Google internship

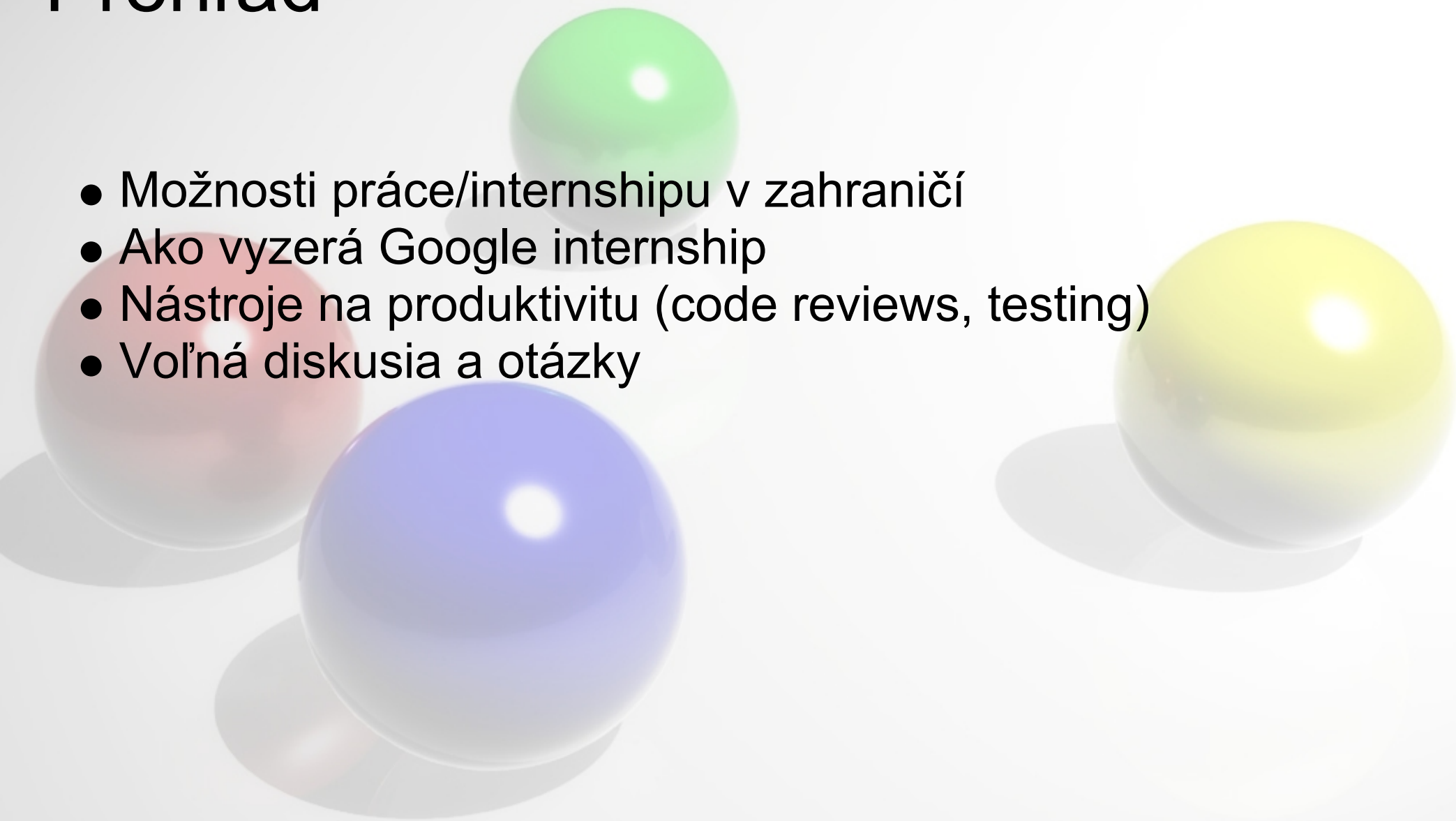
Peter Perešíni
3.11.2009

Google

CAINHOB

Prehľad

- Možnosti práce/internshipu v zahraničí
- Ako vyzerá Google internship
- Nástroje na produktivitu (code reviews, testing)
- Voľná diskusia a otázky



Google

CAINHOB

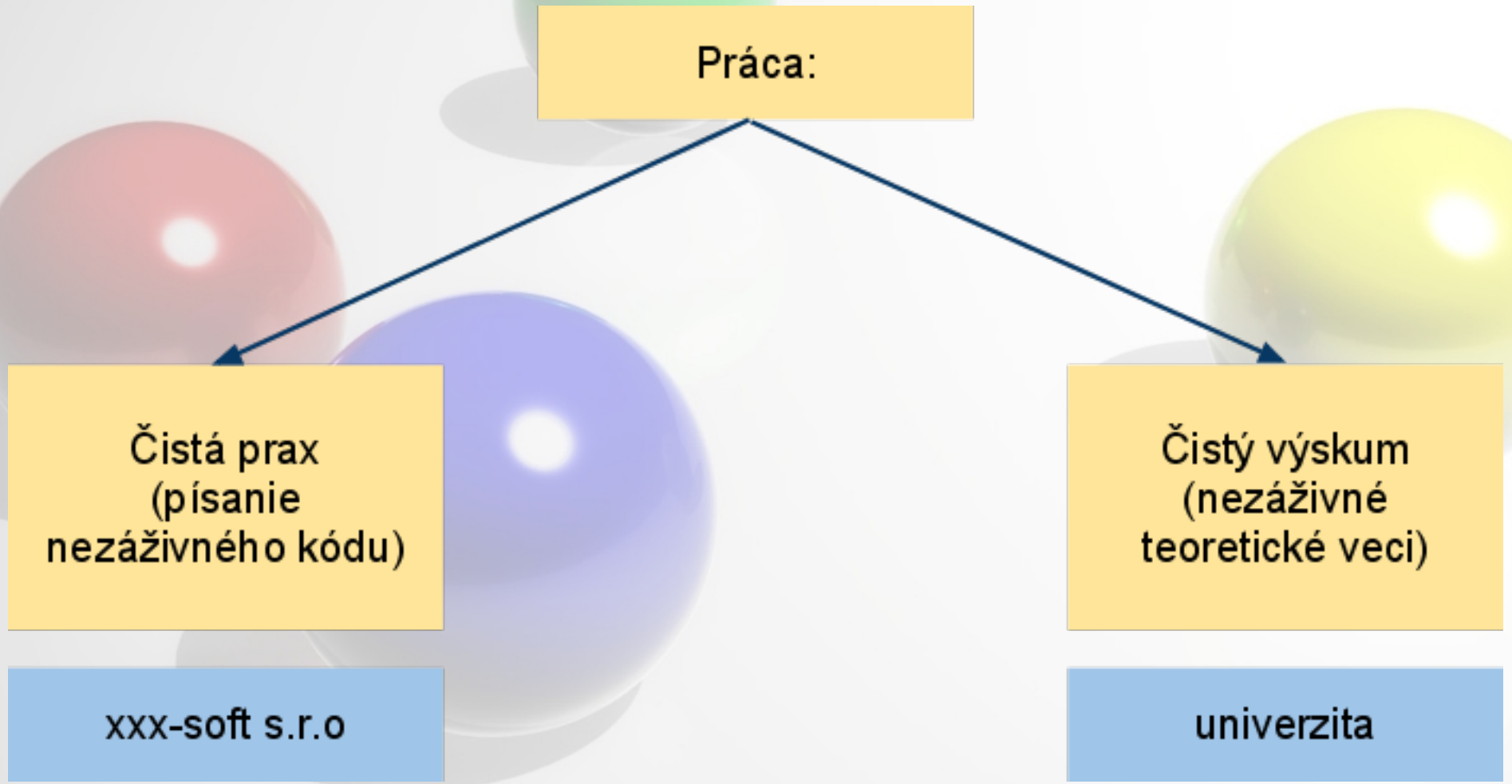
Možnosti Internshipu

alebo "Čo sa dá robiť?"

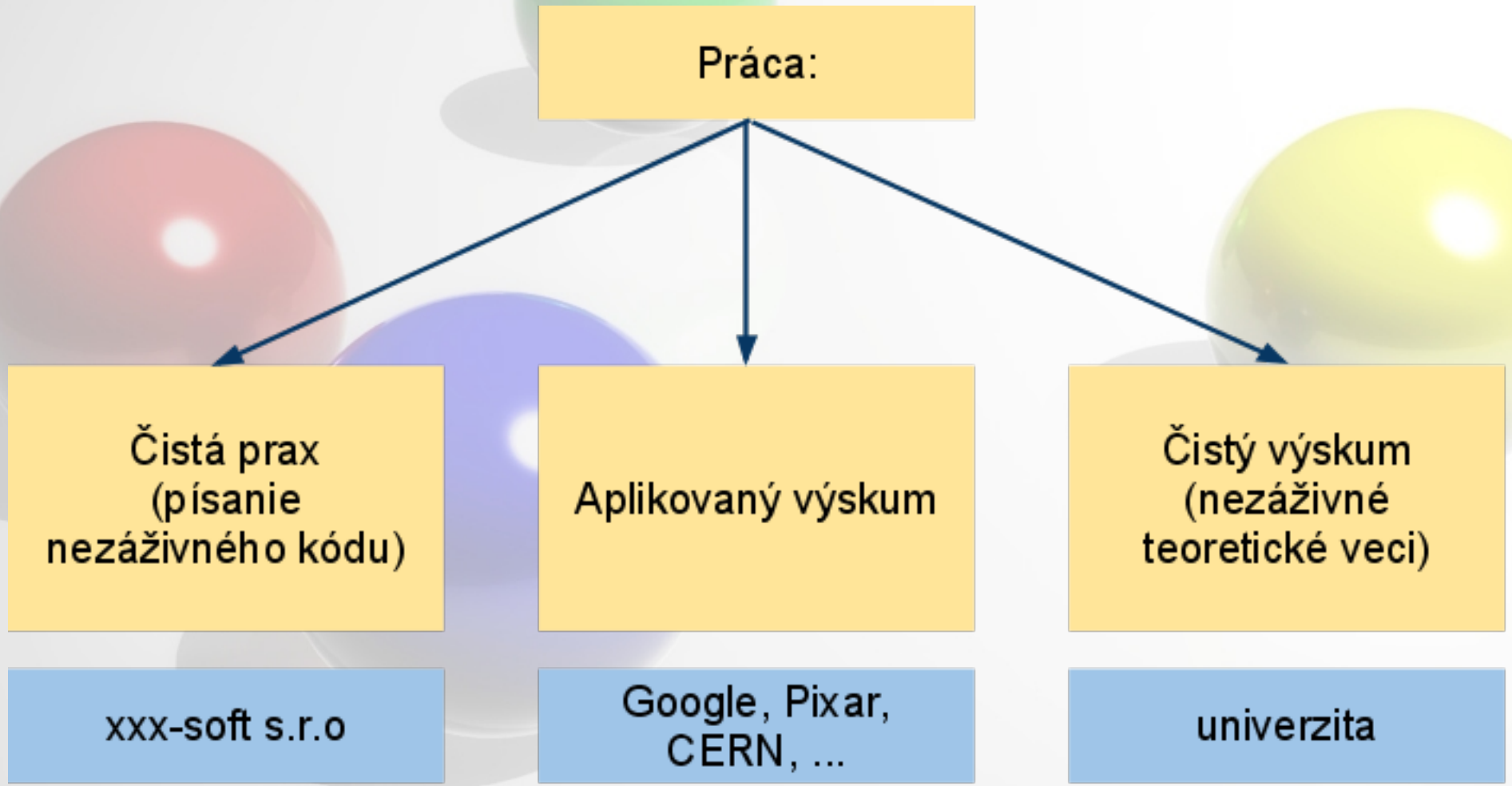
Google

CAINHOB

Možnosti po ukončení štúdia



Možnosti po ukončení štúdia



Google

CAINHOB

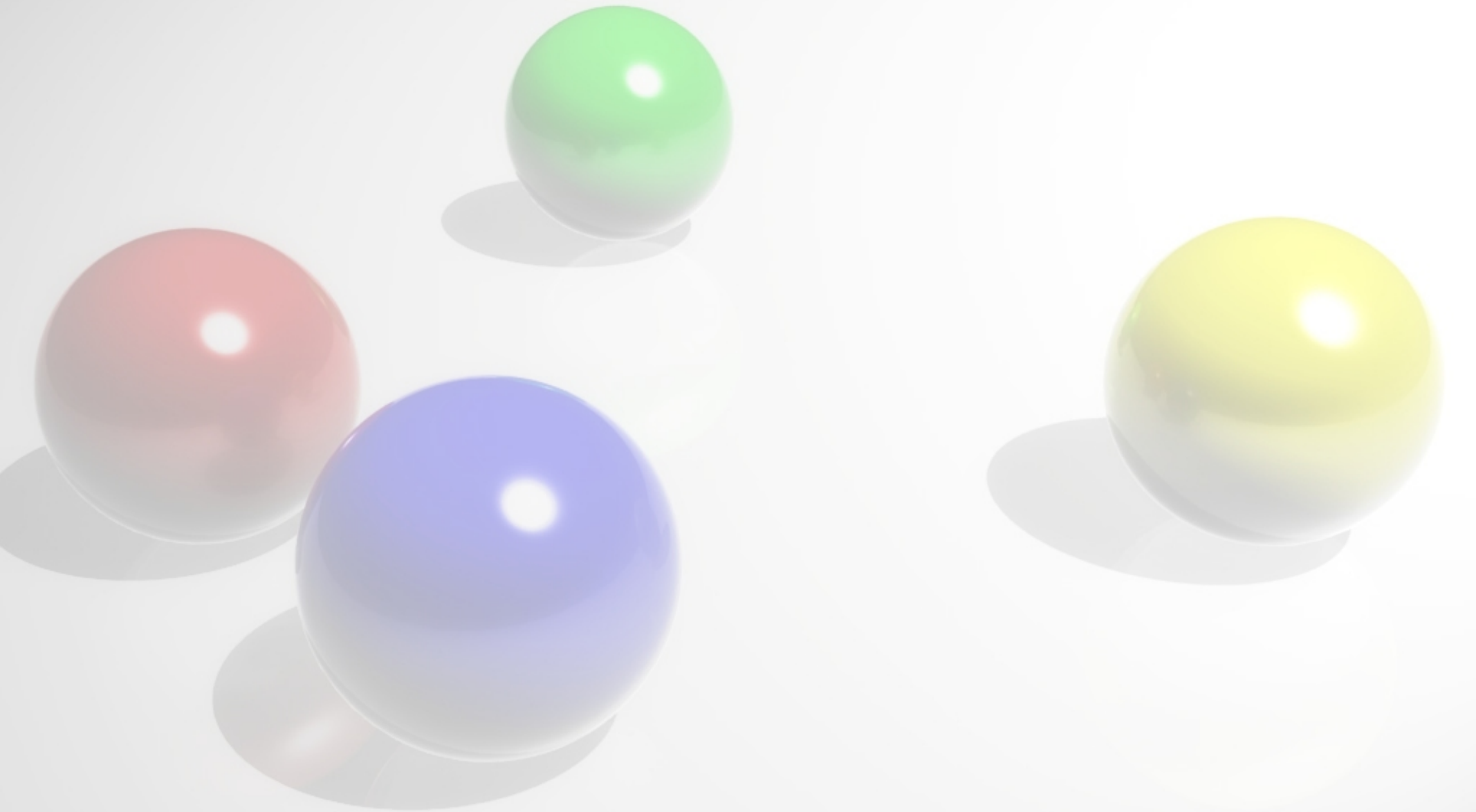
Možnosti internshipu

- IPAM (Institute for pure & applied mathematics) http://www.ipam.ucla.edu/about.aspx
- Microsoft Research
- CERN
- Google
- mnohé ďalšie

Prečo práve Google?

- softwarový gigant
  - 20000 zamestnancov
  - milióny užívaťeľov
- dobrá pracovná atmosféra
- architektúra a problémy, ktoré nemá nikto iný na svete

Google


Google

Ako vyzerá Google internship?

Google

CAINHOB

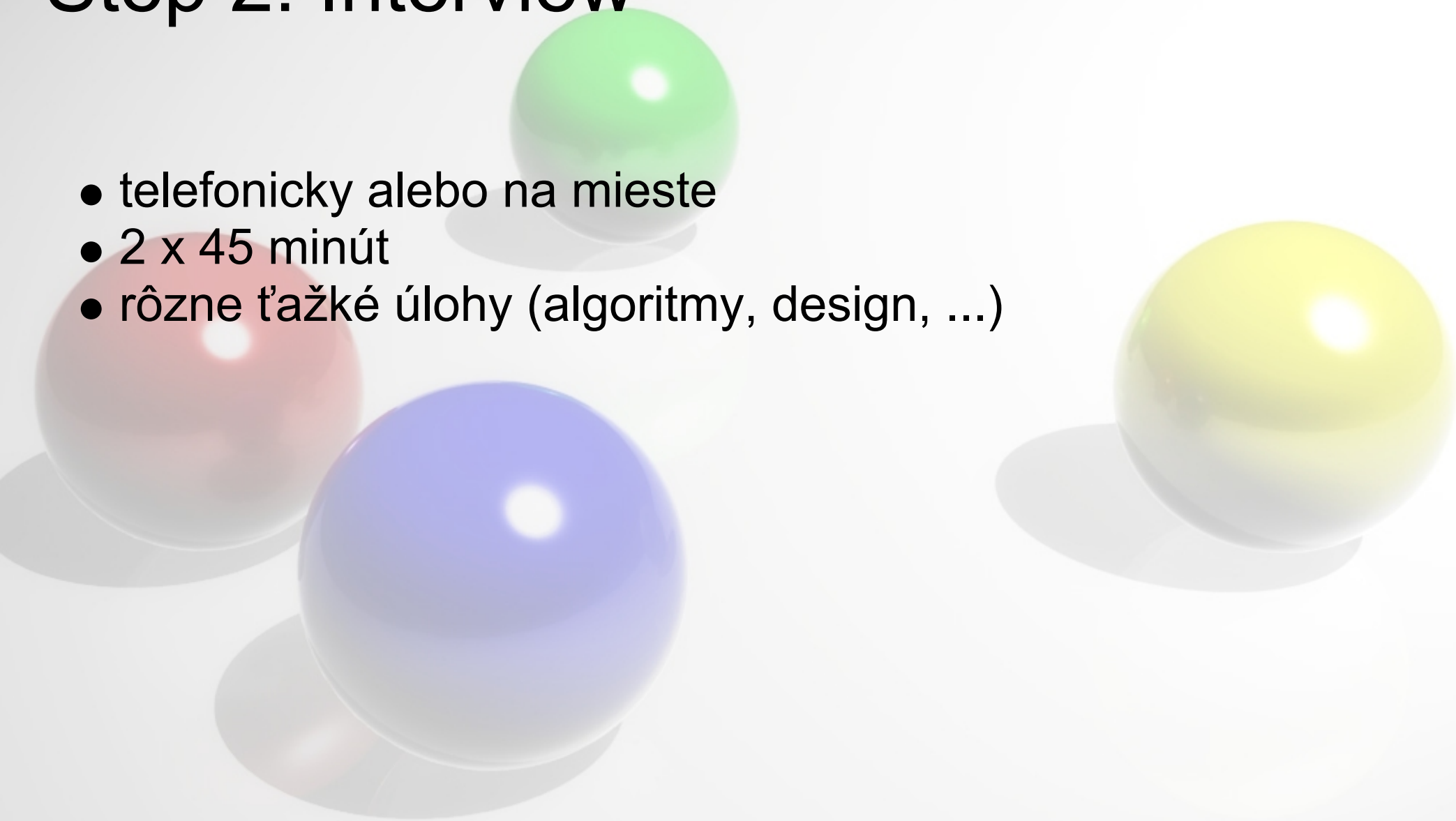
Step 1: Aplikácia na internship

- Treba dať o sebe vedieť
- voľné pozície - http://www.google.com/support/jobs/bin/static.py?page=students.html
- pre záujemcov môžem dodať nejaké kontakty

Google

Step 2: Interview

- telefonicky alebo na mieste
- 2 x 45 minút
- rôzne ťažké úlohy (algoritmy, design, ...)



Google

Step 3: Po interview

- 2 týždne čakať na rozhodnutie
- dohodnutie detailov internshipu
  - začiatok a koniec
  - pracovná zmluva

Google

Step 4: Pred začatím

čakanie na byrokraciu

- pracovné povolenie
- víza

Google

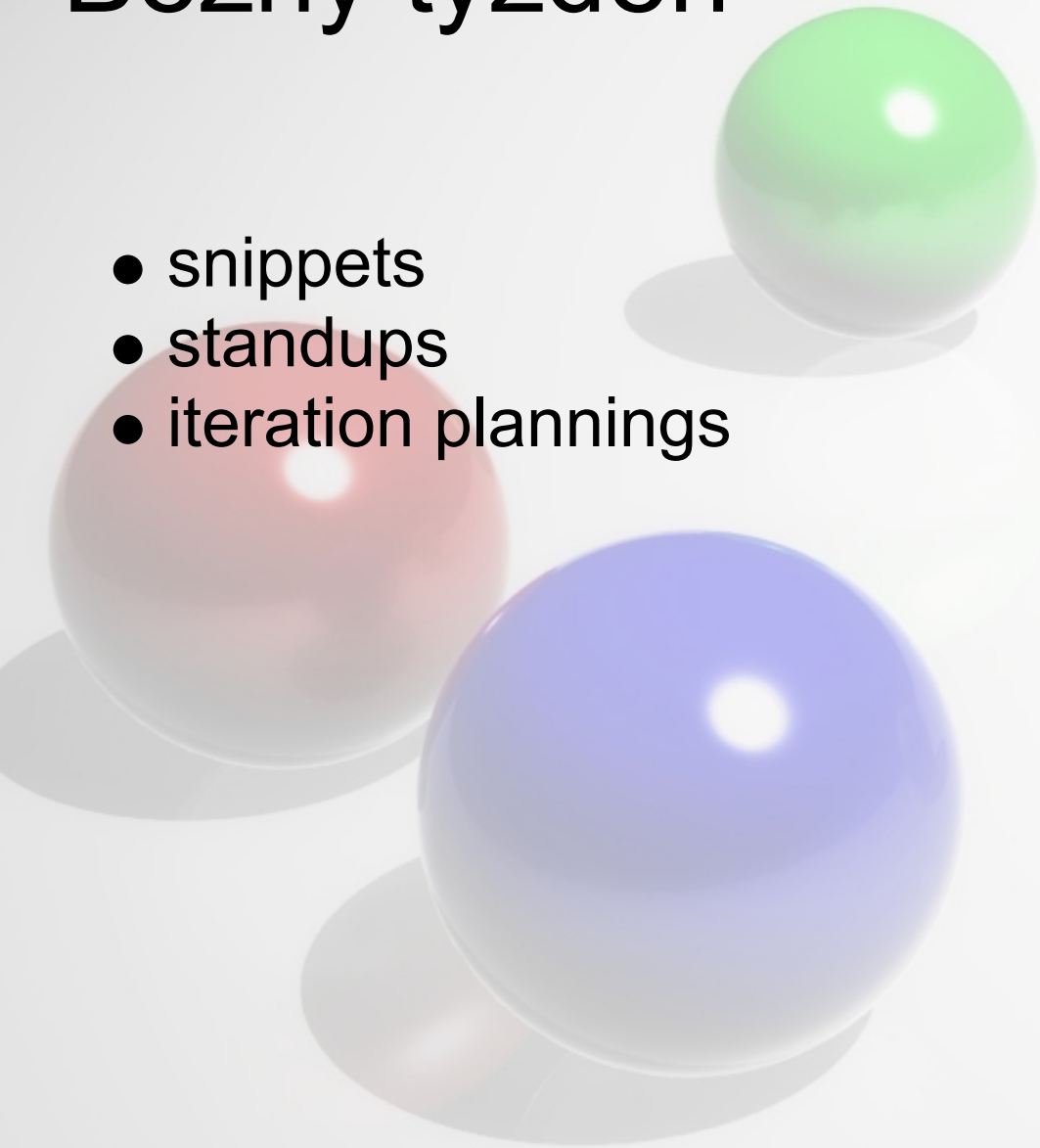
Internship - 1. týždeň

Asi najhorší týždeň

- byrokracia
  - registrovať sa na úradoch
  - bankové konto
  - zdravotné poistenie
  - ...
- prístupové práva
  - vybaviť N+1 rôznych accountov
  - na každé sa čaká, blokuje to ďalšie,

Bežný týždeň

- snippets
- standups
- iteration plannings



Google

Posledný týždeň

- opäť kopa byrokracie

Google

CAINHOB

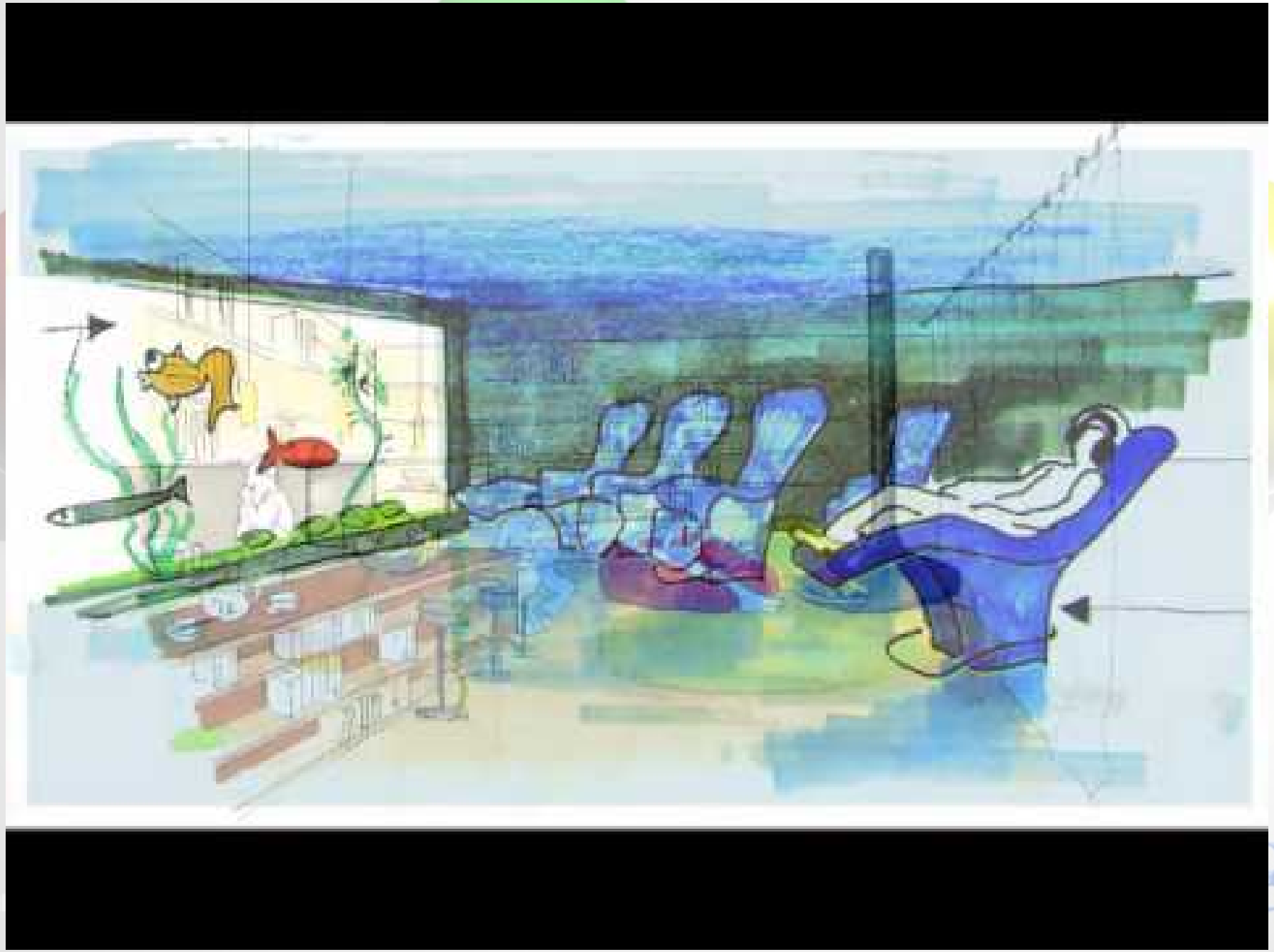
Google Zurich

Prečo chodiť do práce?

Google

CAINHOB

Google Zurich



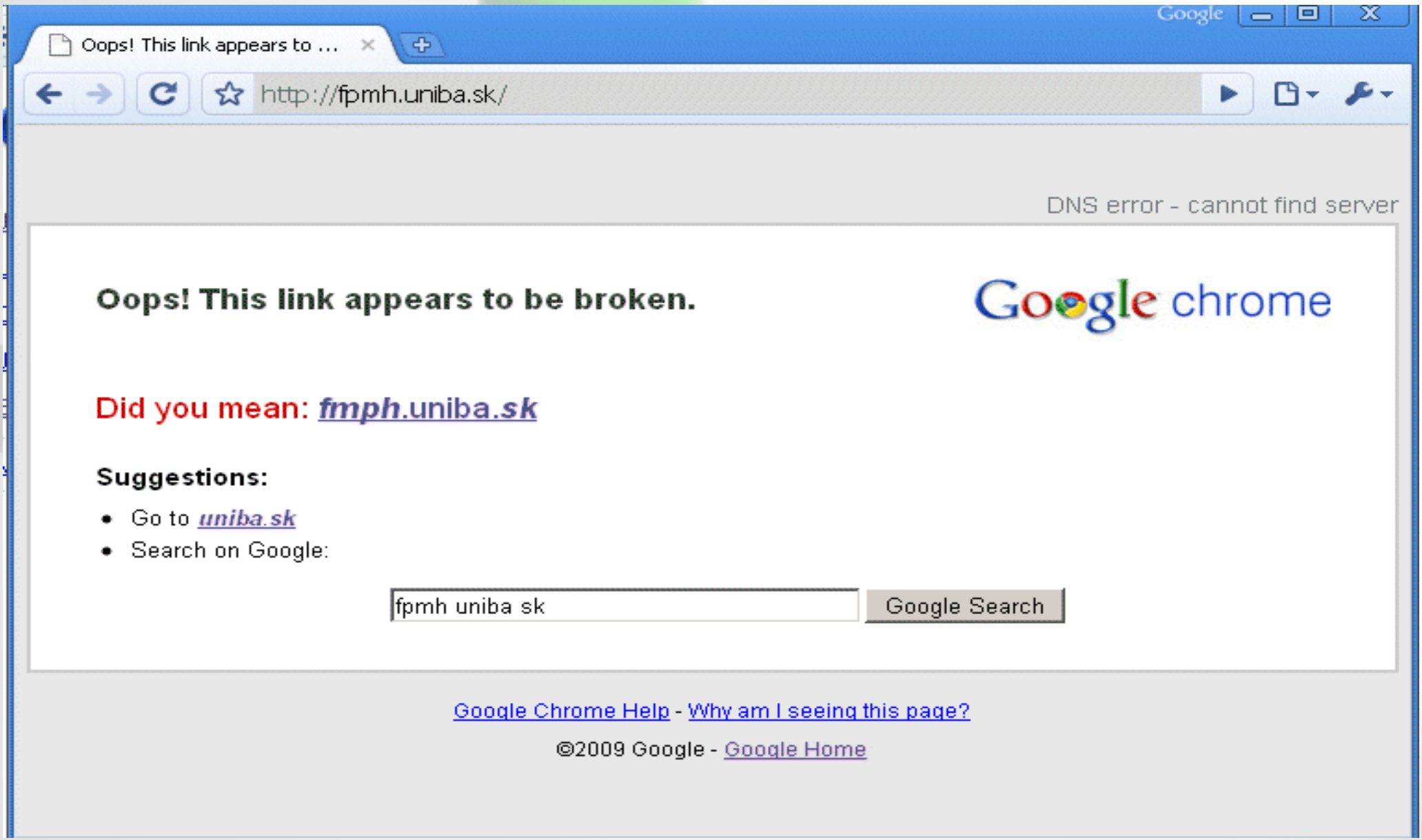
Google Zurich

Projekty na ktorých sa dá pracovať

- gmail
- search quality
- geo (maps, earth)
- ...

Google

Môj projekt


Google
Oops! This link appears to ...
http://fpmh.uniba.sk/
DNS error - cannot find server
Oops! This link appears to be broken.
Google chrome
Did you mean: fmph.uniba.sk
Suggestions:
Go to uniba.sk
Search on Google:
fpmh uniba sk
Google Search
Google Chrome Help - Why am I seeing this page?
©2009 Google - Google Home

Môj projekt

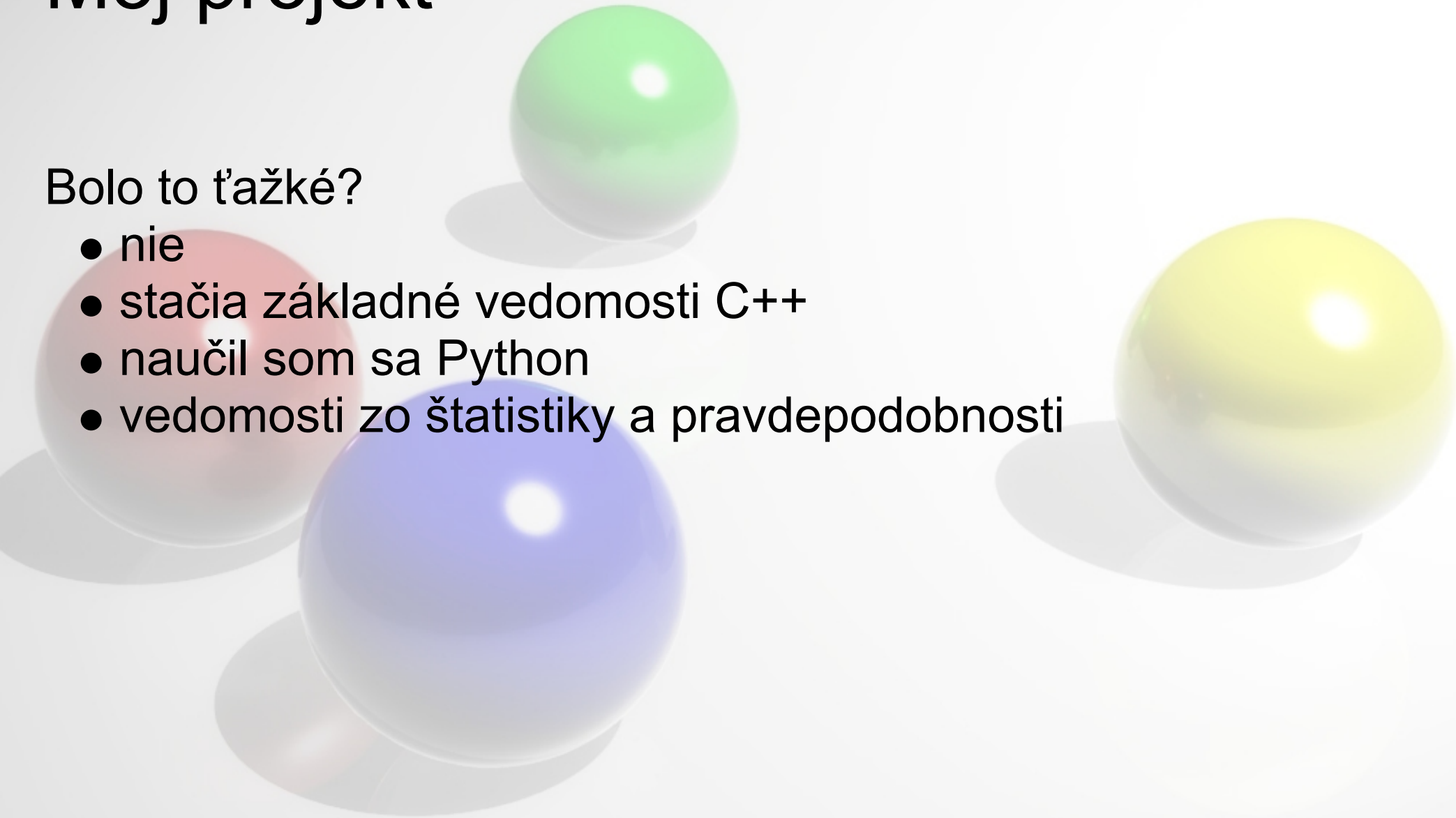
Čo bolo mojou prácou:

- štatistický model na zlepšenie presnosti
- kopec drobných bugov
- interné tooly
- pomoc pri zahájení novej verzie produktu

Môj projekt

Bolo to ťažké?

- nie
- stačia základné vedomosti C++
- naučil som sa Python
- vedomosti zo štatistiky a pravdepodobnosti



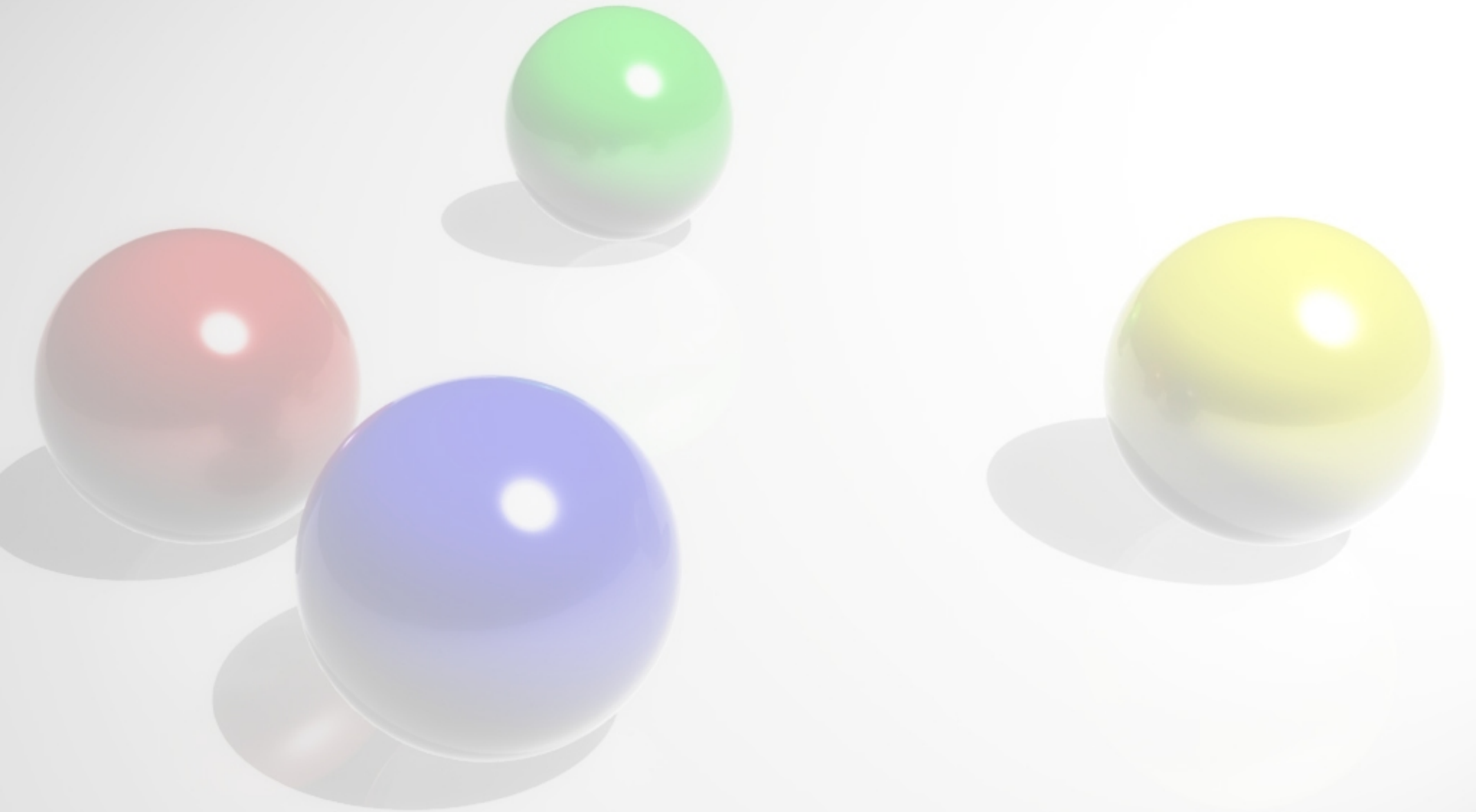
Google

CAINHOB

Môj projekt

Prečo potom Google má najlepších programátorov?

- efektivita
  - kompilácia 5 minút, inkrementálne 30 sekúnd
  - spúšťanie servera - vzdialené datacentrum, pár desiatok počítačov, 10 minút
- spoľahlivosť
  - submitnutý kód musí byť spoľahlivý
- nápady


Google

Productivity tools

Google

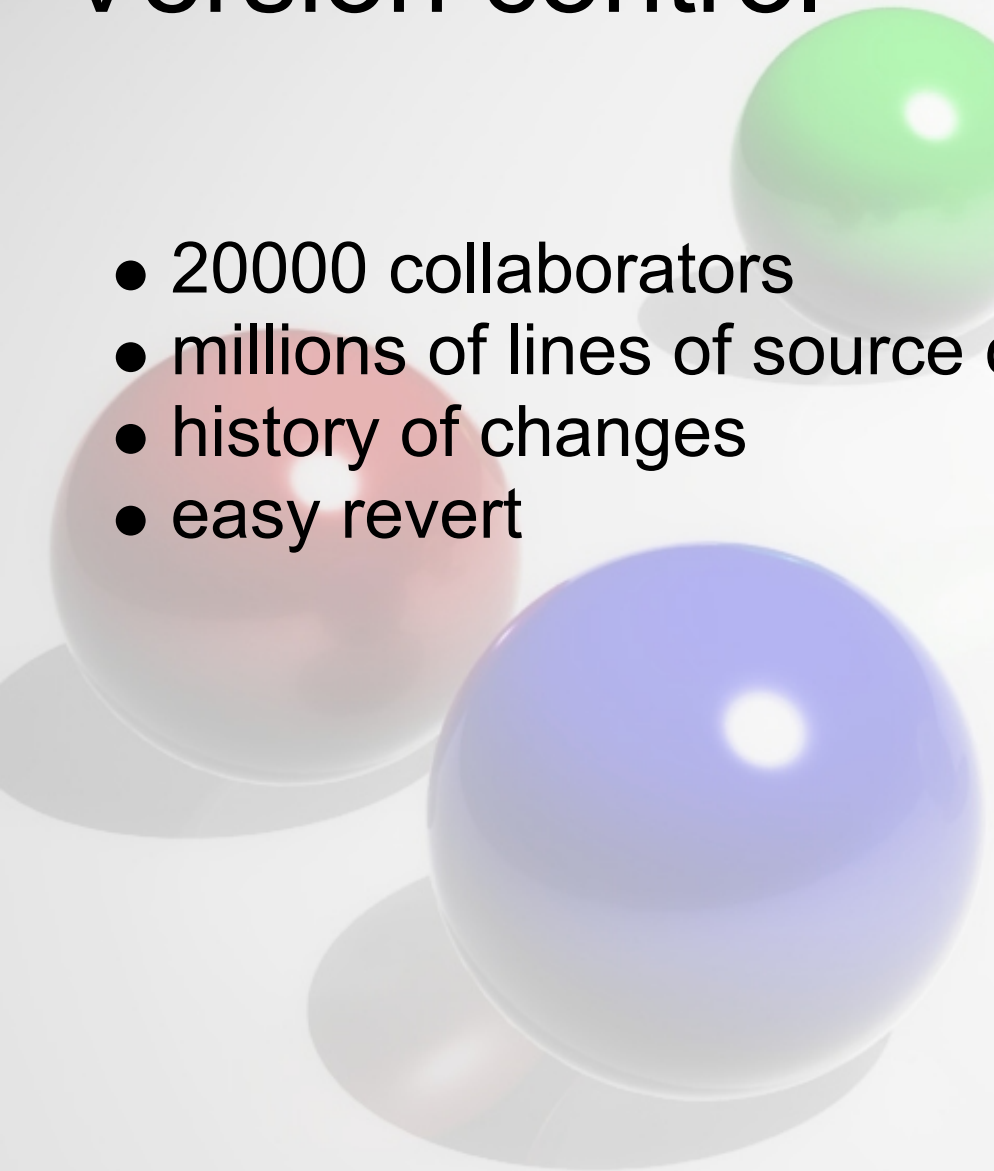
Niektoré nástroje

- perforce/version control
- mondrian/review board
- bug tracker
- unit tests
- regression tests
- load tests

Google

Version control

- 20000 collaborators
- millions of lines of source code
- history of changes
- easy revert



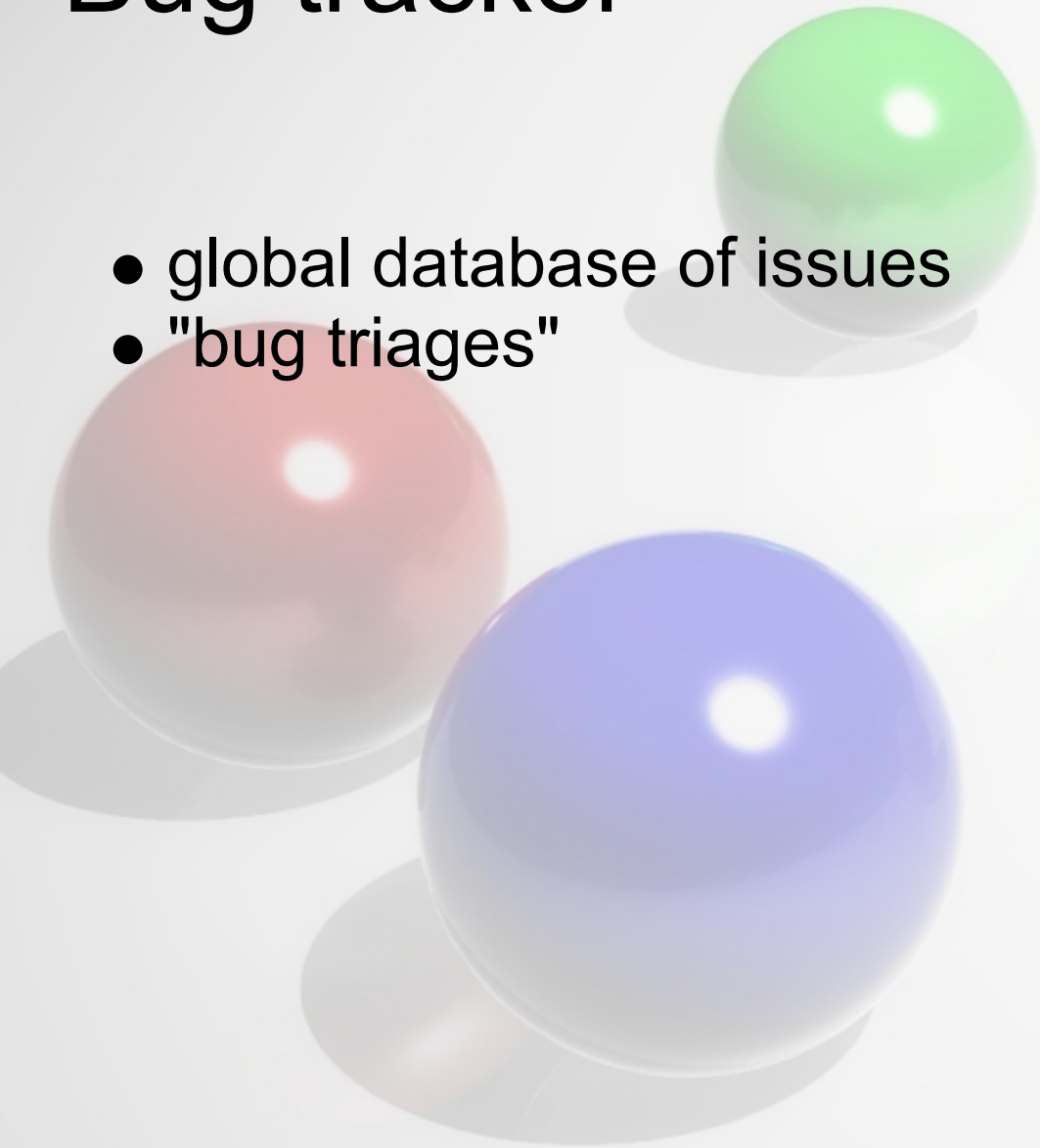
Google

Peer review

- more than one developer knows the code
- force developers to submit clean, understandable code
- reviewer may have good ideas (design/refactoring, naming functions, simplifying code, forgotten todo's, ...)

Bug tracker

- global database of issues
- "bug triages"



Google

Unittests & test driven development

Why to test even simple code?

- it can be buggy
- testing rocks, debugging sucks
- someone other can break it while refactoring/adding new features
- catch problems even before start of your server

- catch problems before submitting the code

Unittests & test driven development

```
// Tests Factorial()
TEST_F(IntegerFunctionTest, Factorial) {
  // Tests factorial of negative numbers.
  EXPECT_EQ(1, Factorial(-5));
  EXPECT_EQ(1, Factorial(-1));
  EXPECT_TRUE(Factorial(-10) > 0);

  // Tests factorial of 0.
  EXPECT_EQ(1, Factorial(0));

  // Tests factorial of positive numbers.
  EXPECT_EQ(1, Factorial(1));
  EXPECT_EQ(2, Factorial(2));
  EXPECT_EQ(6, Factorial(3));
  EXPECT_EQ(40320, Factorial(8));
}
```

```
// Tests IsPrime()
TEST_F(IntegerFunctionTest, IsPrime) {
  // Tests negative input.
  EXPECT_FALSE(IsPrime(-1));
  EXPECT_FALSE(IsPrime(-2));
  EXPECT_FALSE(IsPrime(INT_MIN));

  // Tests some trivial cases.
  EXPECT_FALSE(IsPrime(0));
  EXPECT_FALSE(IsPrime(1));
  EXPECT_TRUE(IsPrime(2));
  EXPECT_TRUE(IsPrime(3));

  // Tests positive input.
  EXPECT_FALSE(IsPrime(4));
  EXPECT_TRUE(IsPrime(5));
  EXPECT_FALSE(IsPrime(6));
  EXPECT_TRUE(IsPrime(23));
}
```

Testing & mocks

- What are mocks?

- Why mocks?
  - break dependencies, dependency injections
  - test object without collaborators
  - smaller test
  - easy test of databases, filesystems, remote servers

Testing & mocks

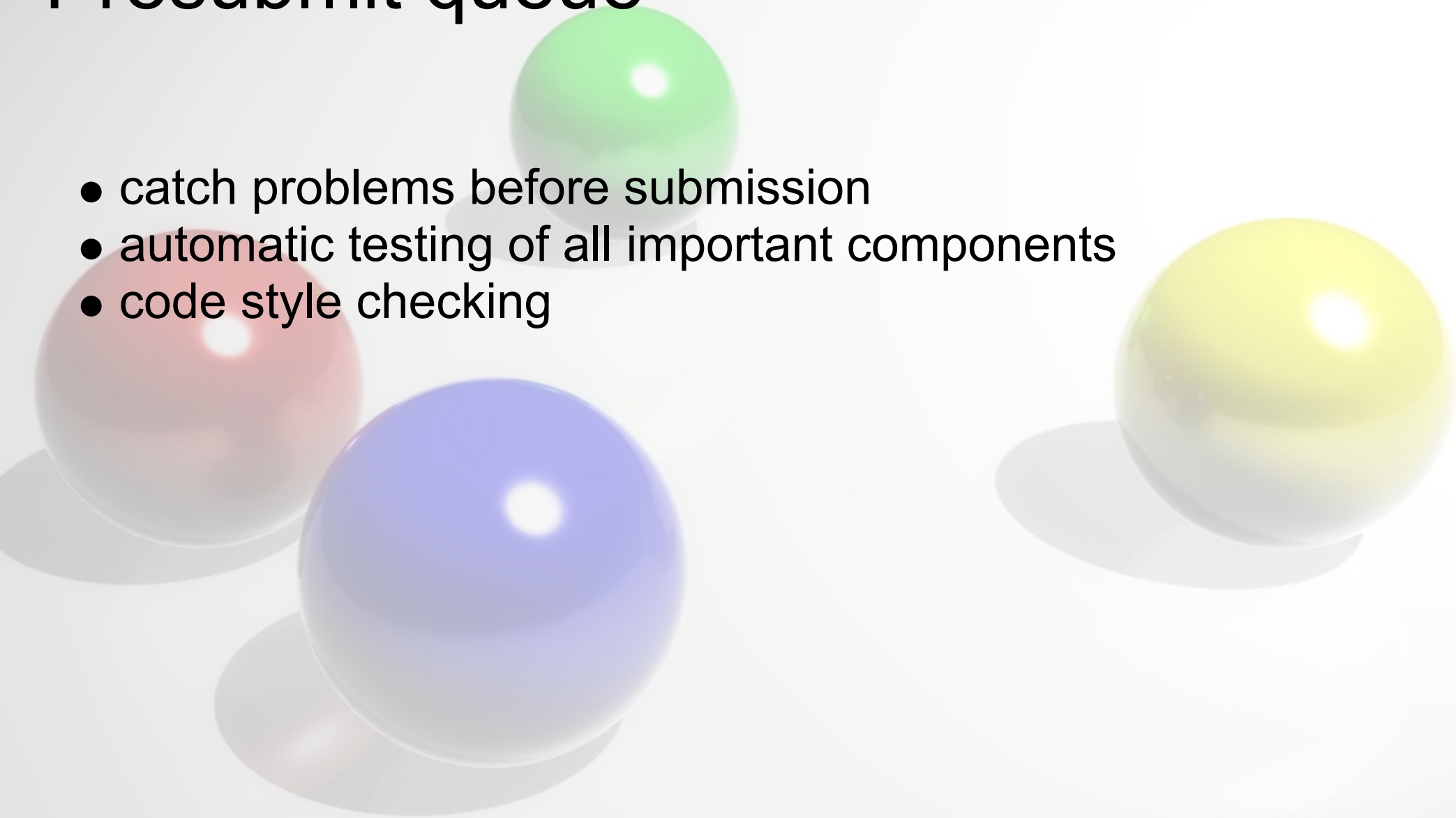
```
class Turtle {
  ...
  virtual ~Turtle();
  virtual void PenUp() = 0;
  virtual void PenDown() = 0;
  virtual void Forward(int distance) = 0;
  virtual void Turn(int degrees) = 0;
  virtual void GoTo(int x, int y) = 0;
  virtual int GetX() const = 0;
  virtual int GetY() const = 0;
};
```

```
#include <gmock/gmock.h>
class MockTurtle : public Turtle {
 public:
  ...
  MOCK_METHOD0(PenUp, void());
  MOCK_METHOD0(PenDown, void());
  MOCK_METHOD1(Forward, void(int distance));
  MOCK_METHOD1(Turn, void(int degrees));
  MOCK_METHOD2(GoTo, void(int x, int y));
  MOCK_CONST_METHOD0(GetX, int());
  MOCK_CONST_METHOD0(GetY, int());
};
```

```
TEST_F(DrawerTest, TestSomething) {
MockTurtle turtle;
EXPECT_CALL(turtle, GetY())
.WillOnce(Return(100))
.WillOnce(Return(200))
.WillRepeatedly(Return(300));
Drawer drawer;
drawer.do_something(turtle);
}
```

Presubmit queue

- catch problems before submission
- automatic testing of all important components
- code style checking



Google

Regression tests

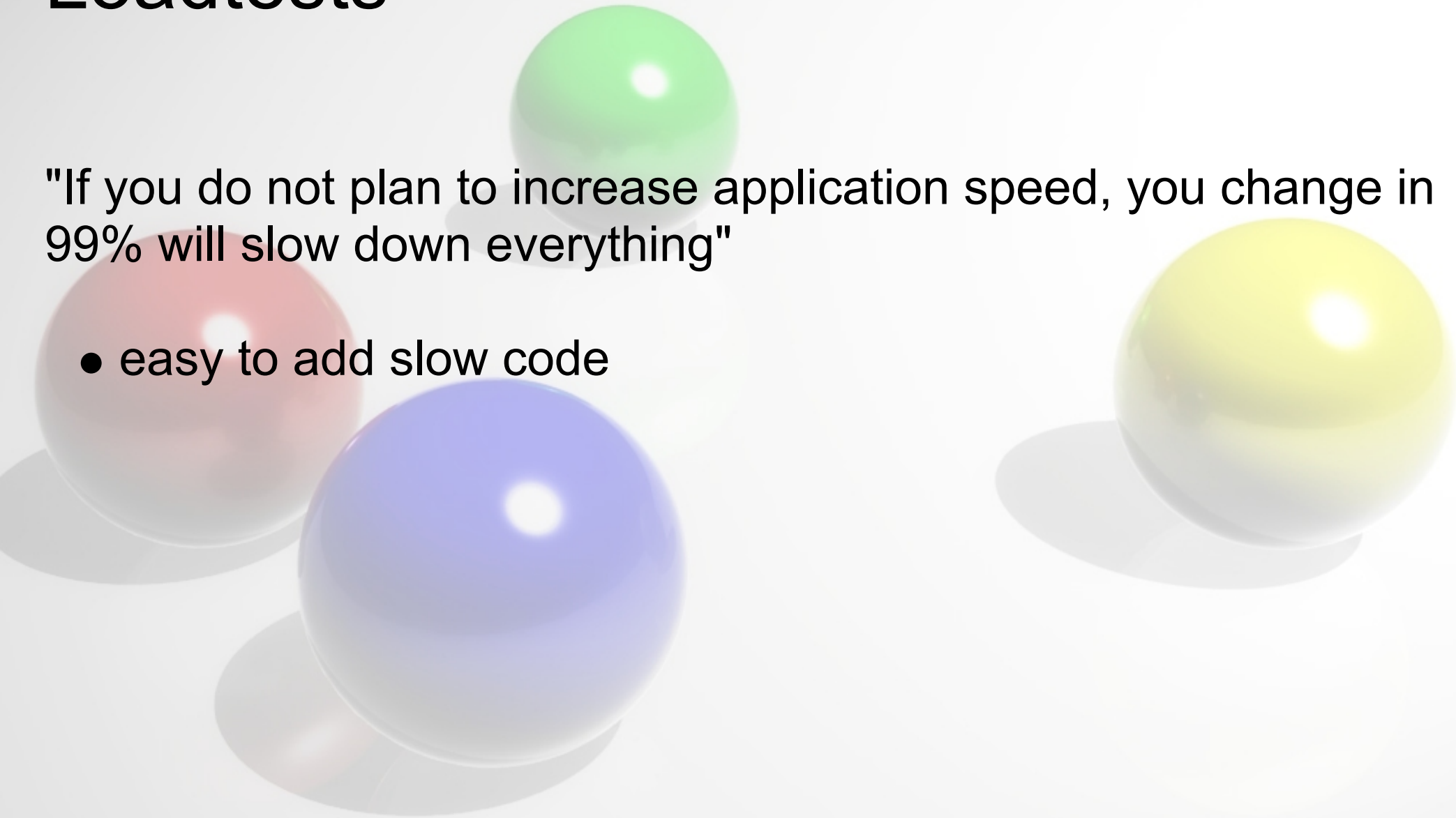
We have unittests, why to bother?

- tested code may not run in production (commented out)
- need of production environment
  - client-server, network, database, real queries
- unittest should be small, simple & fast

Loadtests

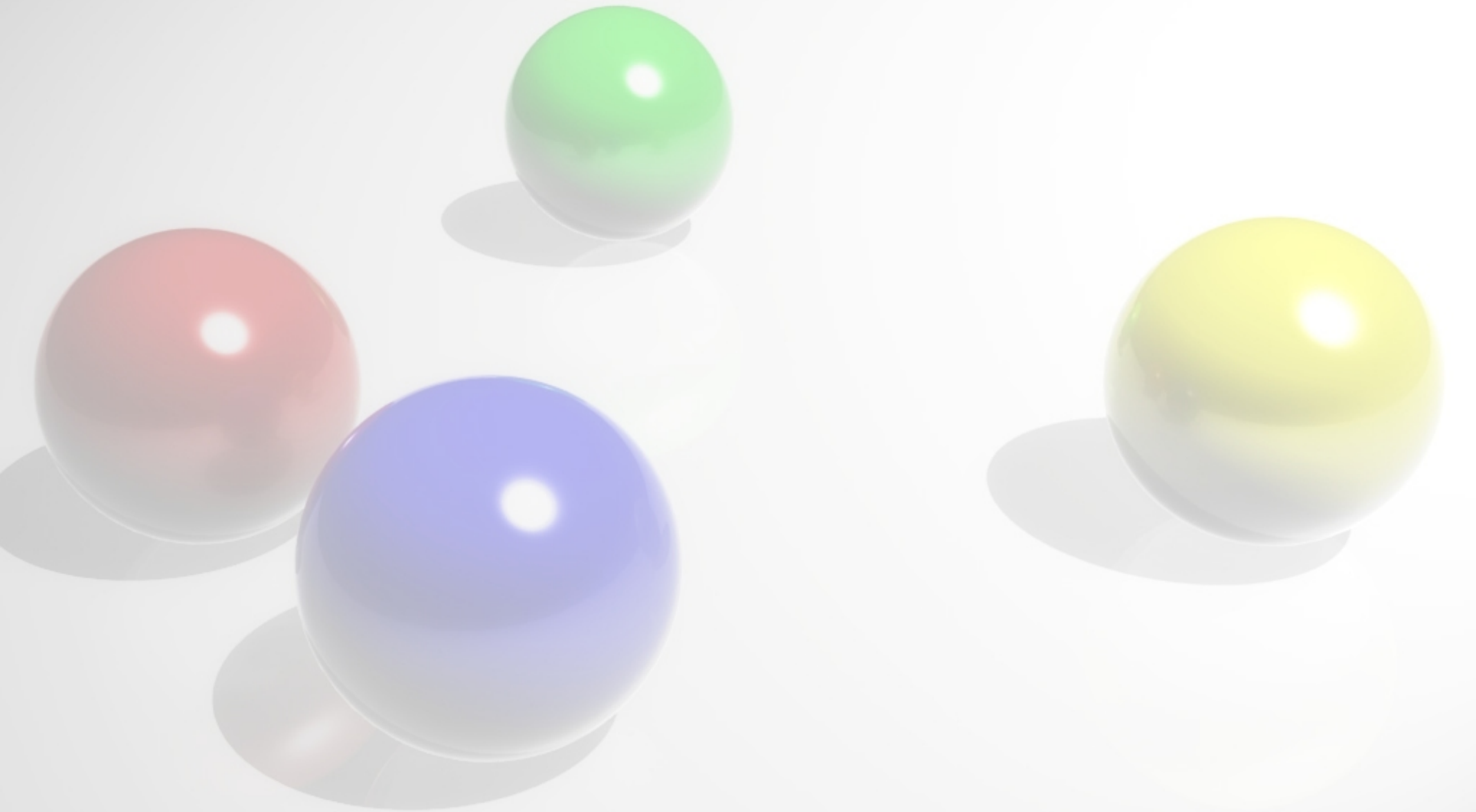
"If you do not plan to increase application speed, you change in 99% will slow down everything"

- easy to add slow code



Open source variants

- version control - svn, csv, git, ...
- reviews - reviewboard
- bug tracker - bugzilla, ...
- unittests - google tests (http://code.google.com/p/googletest/), CppUnit
- mocks
  - Java: jMock, EasyMock
  - C++: google mocks (http://code.google.com/p/googlemock)


Google

Voľná diskusia

Máte nejaké otázky?

Google

CAINHOB