2);
    auto hs_func_dz = clad::differentiate(hyperbolic_solid_func, 3);
    float Nx = hs_func_dx.execute(pt.x, pt.y, pt.z);
    float Ny = hs_func_dy.execute(pt.x, pt.y, pt.z);
    float Nz = hs_func_dz.execute(pt.x, pt.y, pt.z);
    return Vec(Nx, Ny, Nz).norm();
}
```



Проект Clad - проблеми при численото диф.



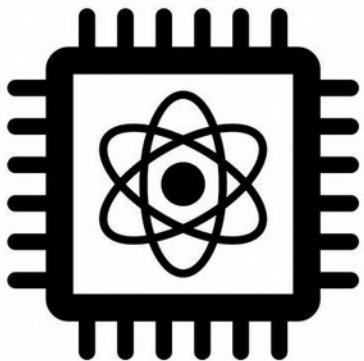
Публикации и други

10. A. Golev, A. Penev, Stefanova K., Hristova S., *"Using GPU to Speed up Calculation of Some Approximate Methods for Fractional Differential Equations"*, International Journal of Pure and Applied Mathematics, Academic Publications, vol. 119, No. 3, 2018
11. V. Vassilev, M. Vassilev, A. Penev, Moneta L., Ilieva V., *"Clad - Automatic Differentiation Using Clang and LLVM"*, Journal of Physics: Conference Series, IOP Publishing, vol. 608, article 01205512, 2015 [SCOPUS]
12. V. Vassilev, A. Penev, M. Vassilev, *"SolidOpt - A Multi-Model Software Optimization Framework"*, International Journal of Computer Science Issues, vol. 12, No.2, 2015
13. H. Lesev, A. Penev, *"A Framework for Visual Dynamic Analysis of Ray Tracing Algorithms"*, Cybernetics and Information Technologies, vol. 14, No.2, 2014 [SCOPUS] с два SCOPUS цитата до момента (единият с IF)
14. A. Penev, *"F-Rep Designer"*, Proceedings of the International Conference "From DeLC to Velspace", Plovdiv, 26-28 March 2014



Бъдеще

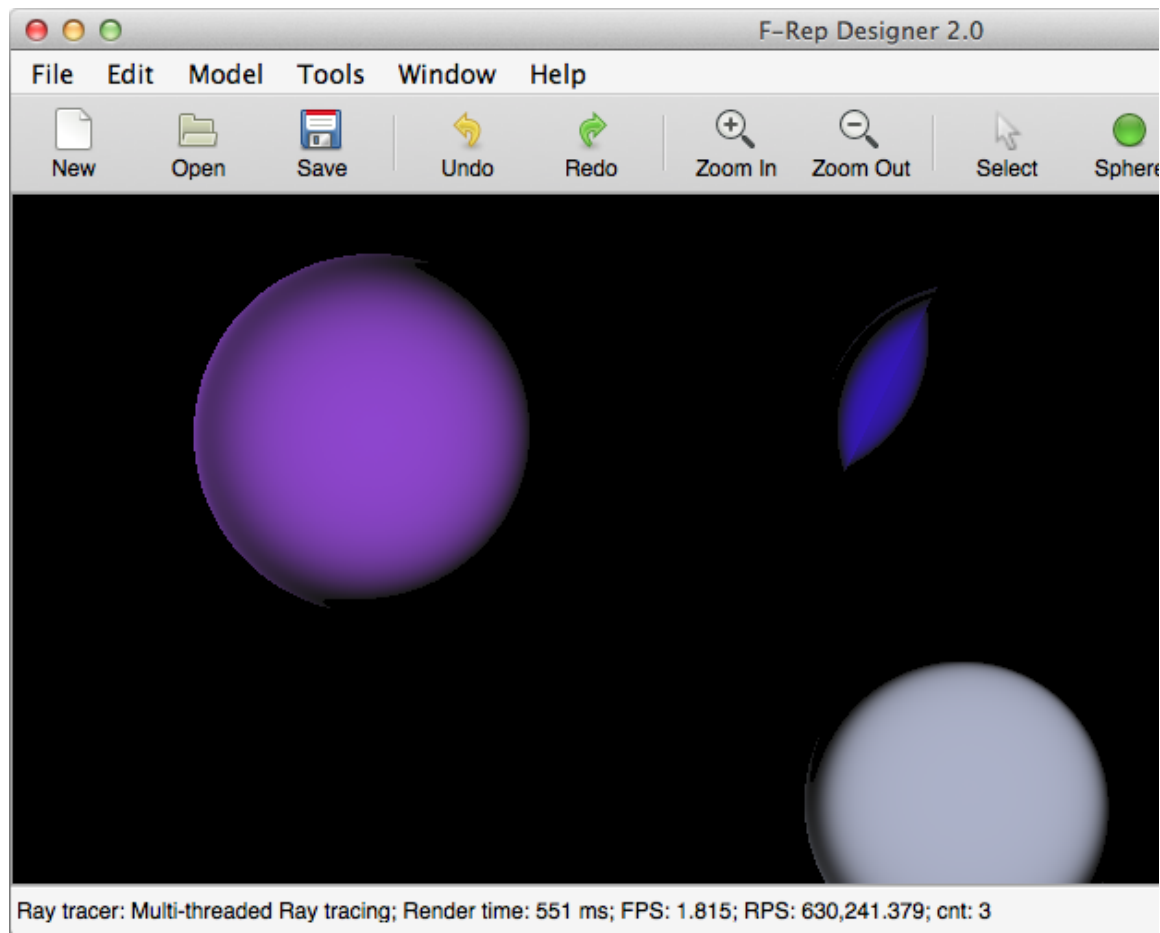
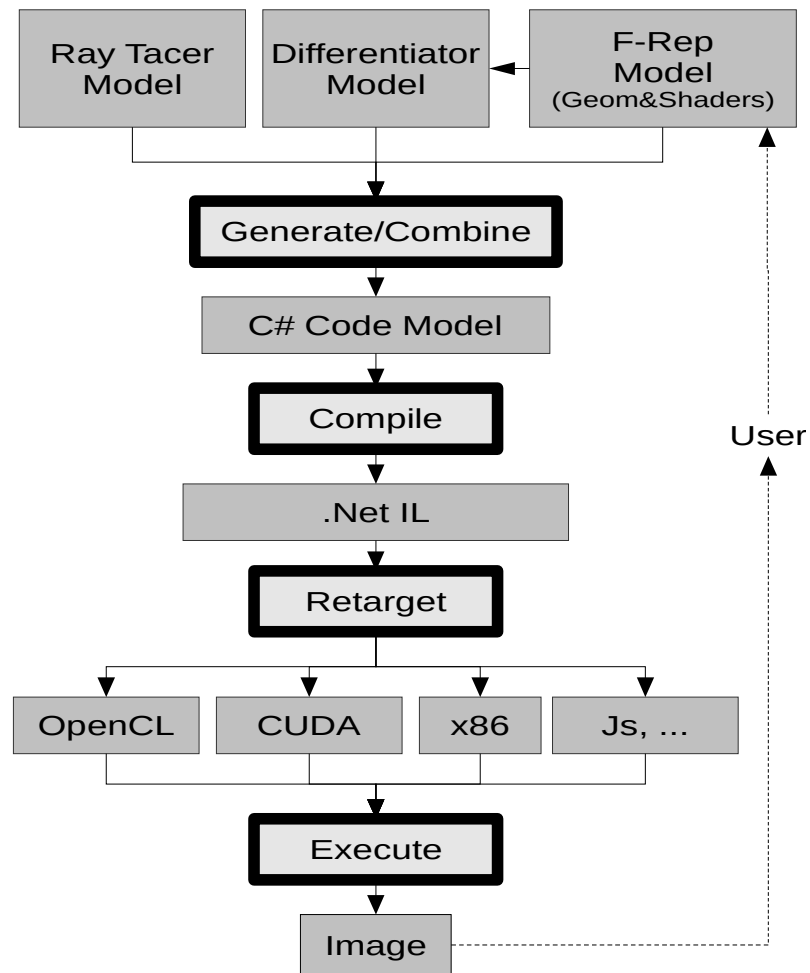
или с какво смятам да се занимавам занапред



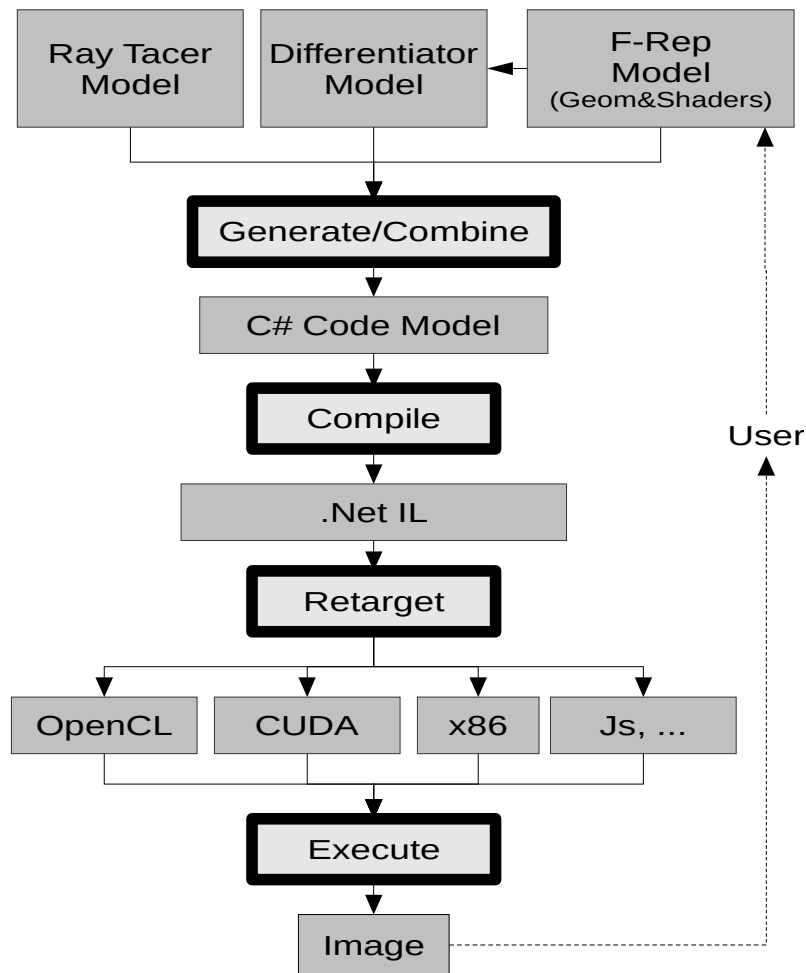
...



F-Rep Designer 2.0



F-Rep Designer 2.0



Подхода в системата е различен от повечето системи за геометрично моделиране по няколко причини:

- Използва се F-Rep, а не B-Rep
- Визуализацията е изцяло Ray tracing базирана
- Визуализацията действа по скоро като компилятор на Domain специфичен визуален език за прорамиране, а не като интерпретатор на данните в сцената

F-Rep Designer 2.0 - предимства

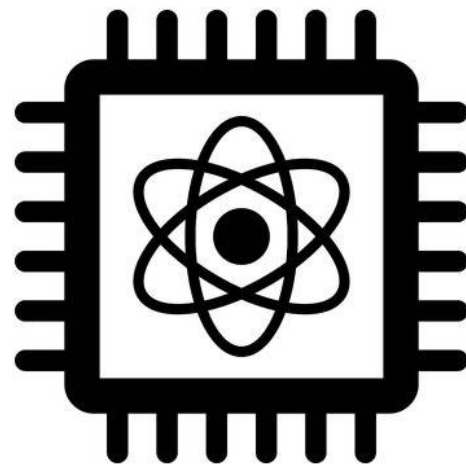
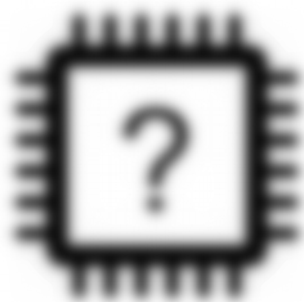
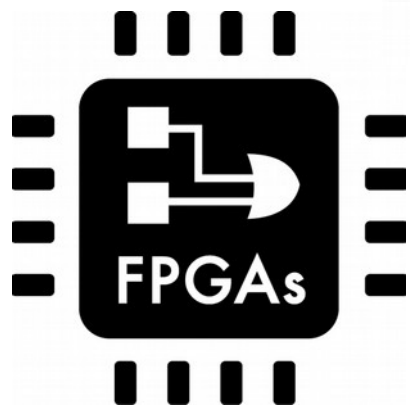
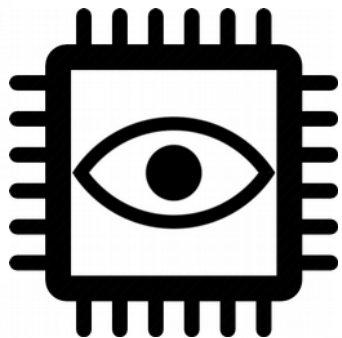
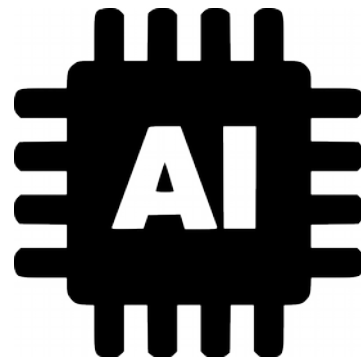
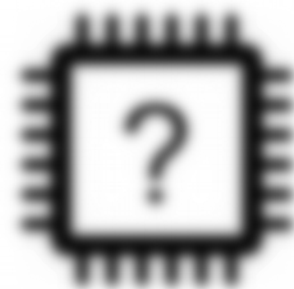
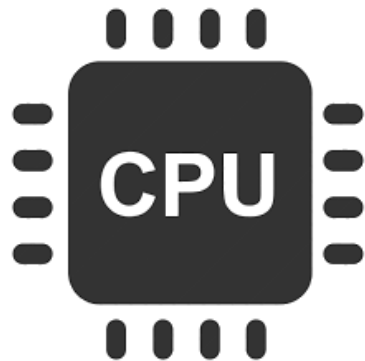
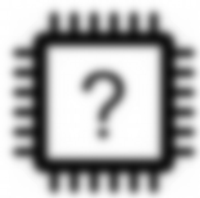
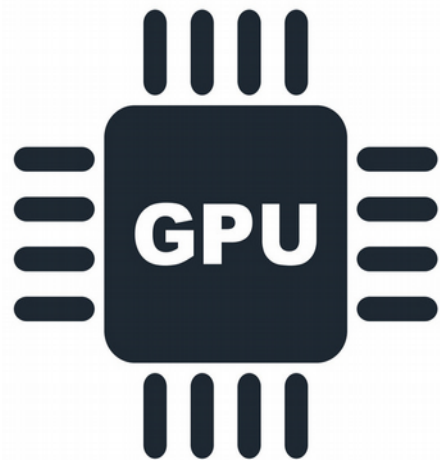
- Представящата схема F-Rep е много по-мощна от B-Rep
- Mesh-подобни структури не се използват в нито един етап от работата на системата
- Визуализацията е изцяло Ray tracing базирана
- Модела на сцената (геометрията), Модела на осветяване (shaders и други), Пресмятането на нормалните вектори (чрез AD), Самия Ray tracing алгоритъм се компилират заедно. Това дава възможност за оптимизация (и Domain-специфична оптимизация) не само на алгоритмите, но и на самите данни (геометрия, шейдъри, ...)
- Подхода позволява лесен ретаргетинг към различни платформи за изпълнение (включително хетерогенно)



Публикации и други

15. A. Penev, "*F-Rep Designer 2.0 - Everything is a Code*", International Journal of Computer Science Issues, vol. 15, No.4, 2018
16. M. Vassileva, V. Vassileva, A. Penev, P. Vassileva, "FREP-Cloud - A Cross-Platform Mobile Application for FEP Events", World of Conference Journal, vol. 25, No.4, 2019
17. A. Penev, "Quantum Computer Graphics", International Journal of Computer Science Issues, vol. 15, No.2, 2019
18. ...

„Изчислители“



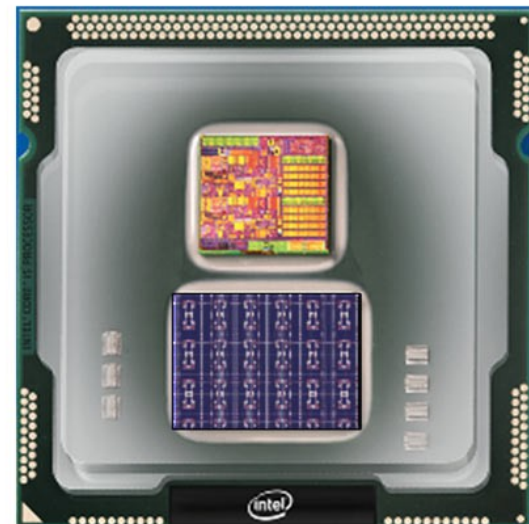
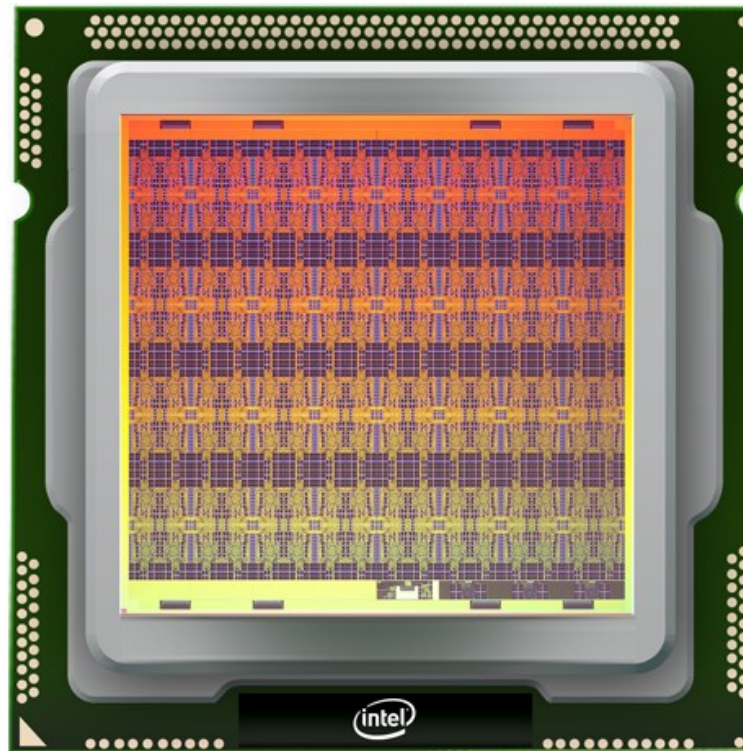
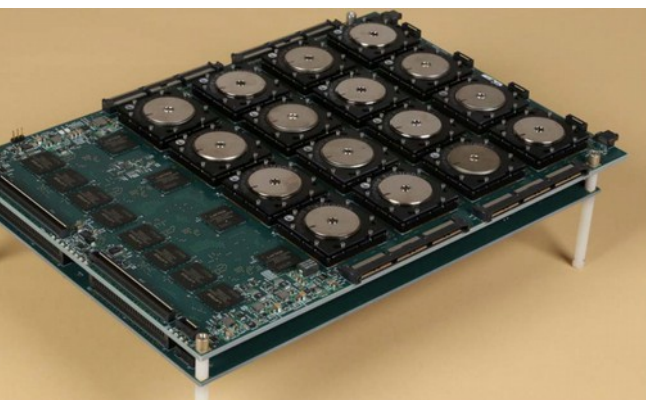
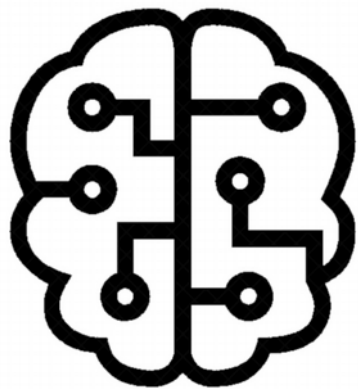
Denoising (с използване на CNN)



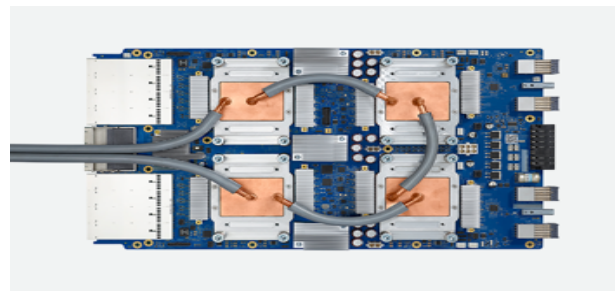
През 2017 nVidia представиха своя деноисер работещ в реално време, базиран на невронни мрежи



Нервоморфни (NPU) системи

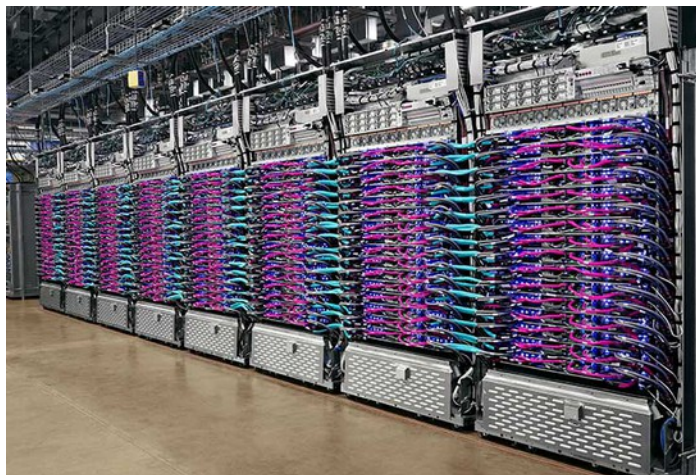


Доставчиците на Cloud услуги предлагат лесен достъп до специализирания хардуер

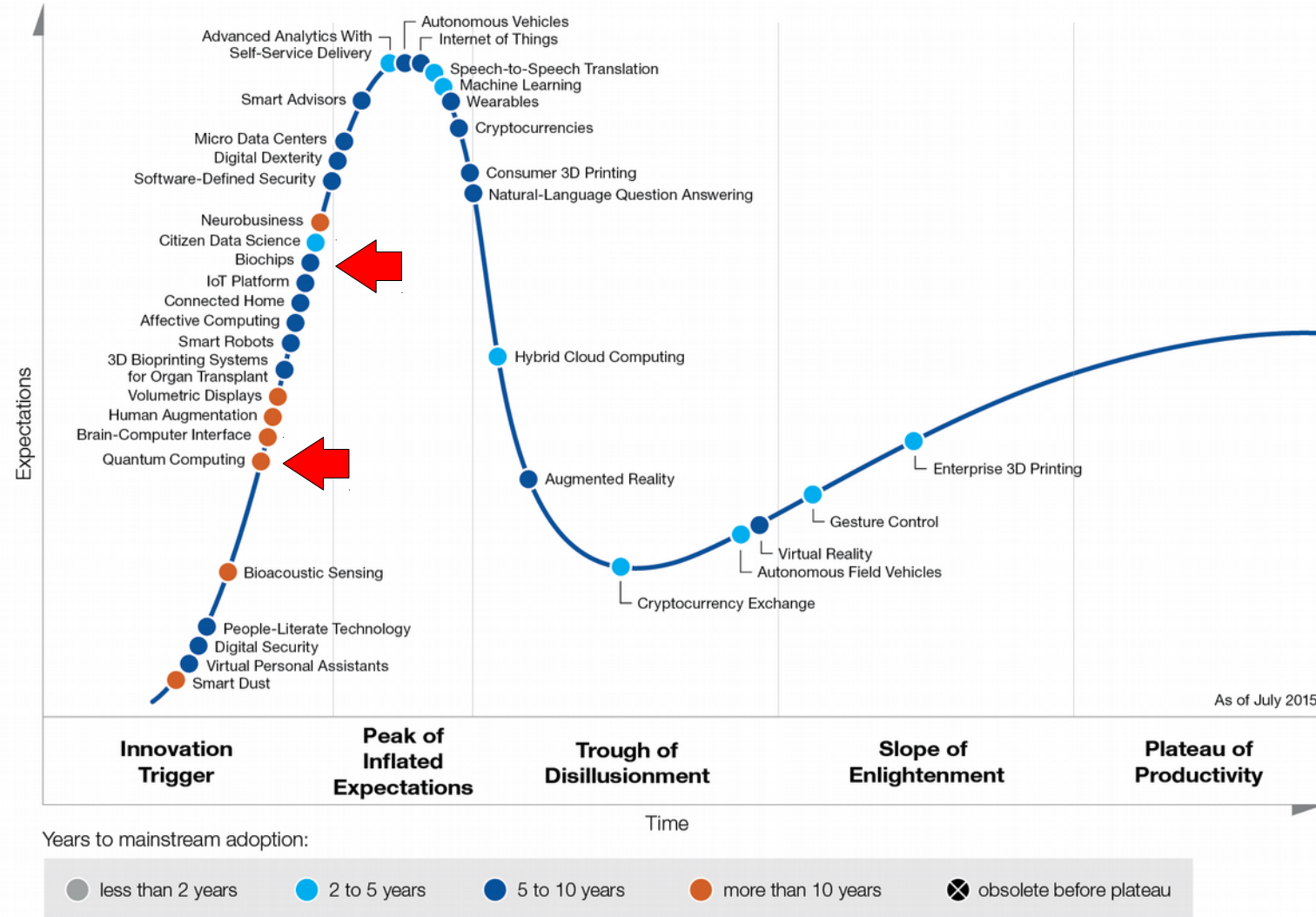


- TPU Card to replace a disk
- Up to 4 cards / server

TPU Card & Package



Emerging Technology Hype Cycle



The three known types of quantum computing and their applications, generality, and computational power.

Quantum Annealer

The quantum annealer is least powerful and most restrictive form of quantum computers. It is the easiest to build, yet can only perform one specific function. The consensus of the scientific community is that a quantum annealer has no known advantages over conventional computing.



A very specialized form of quantum computing with unproven advantages over other specialized forms of conventional computing.

DIFFICULTY LEVEL

APPLICATION
Optimization Problems

GENERABILITY
Restrictive

COMPUTATIONAL POWER
Same as traditional computers

Analog Quantum

The analog quantum computer will be able to simulate complex quantum interactions that are intractable for any known conventional machines, or combinations of these machines. It is conjectured that the analog quantum computer will contain somewhere between 50 to 100 qubits.



The most likely form of quantum computing that will first show true quantum speedup over conventional computing. This could happen within the next five years.

DIFFICULTY LEVEL

APPLICATIONS
Material Chemistry
Optimization Problems
Sampling
Quantum Dynamics

GENERABILITY
Partial

COMPUTATIONAL POWER
High

Universal Quantum

The universal quantum computer is the most powerful, the most general, and the hardest to build, posing a number of difficult technical challenges. Current estimates indicate that this machine will comprise more than 100,000 physical qubits.



The true grand challenge in quantum computing. It offers the potential to be exponentially faster than traditional computers for a number of important applications for science and businesses.

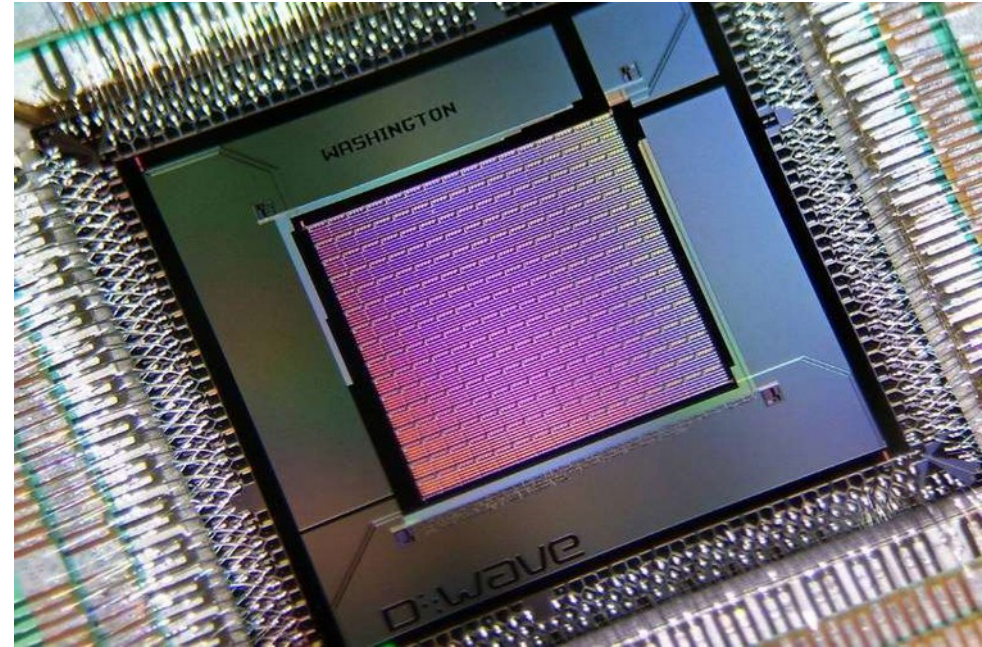
DIFFICULTY LEVEL

APPLICATIONS
Secure computing
Machine Learning
Cryptography
Material Chemistry
Optimization Problems
Sampling
Quantum Dynamics
Searching

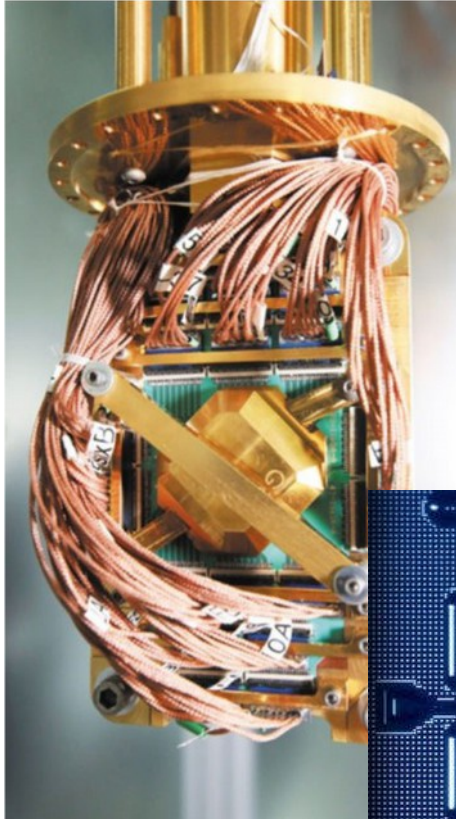
GENERABILITY
Complete with known speed up

COMPUTATIONAL POWER
Very High

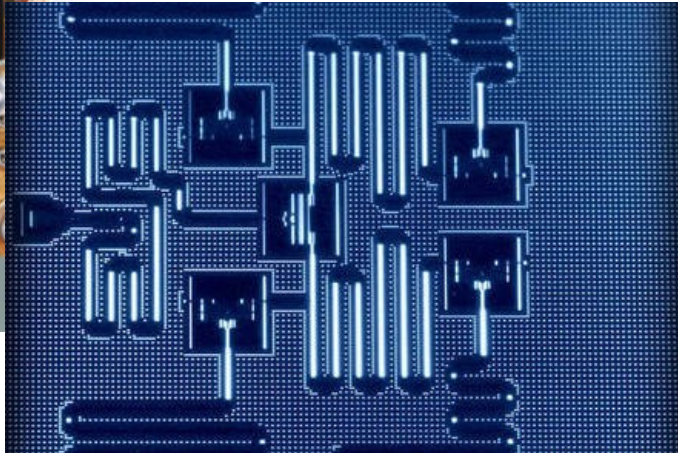
Квантови Компютри



Квантови Компютри



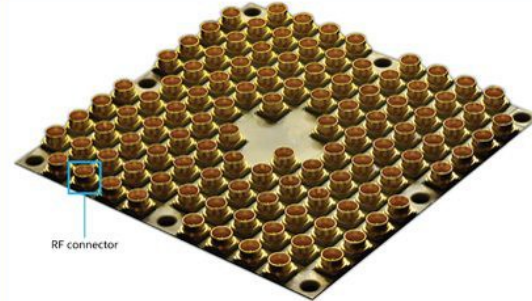
5 Qbit-ов процесор на IBM



INTEL'S 49-QUBIT PROCESSOR

During his keynote at CES 2018 in January, Intel CEO Brian Krzanich unveiled our 49-qubit superconducting quantum test chip, code-named "Tangle Lake." The 3-inch by 3-inch chip and its package is now in the hands of Intel's quantum research partner QuTech in the Netherlands for testing at low temperatures. Quantum computing is heralded for its potential to tackle problems that today's conventional computers can't handle. Scientists and industries are looking to quantum computing to speed advancements in areas like chemistry or drug development, financial modeling, and even climate forecasting.

TOP



WORTH ITS WEIGHT IN GOLD

There are 108 radio frequency (RF) connectors on Tangle Lake that carry microwave signals into the chip to operate the quantum bits (qubits). They are made of gold, which is excellent for anti-corrosion and signal transmission.

49 Qbit-ов процесор на Intel



- ва Компютърна Графика
ви компютри и алгоритми
ни езици за програмиране и компилатори
изация на софтуер в силно хетерогенни среди
н специфични визуализации и визуално дебъгиране
- 
- The image shows a quantum circuit diagram with five horizontal lines representing qubits, labeled $q[0]$ through $q[4]$ on the left, each starting with a $|0\rangle$ state. The circuit includes several gates:
 - On $q[0]$: H, T, H, T, S gates.
 - On $q[1]$: H, T, H, T, S gates.
 - On $q[2]$: H, T, H, T, S gates.
 - On $q[3]$: H, T, H, T, S gates.
 - On $q[4]$: H, T, H, T, S gates.
 - Control lines for CNOT gates:
 - A dashed line from $q[0]$ to a CNOT on $q[4]$.
 - A solid line from $q[1]$ to a CNOT on $q[4]$.
 - A solid line from $q[2]$ to a CNOT on $q[4]$.
 - A solid line from $q[3]$ to a CNOT on $q[4]$. In the bottom right corner, there is a small rectangular box containing a cartoon illustration of a white cat with a blue vertical stripe on its chest, sitting and looking forward.

Благодарности

на всички, които ме подкрепиха и още...



Благодарности

- На семейството ми, което ми дава смисъл и подкрепа
- На всички мои студенти, които ми дават основанията и задължението да се развивам
- На моите дипломанти, с които взаимно се възпламеняваме и ми напомнят, че и аз съм бил такъв
- На моите съавтори, които ми показват, че има и други хора с моите интереси и ме допълват интелектуално
- На моите ментори и учители, които са ми показвали грешките и са ми дали основата, на която да стъпя здраво
- На ръководствата на ФМИ, подпомагащи всички нас по трънливия път на процедурите
- ... и на всички останали, които ме карат да искам да бъда още по-добър във всичко, което правя



Компютърна Графика в Хетерогенна Среда

Благодаря за вниманието!

Въпроси?

apenev@uni-plovdiv.bg

