



# Практика разработки и сопровождения приложений для встраиваемых систем

Александр Рыльцов



**АСТРОСОФТ**

[www.astrosoft.ru](http://www.astrosoft.ru)

## План

Цикл разработки

Стоимость и ценность

Правила руки:

сохранение знаний

подбор инструментов

методы «партизанской» оптимизации

Q&A

Встраиваемая система - это:

**система,**

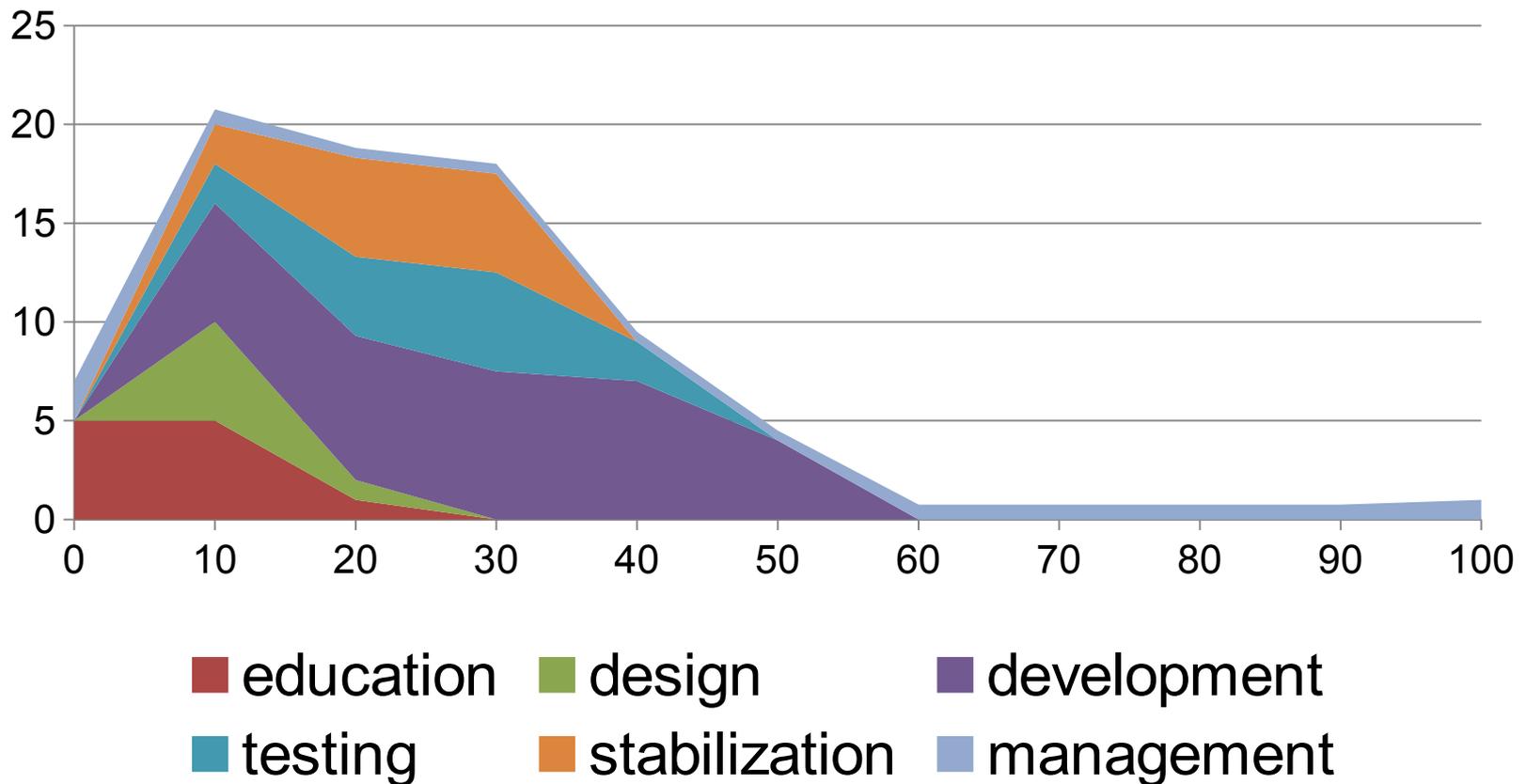
**при разработке которой**

**нельзя игнорировать**

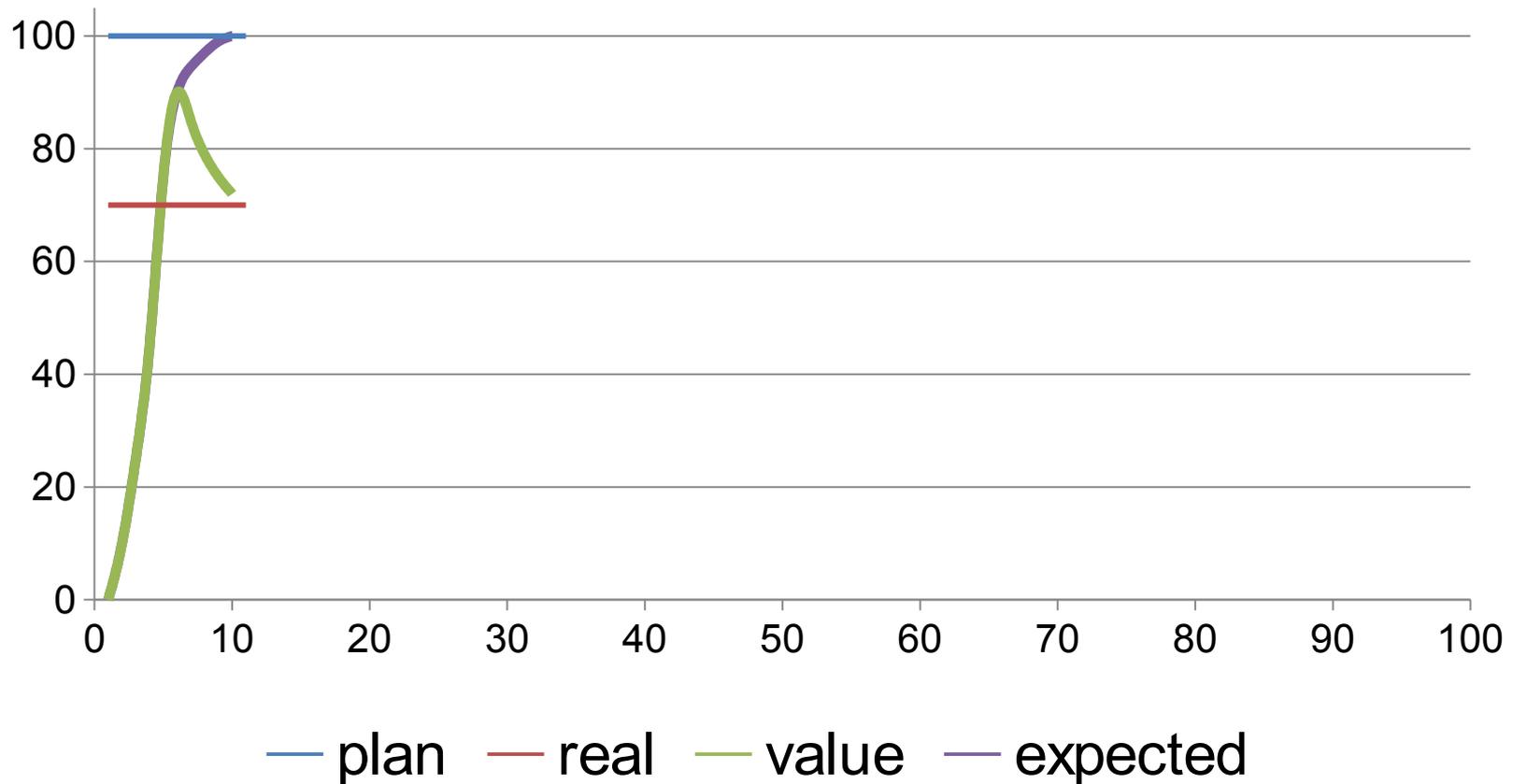
**ограниченность**

**ресурсов**

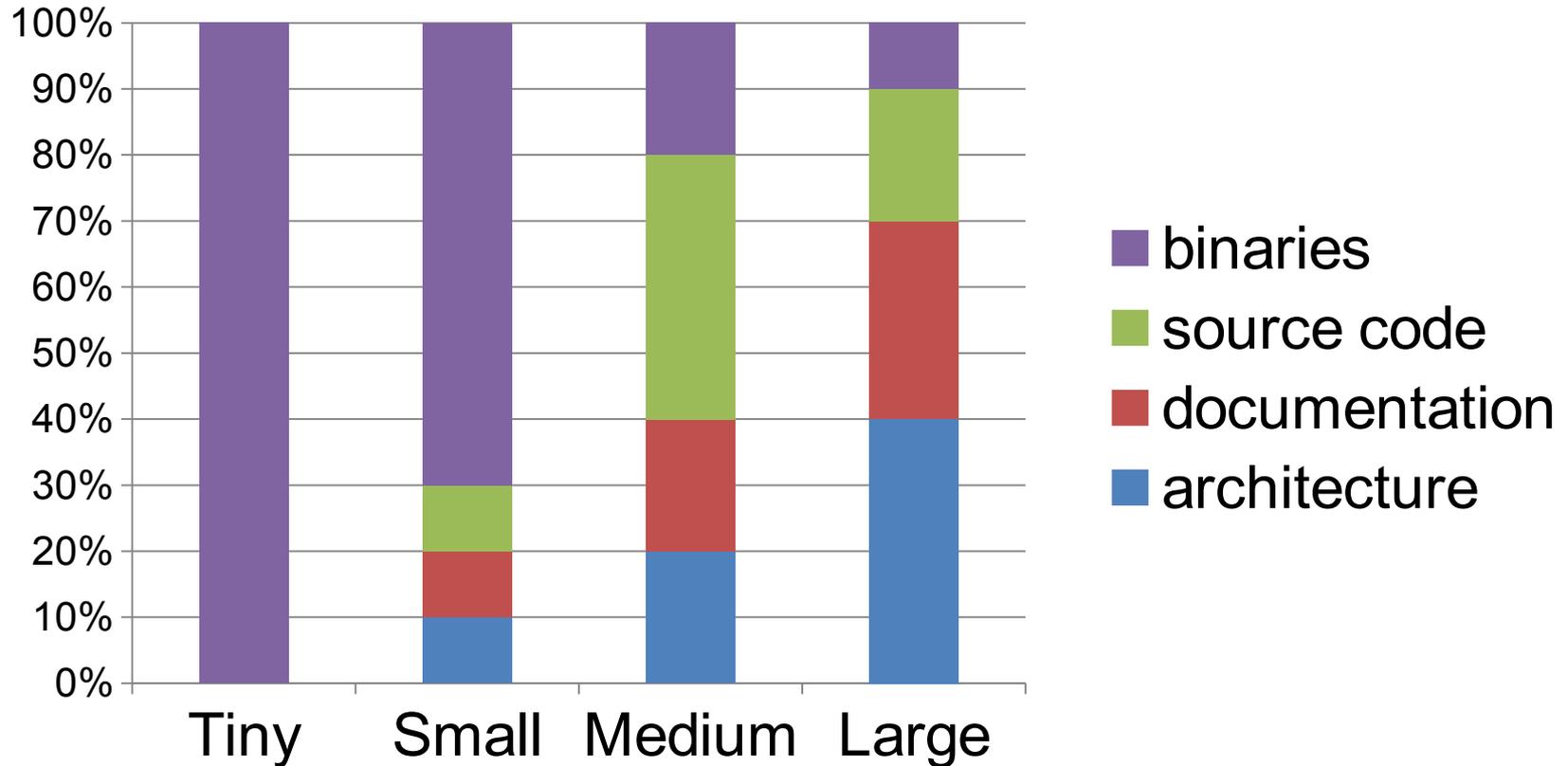
## Цикл проекта – структура работ



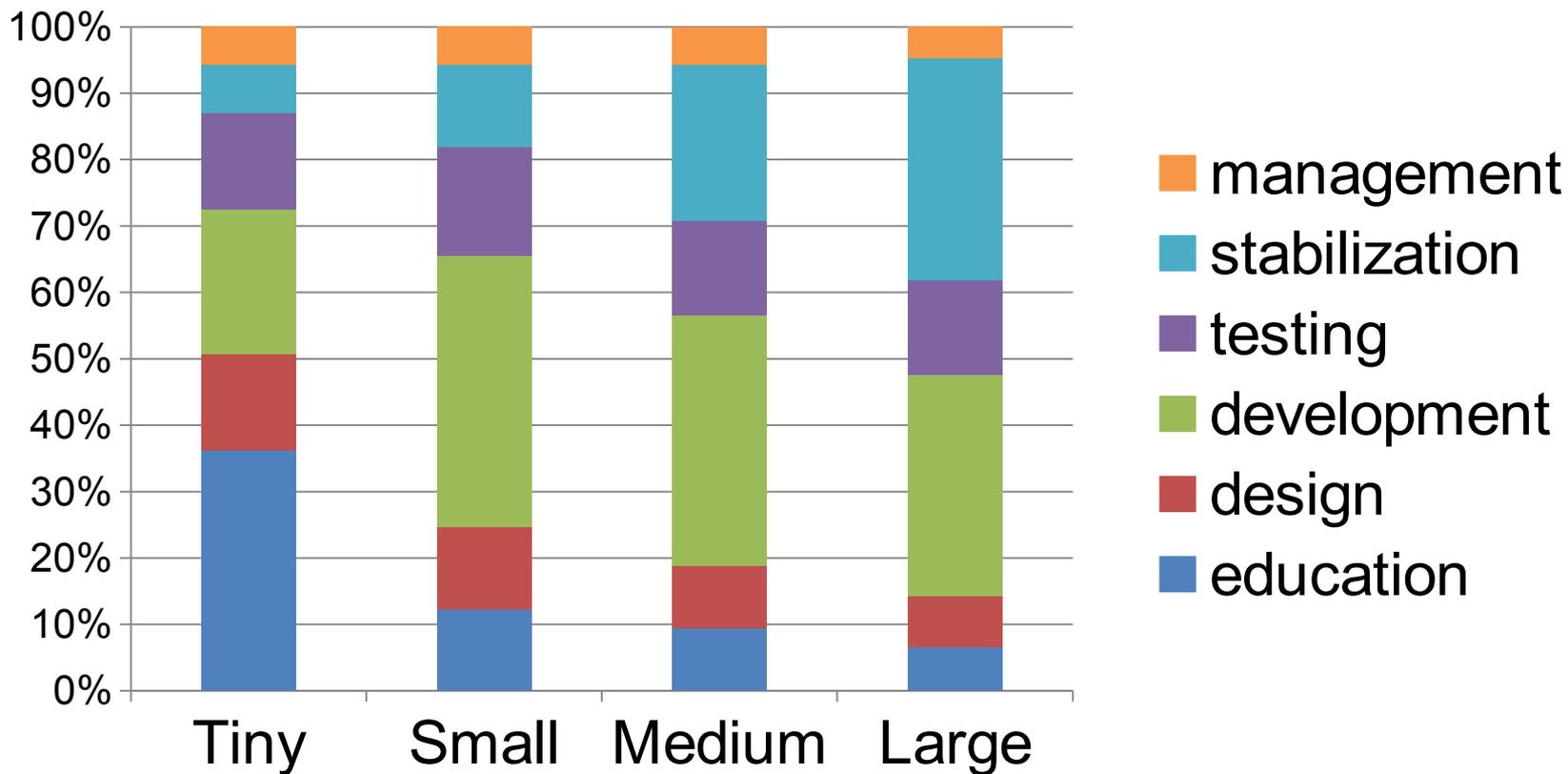
## Цикл проекта – рост функционала



## Ценность в проекте



## Стоимость работ в проекте



# Ценности проекта

цели

архитектура (ADS)

идиомы / примеры

«грабли»

детальная документация (SDD)

код

внешние библиотеки

устаревшие примеры и документы

промежуточные формы (binaries, \_l.h, etc.)

## Ценности проекта

цели

архитектура (ADS)

идиомы / примеры

«грабли»

детальная документация (SDD)

код

внешние библиотеки

устаревшие примеры и документы

промежуточные формы (binaries, \_l.h, etc.)

## Ценности проекта

цели  
архитектура (ADS)  
идиомы / примеры  
«грабли»

детальная документация (SDD)

код

внешние библиотеки

устаревшие примеры и документы  
промежуточные формы (binaries, \_l.h, etc.)

## Ценности проекта

цели  
архитектура (ADS)  
идиомы / примеры  
«грабли»

детальная документация (SDD)

код

внешние библиотеки

устаревшие примеры и документы  
промежуточные формы (binaries, \_l.h, etc.)

## Документация – «правила руки»

Не пишите детальную документацию

Используйте:

структуру / имена

комментарии

компактное автодокументирование

ссылки на внешние спецификации

Пишите **работающие** примеры

Используйте идиомы и синонимы

Удаляйте мусор

Подробно описывайте в коде найденные «грабли»

## Используйте «жаргон»

Жаргон (project-specific lingua) - это:  
синонимы для сущностей «стандартной» библиотеки

идиомы использования «стандартной» библиотеки

кирпичики для построения библиотек в предметной области

Для чего:

структура и единообразие

выделение рекомендованного подмножества «стандартной» библиотеки

выделение шаблонов, специфичных для предметной области проекта

скрытие деталей

## Пример жаргона (C#) – валидация

```
NetworkZeroConfiguration zeroConf = GetZeroConfiguration();
RunNormal("Verifying appliance of IPv4 zero settings", new Action(() =>
{
    MyAssert(
        LogLog("Check for interface token = " + token) &&
        LogAssert(zeroConf, "zero configuration") &&
        LogAssert(zeroConf.Enabled, "zero configuration not enabled") &&
        LogAssertComp(zeroConf.InterfaceToken, token, "interface token") &&
        LogAssert(zeroConf.Addresses, "Addresses") &&
        LogAssertPrefixString(zeroConf.Addresses,
modified.Ipv4.Config.LinkLocal.Address)
        , "Zero settings not applied", "Zero settings applied successfully");
    }));
```

## Пример жаргона (C++) – сериализация

```
DECLARE_SERIAL_CLASS(serialxml::CXmlContext, onvif::SConfig)
  SET_CURRENT_NAMESPACE("xs", "http://www.w3.org/2001/XMLSchema")
  SERIALIZE_ATTRIBUTE(Name)
  SERIALIZE_ATTRIBUTE(Type)
  SET_CURRENT_NAMESPACE("tt", "http://www.onvif.org/ver10/schema")
  SERIALIZE_MEMBER(Parameters)
DECLARE_SERIAL_END(serialxml::CXmlContext, onvif::SConfig)

DECLARE_SERIAL_CLASS(myPrivateStream, onvif::SDeviceInformation)
  SERIALIZE_MEMBER2(CameraName, "/13/1/1/1")
  SERIALIZE_MEMBER2(FirmwareVersion, "/1/1/3")
  SERIALIZE_MEMBER2_AS(Manufacturer, "/1/1/7", TUTF8String)
DECLARE_SERIAL_END(myPrivateStream, onvif::SDeviceInformation)
```

## Пример жаргона (C++) – сопроцедуры

```
reenter(this) {  
    ISSUE_CALL_1(API_INIT_DONE, lib_showInfo, 0); yield return;  
    SKIP_FUNCTION(API_PRE_INFO);  
    ISSUE_CALL_3(API_SHOW_INFO, lib_reConfigure, MOVE_TO_BEGIN, 0,  
0); yield return;  
    if (ulFunctionID == API_RE_CONFIGURE) {  
        ts::TLiterals configureResult = *((ts::TLiterals*)pResult);  
        BOOST_TEST_MESSAGE("\n" << configureResult << "\n");  
    }  
    ISSUE_CALL(API_RE_CONFIGURE, lib_cancel); yield return;  
    SKIP_FUNCTION(API_RE_CONFIGURE);  
    ISSUE_EPILOG(API_CANCEL);  
};
```

## Обучение – «правила руки»

Выберите инструмент, архитектуру и стиль кодирования, подходящие друг другу, проверьте:

единообразие на всех программно-аппаратных платформах

стабильность / естественность

закономерность / последовательность / детерминированность

ресурсоёмкость / масштабируемость

Только выбранные инструменты **официально** используются в разработке

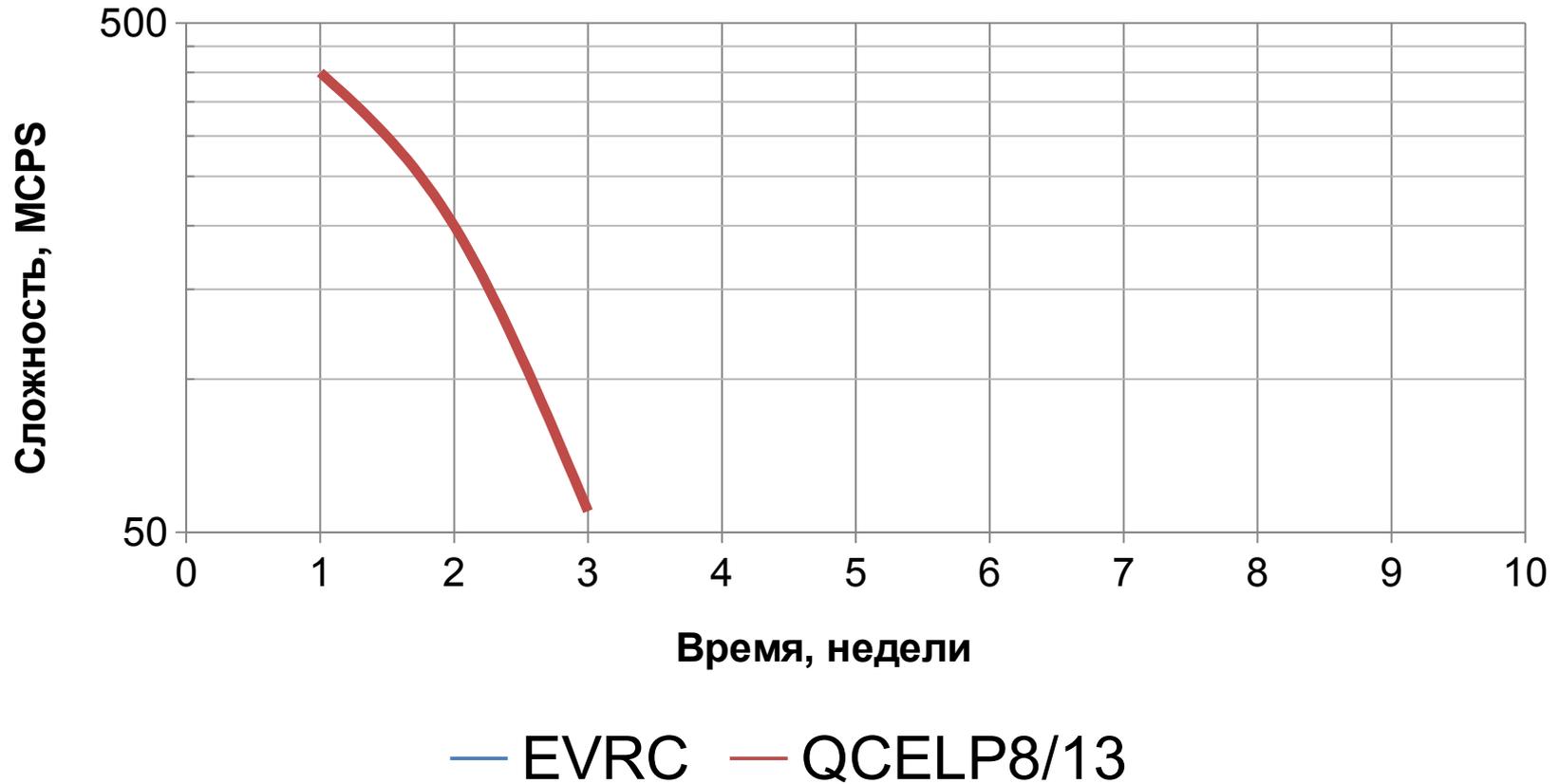
## «Партизанская» оптимизация

Не лучший результат,

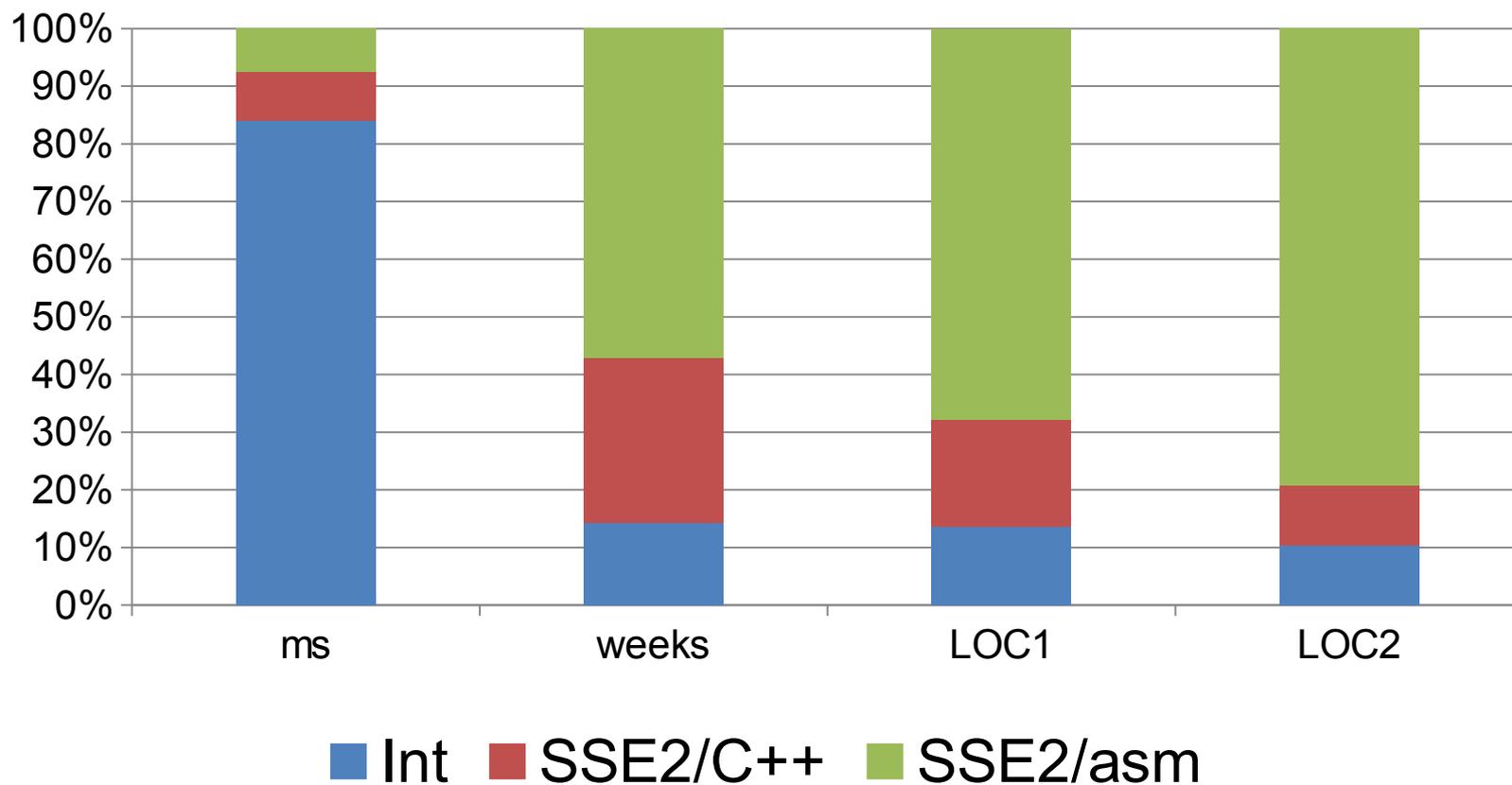
но **приемлемый** и **максимально дешёвый**

Нестандартные комбинации стандартных средств

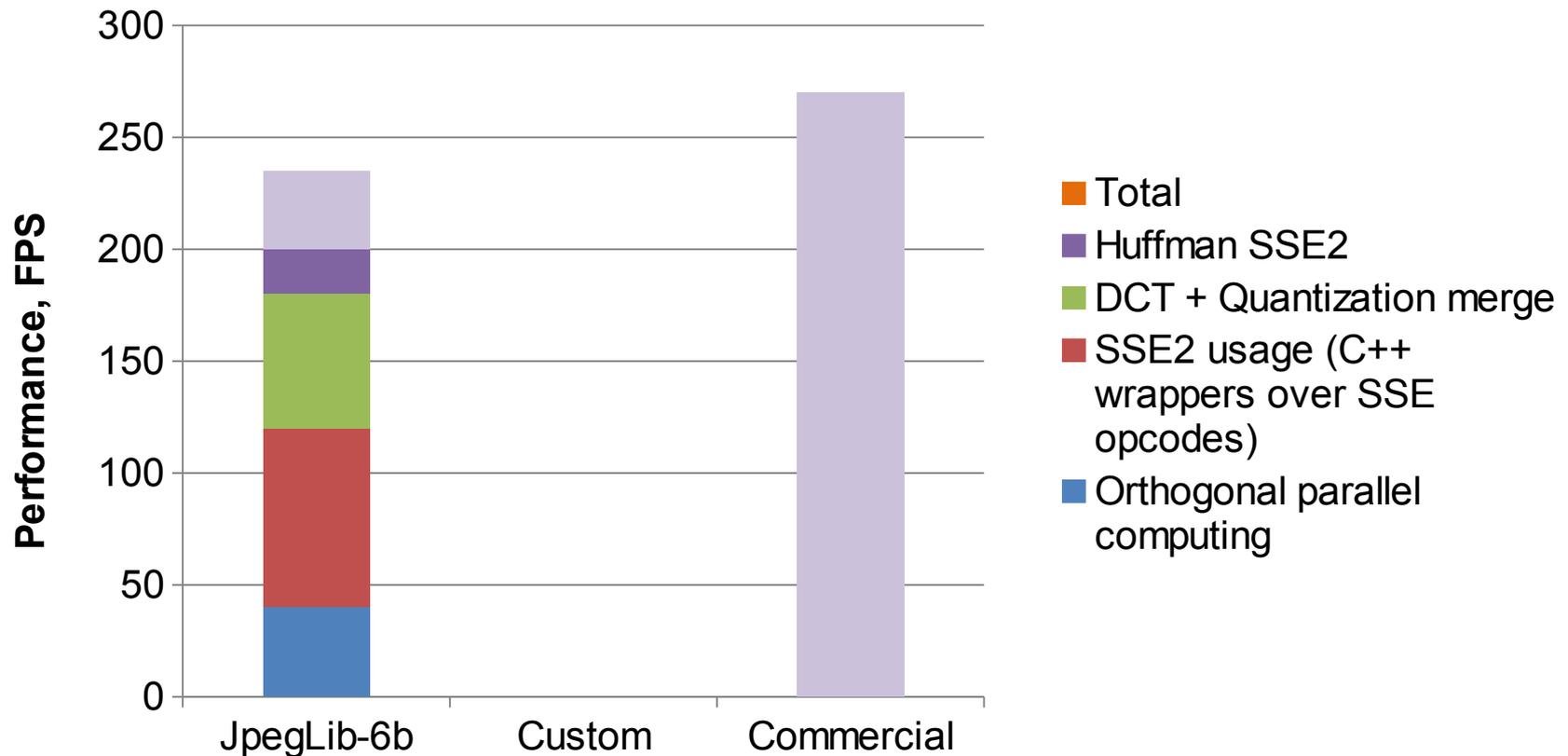
## Пример – CDMA кодек



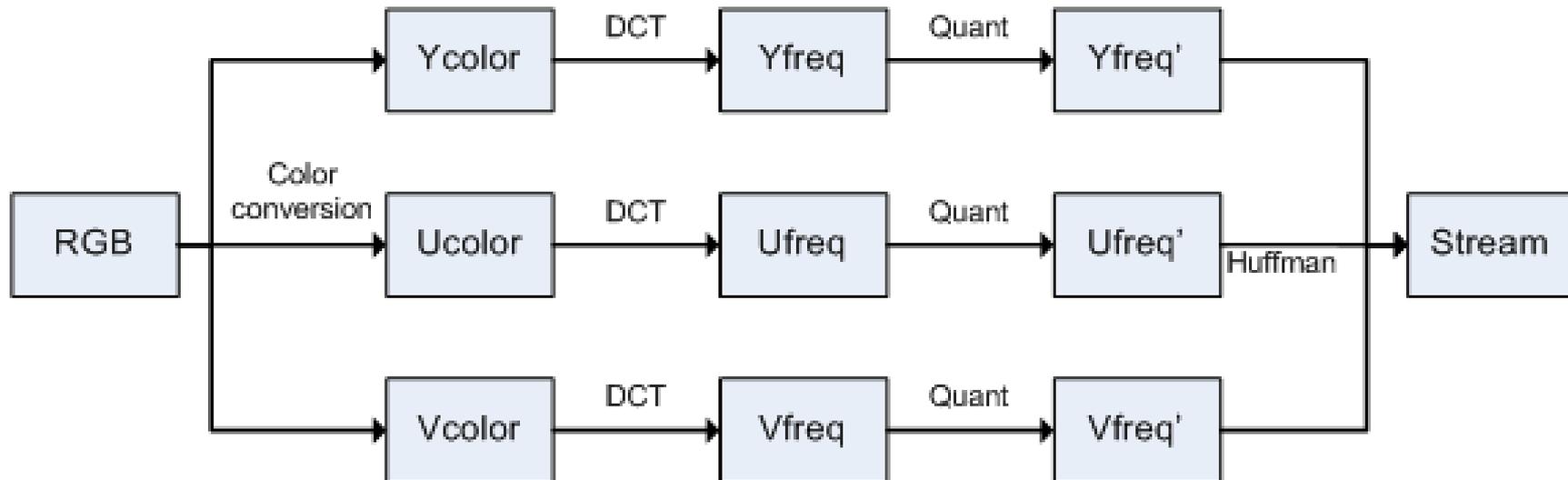
## Пример – bi-cubic scaling



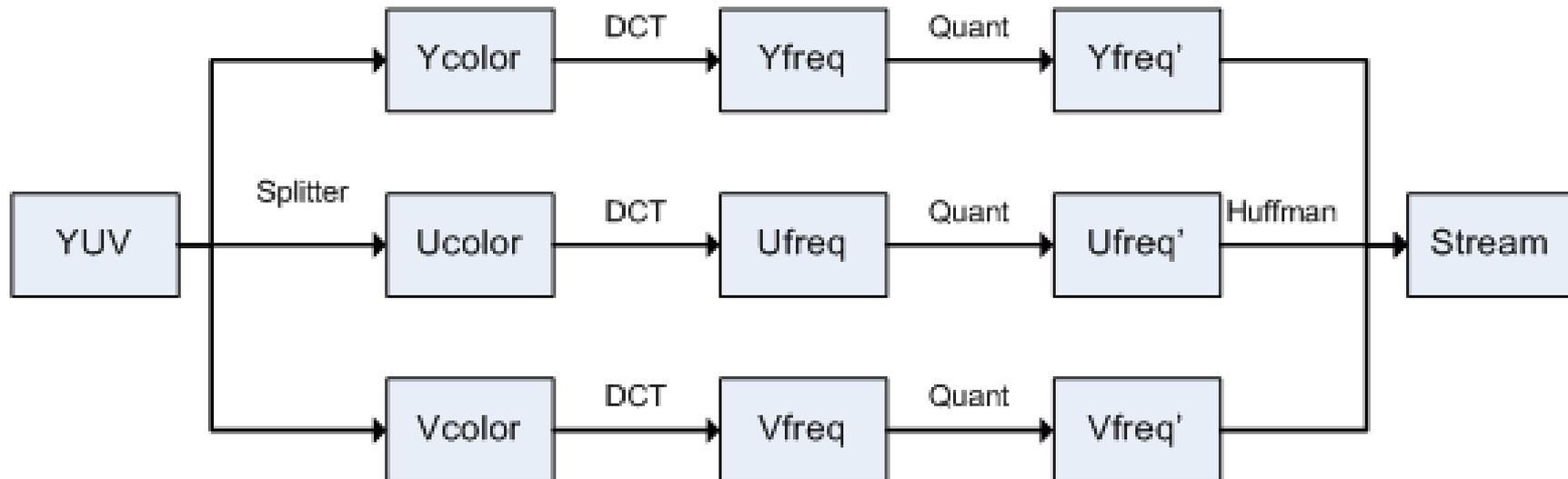
## Пример – JPEG coding



# JPEG coding - стандарт



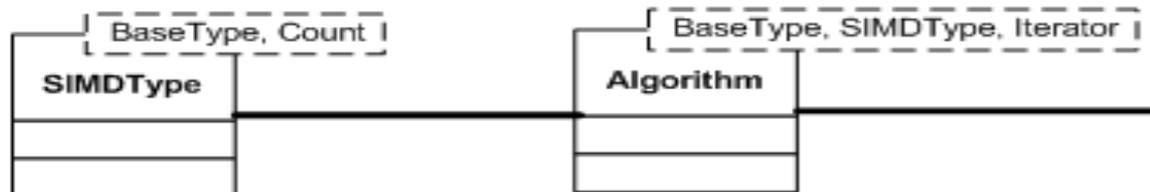
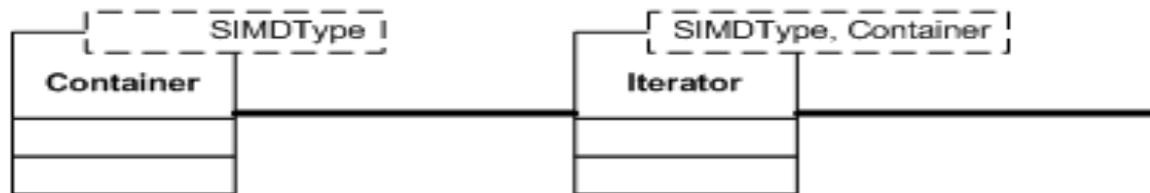
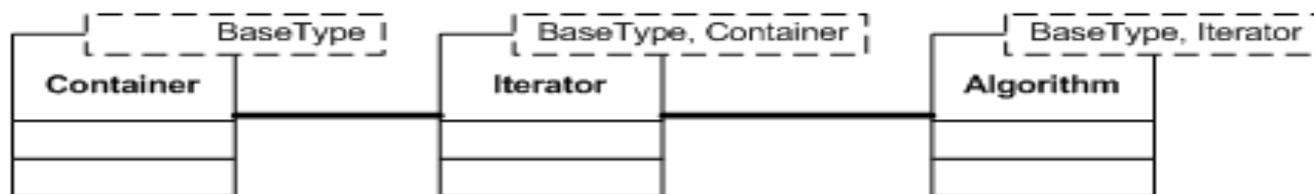
# JPEG coding - реализация



## JPEG coding - оптимизация



# SIMD-TL





## Q&A

### **Санкт-Петербург**

Адрес: 194044, Россия, Санкт-Петербург,  
Гельсингфорсская ул., д. 2  
Телефон: +7 (812) 494-90-90  
e-mail: [market@AstroSoft.ru](mailto:market@AstroSoft.ru)

### **Москва**

Адрес: 115162, Москва, ул. Хавская, д.11,  
(ст.м. Шаболовская)  
Телефон: (495) 664-21-61  
e-mail: [market@AstroSoft.ru](mailto:market@AstroSoft.ru)