ГУАП Государственный университет аэрокосмического приборостроения

www.guap.ru

Технология и инструменты разработки ФПО для встраиваемых многопроцессорных систем и распределенных комплексов ИМА

Юрий Евгеньевич Шейнин, директор института Высокопроизводительных Компьютерных и Сетевых Технологий

Алексей Юрьевич Сыщиков, заведующий лабораторией программных систем

Борис Николаевич Седов, научный сотрудник

VIPE www.vipetech.ru

Проектирование ФПО для КБО ИМА

Проектирование архитектуры системы

Разработка ФПО

Оценка характеристик ФПО

Генерация кода ФПО

Размещение ФПО на вычислителях КБО

Прототипирование, интеграционное тестирование, сопровождение, модернизация ФПО

Аспекты разработки ФПО

1. ФПО должно реализовывать функции различных предметных областей

специалисты предметной области А

специалисты предметной области В

специалисты предметной области С

Аспекты разработки ФПО

2. Разработка ФПО – комплексный процесс, где каждому специалисту требуется свой уровень абстракции

специалисты предметной области А

системные архитекторы

специалисты предметной области В

+

алгоритмисты

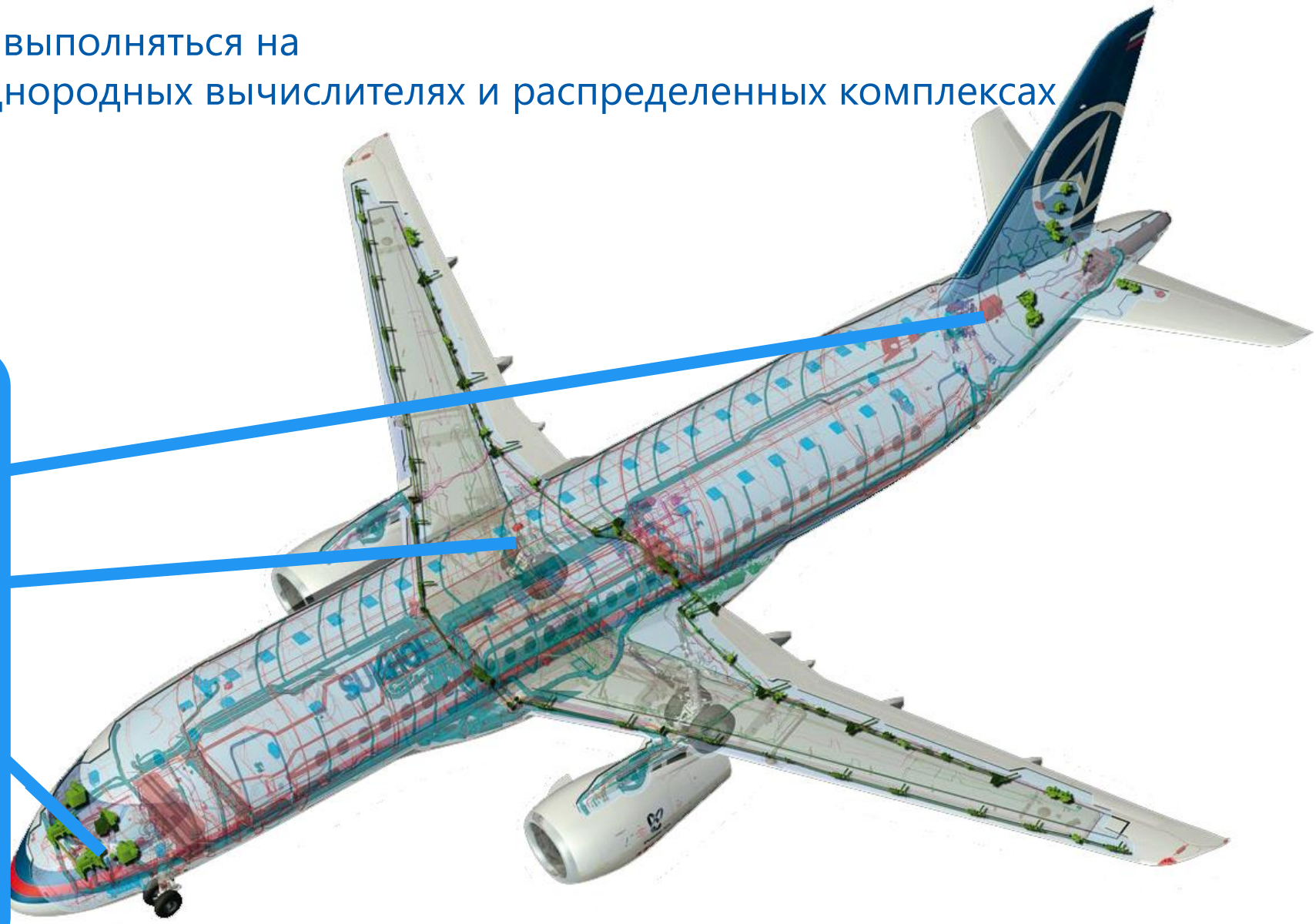
специалисты предметной области С

программисты

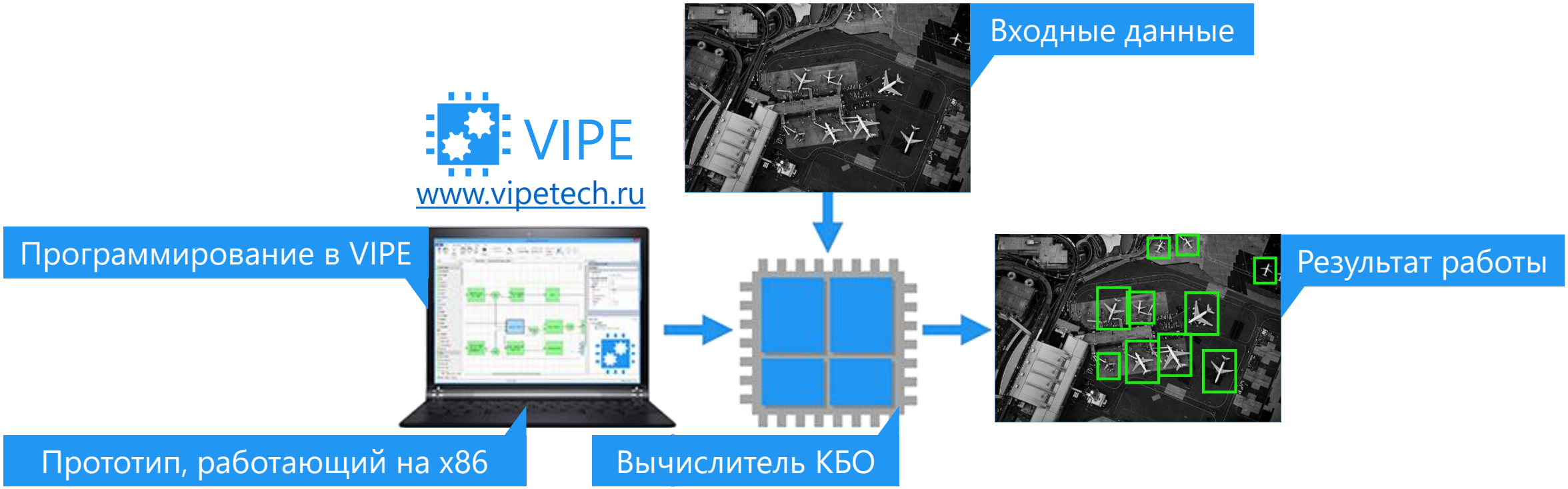
Аспекты разработки ФПО

3. ФПО должно эффективно выполняться на многопроцессорных неоднородных вычислителях и распределенных комплексах

- многопоточное
- параллельное
- распределенное



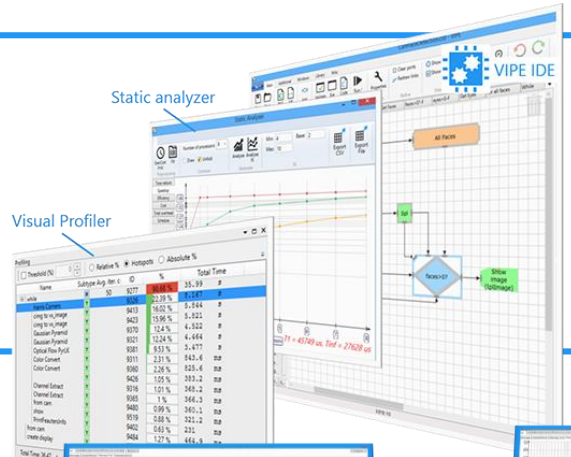
Визуальная интегрированная среда разработки ПО VIPE для встраиваемых и распределенных систем



- Быстро создать прототип ФПО
- Разработать параллельную программу, эффективно выполняющуюся на неоднородном вычислителе КБО
- Подготовить размещение ФПО на вычислители КБО
- Повторно использовать разработанную программу для других конфигураций КБО
- Обеспечить портируемость ФПО на новые вычислительные платформы
- Существенно сократить время разработки ФПО

Стадии создания ФПО в VIPE

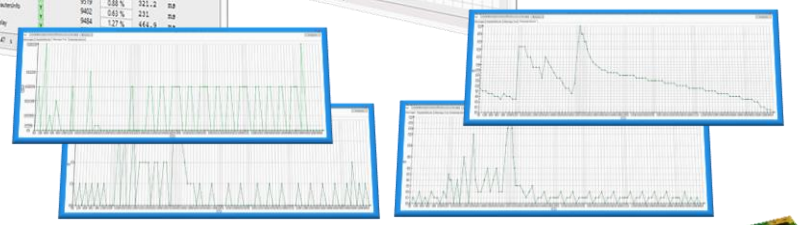
1 Разработка



Графическая среда программирования

- Проектирование алгоритмов как систем взаимодействующих процессов
- Разработка и отладка ФПО

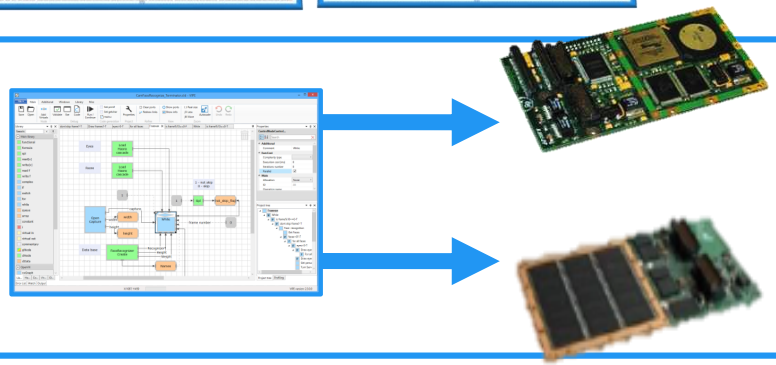
2 Оценка характеристик



Анализатор и симулятор

- Анализ свойств программы
- Симуляция на модели платформы

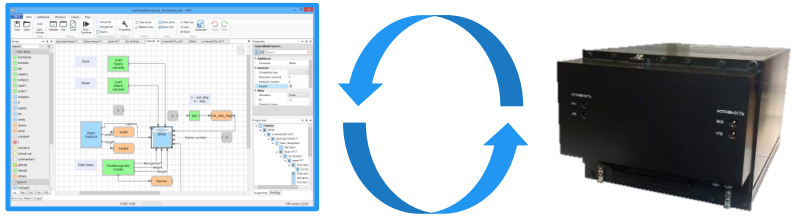
3 Размещение на вычислителях КБО



Генератор кода
+ внешние инструменты

- Генерация кода
- Вызов компилятора
- Загрузка в платформу

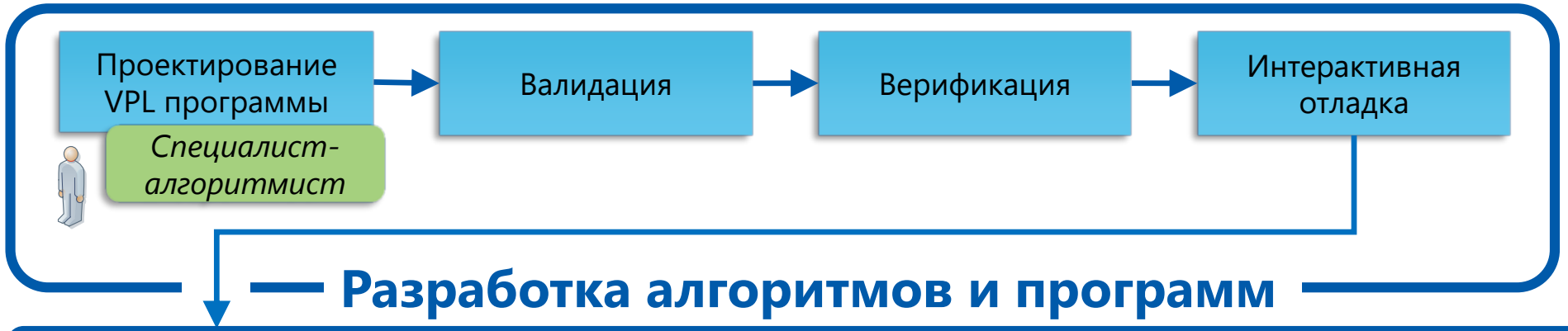
4 Отладка на платформе



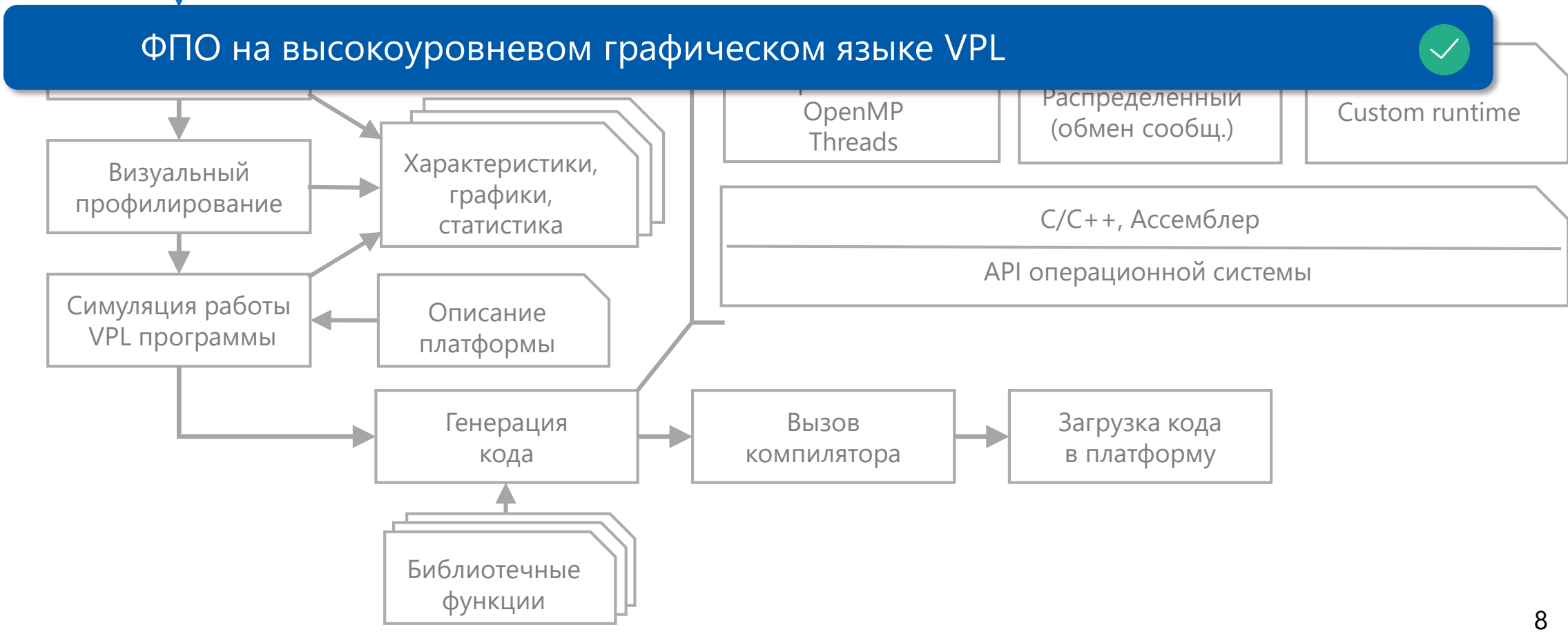
Профилировщики, отладчики…

- Получение характеристик работы ФПО на платформе
- Анализ работы ФПО на платформе
- Отладка ФПО в среде

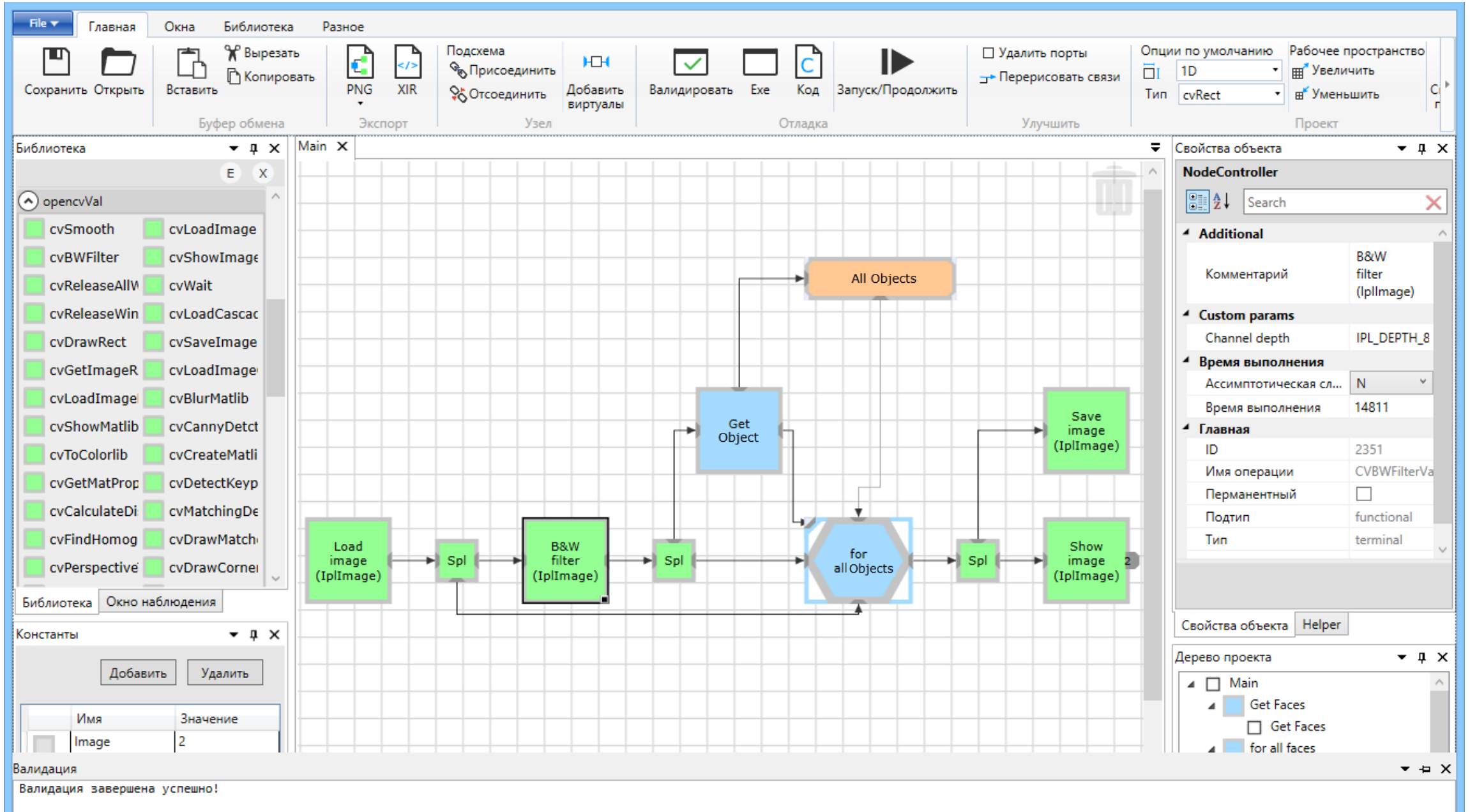
Поток разработки встраиваемого ПО в VIPE



Разработка алгоритмов и программ

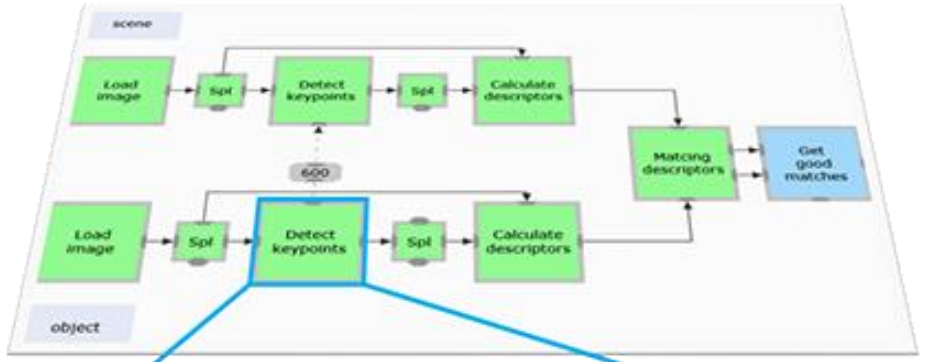


Визуальная среда программирования

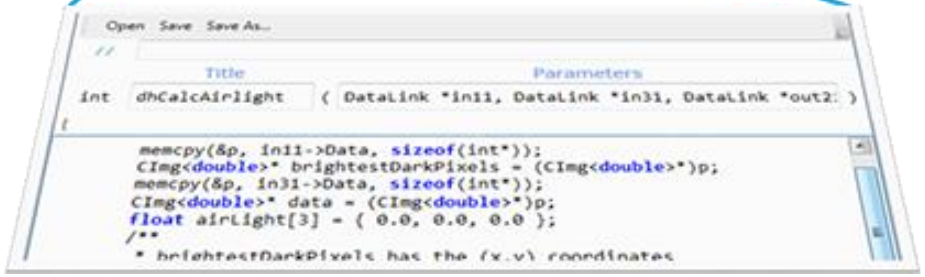


Проектирование в VIPE

Уровень предметной области



Уровень программной реализации

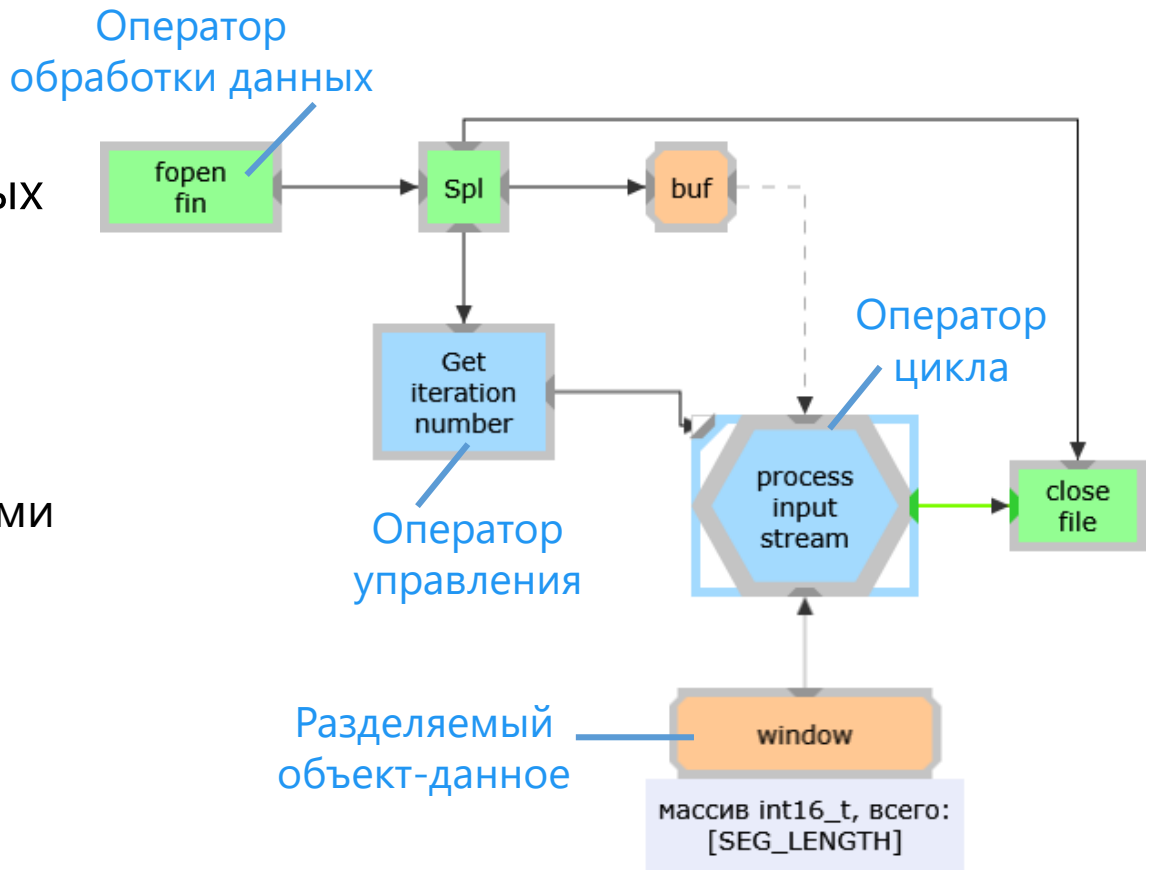


- Разработка параллельной ФПО и программирование функций обработки отделены друг от друга
- Все управление параллельными вычислениями происходит только на уровне схемы параллельной программы
- Меньше вероятность ошибок без потери читаемости программы
- Просто вносить изменения в структуру параллельной программы

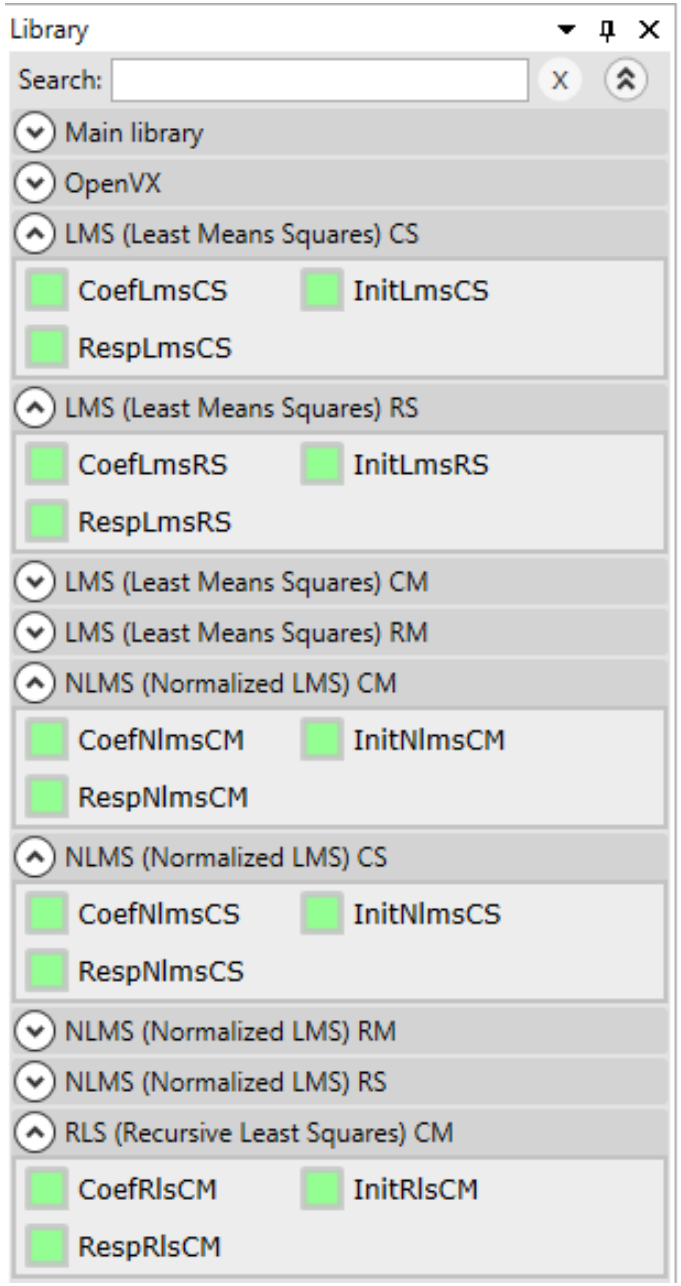
У каждого специалиста свой уровень абстракции

Графический язык программирования VPL

- Программа на языке VPL – параллельная программа из операторов обработки, операторов управления и разделяемых данных
- Визуальный подход обеспечивает
  - прозрачность процесса разработки
  - прослеживаемость структуры зависимостей
  - наглядность структуры управления вычислениями
- Общие и распределенные данные
- Программные конструкции построения иерархической схемы программы
- Условные операторы, операторы циклов
- Естественный параллелизм и потенциальная конвейеризация



Предметно-ориентированная разработка



Предметно ориентированные библиотеки

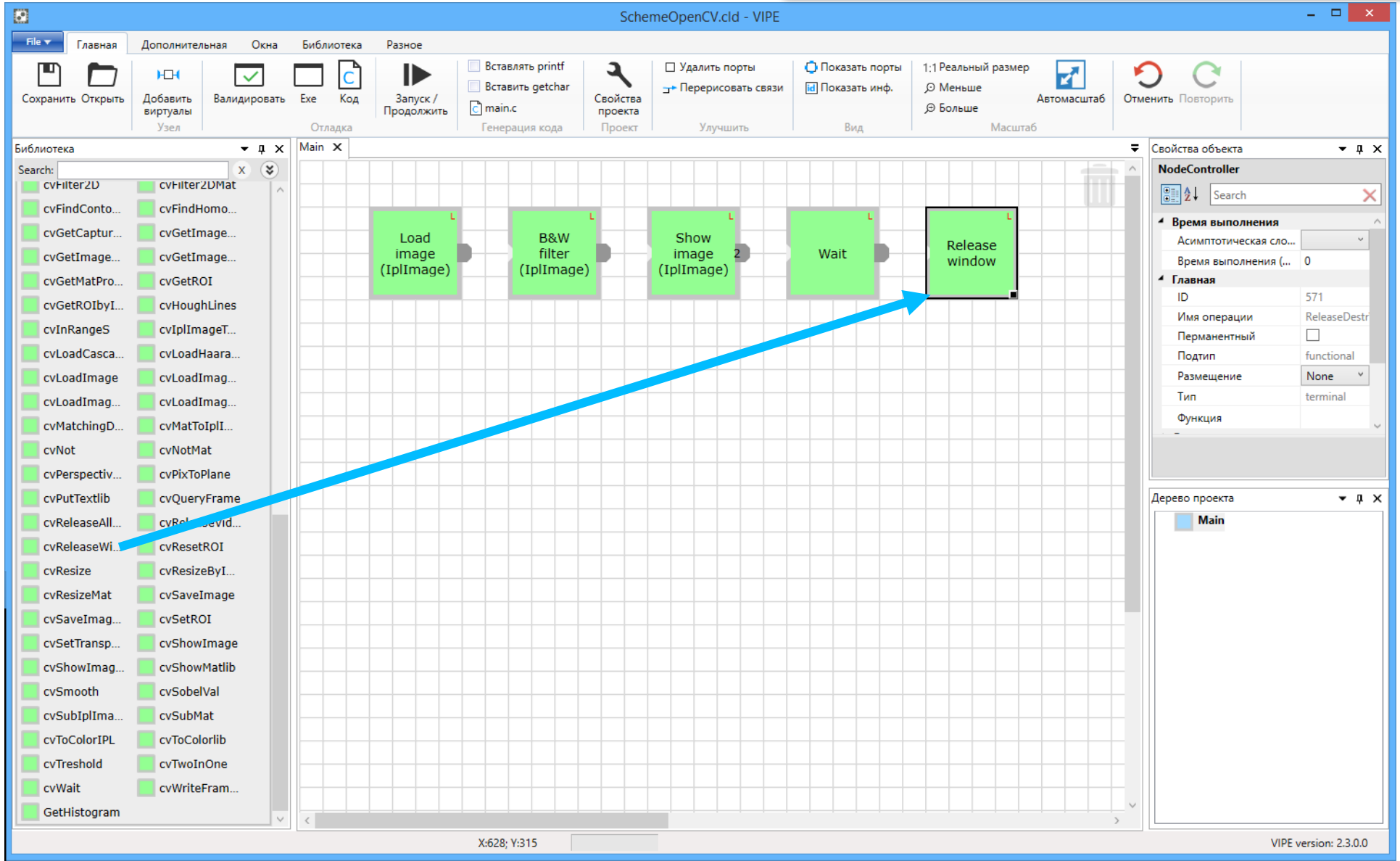
VIPE позволяет интегрировать в среду разработки программные предметно ориентированные библиотеки.

Интегрированные библиотеки VIPE:

- Ускоряют разработку / получение прототипа ФПО
- Включают основные типовые и часто используемые функции предметной области
- Расширяются пользователем
- Обеспечивают повторное использование программных модулей
- Повышают точность оценки характеристик ФПО

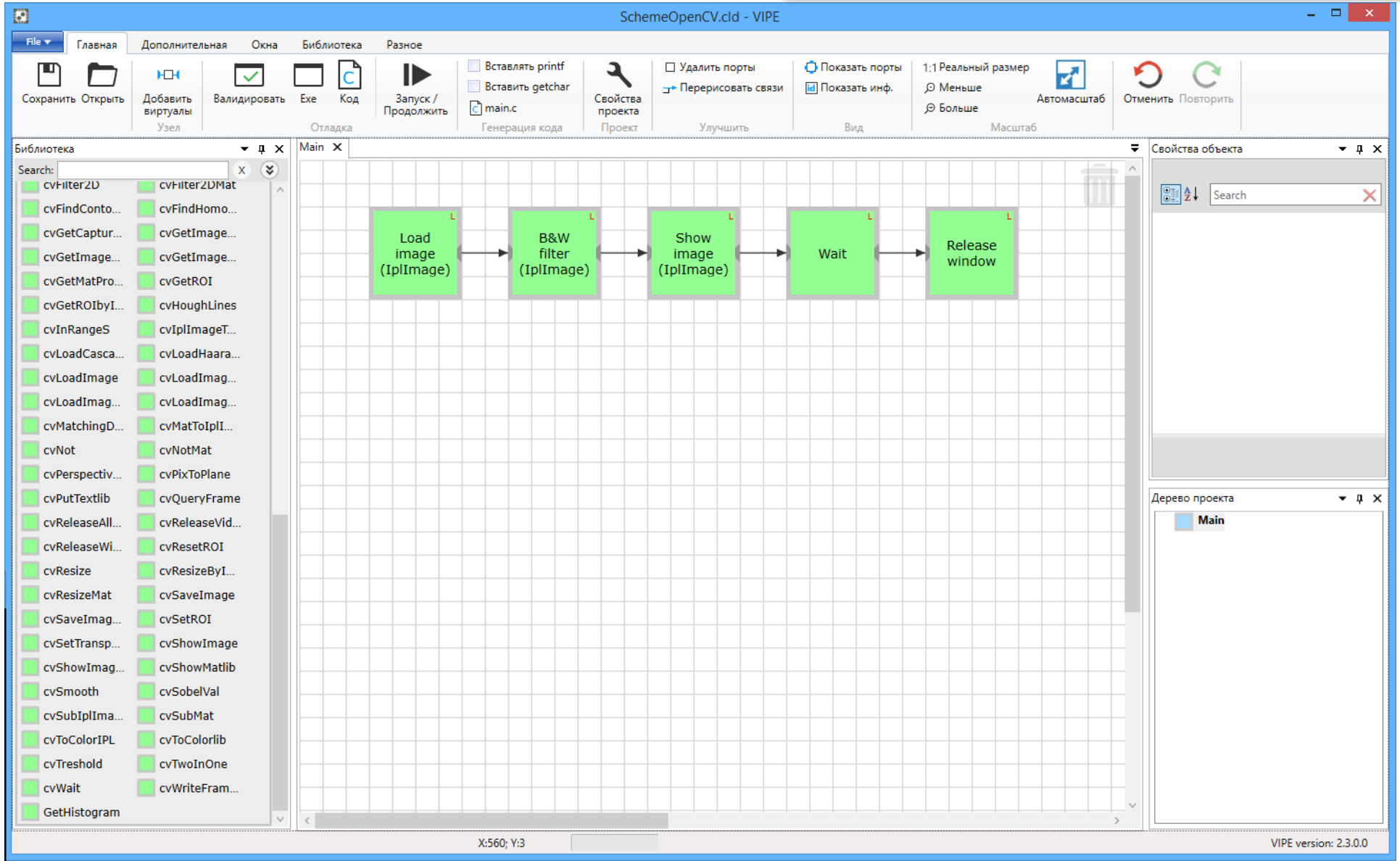
Создание ПО в VIPE

Прототипирование и разработка ФПО



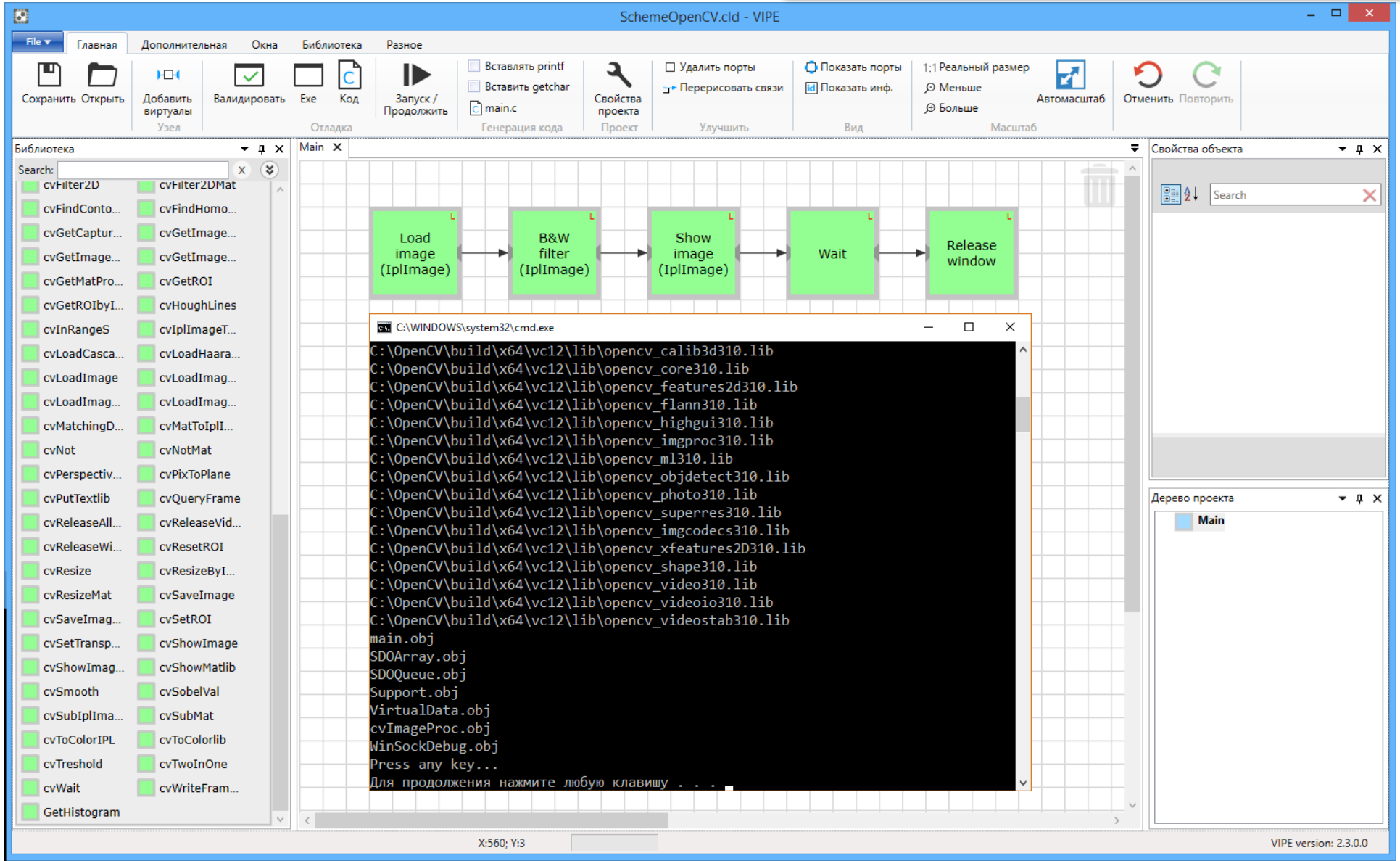
Создание ПО в VIPE

Прототипирование и разработка ФПО



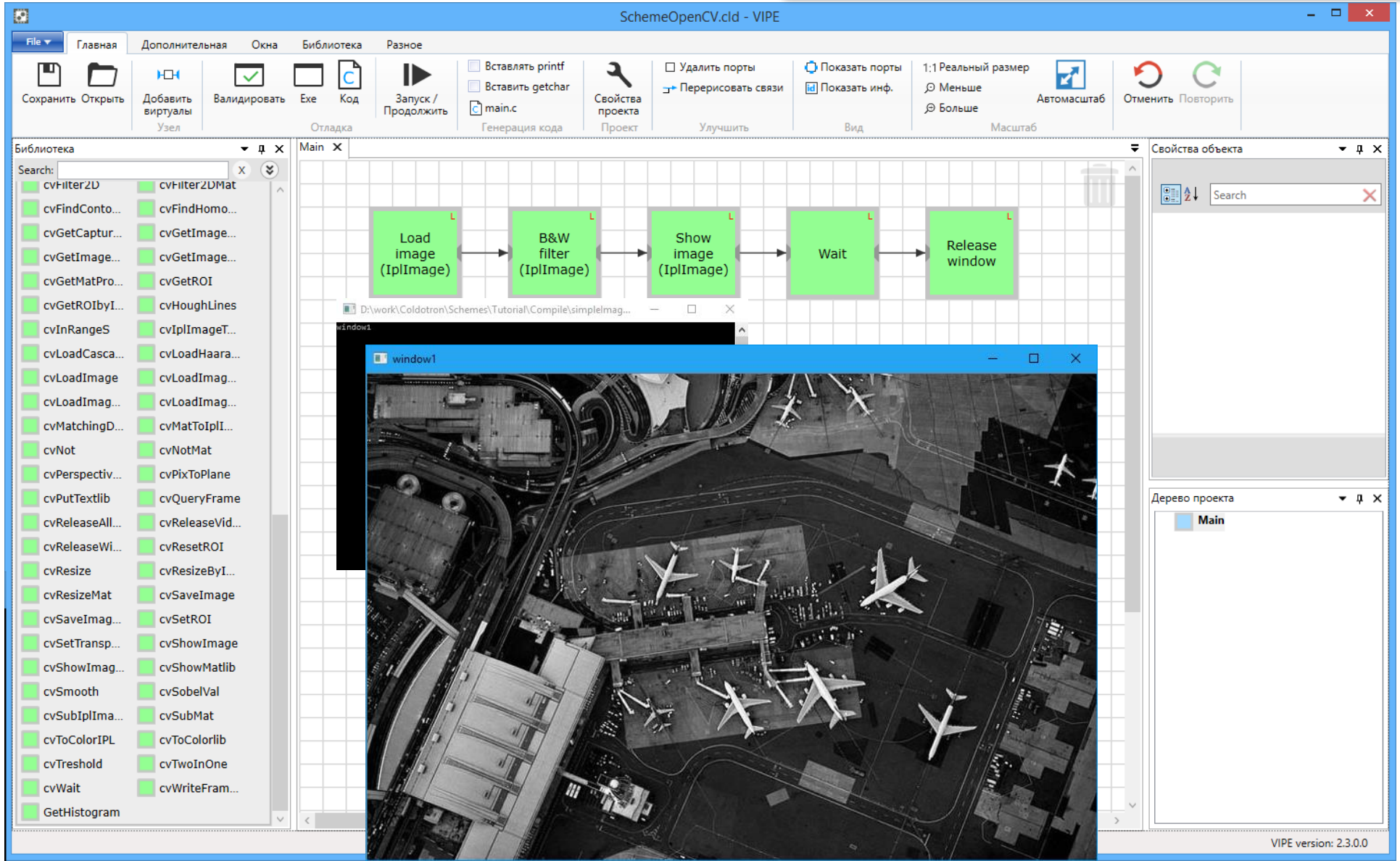
Создание ПО в VIPE

Прототипирование и разработка ФПО



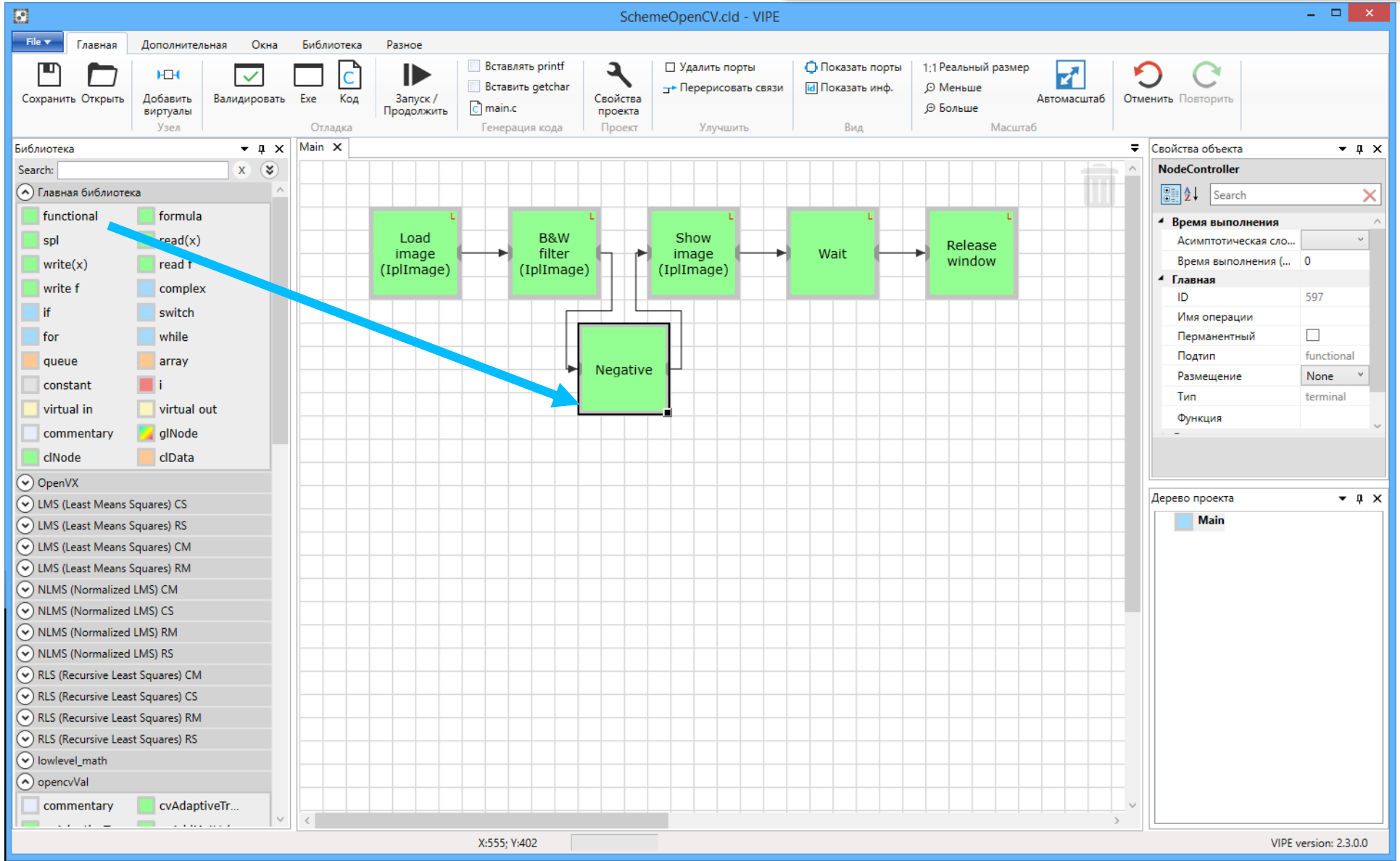
Создание ПО в VIPE

Прототипирование и разработка ФПО



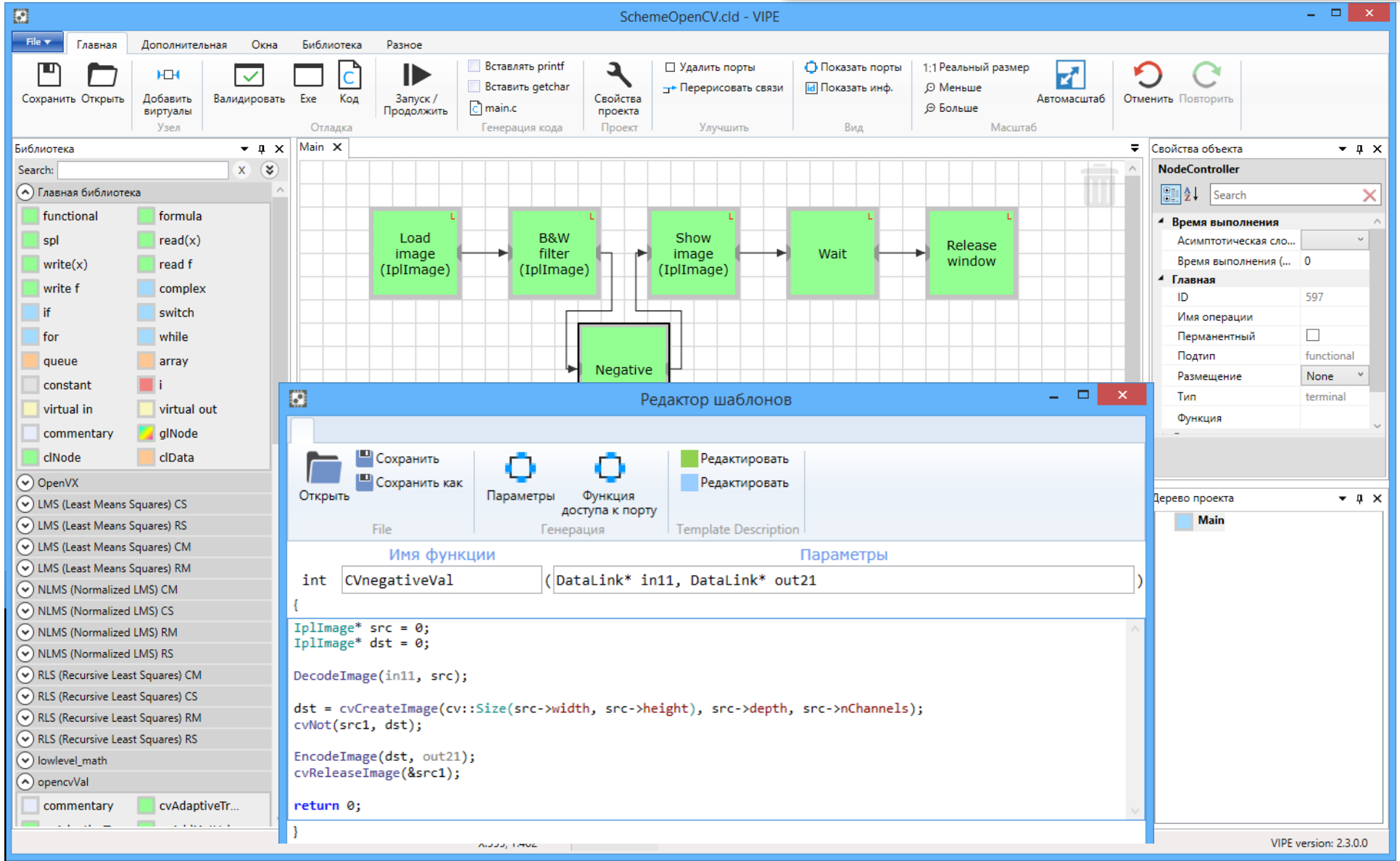
Создание ПО в VIPE

Прототипирование и разработка ФПО



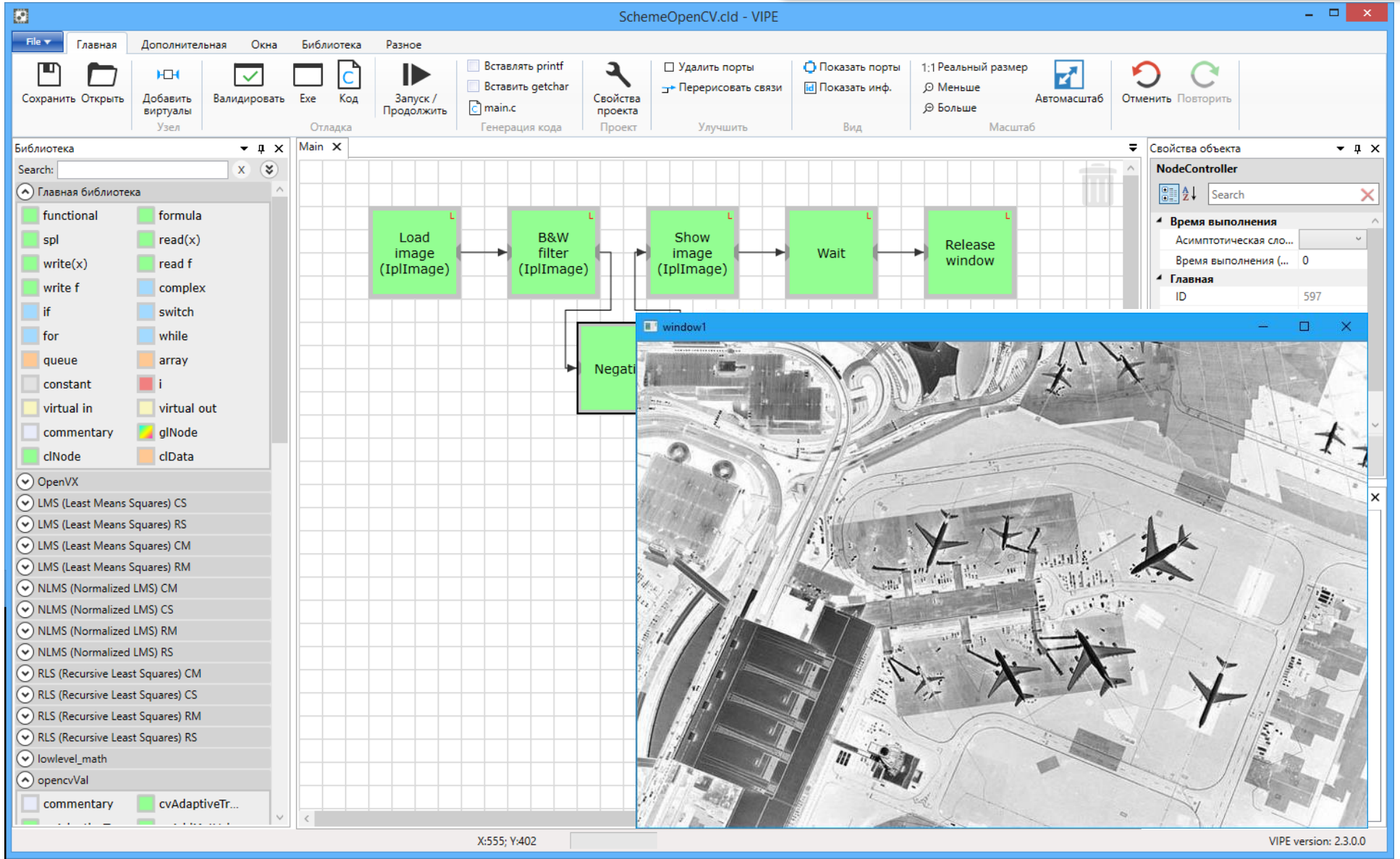
Создание ПО в VIPE

Прототипирование и разработка ФПО



Создание ПО в VIPE

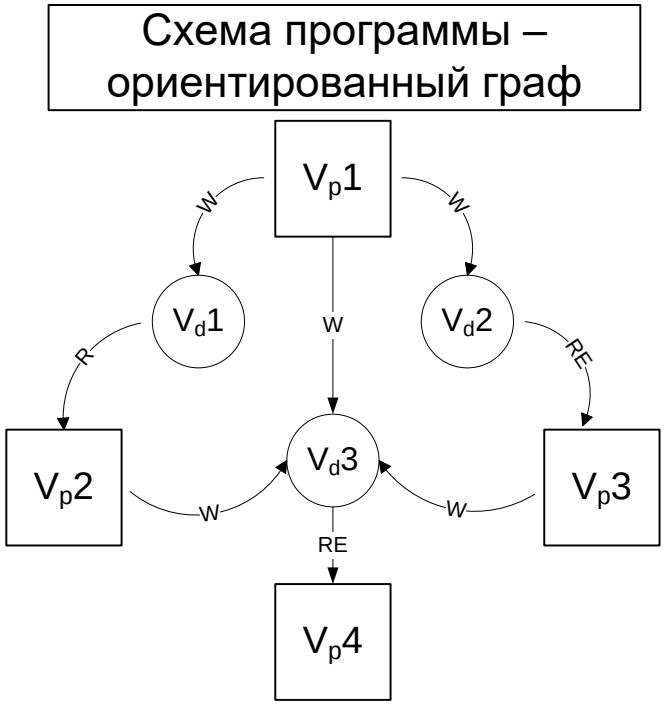
Прототипирование и разработка ФПО



Асинхронные Развивающиеся Процессы (АРП) формальная модель вычисления

АРП-модель определяет

- синтаксис языка
- семантику объектов языка
- управляющие конструкции



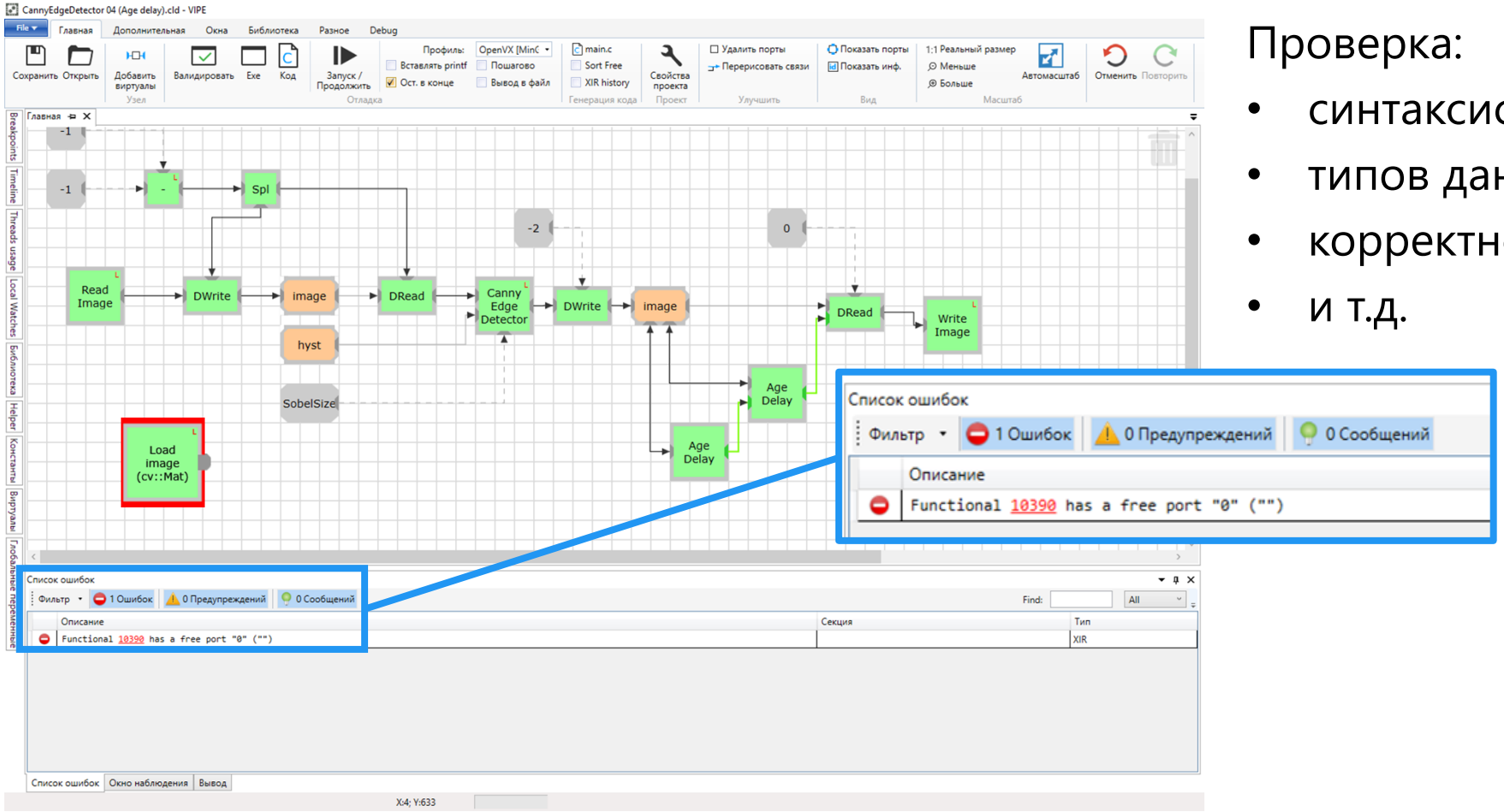
V_p – вершина-оператор,
V_d – вершина-данное,
W/R/RE – дуги (связи) и их разметка

VIPE на основе АРП-модели

- *предоставляет* методы формальной верификации
- *обеспечивает* идентичность результатов при работе в разных окружениях
- *даёт возможность* переносимости вычислений
- *даёт* представление динамики параллельного вычисления
- *позволяет* совместить работу с общими данными и моделями распределенной памяти

VPL – язык, основанный на формальной модели

- Валидация



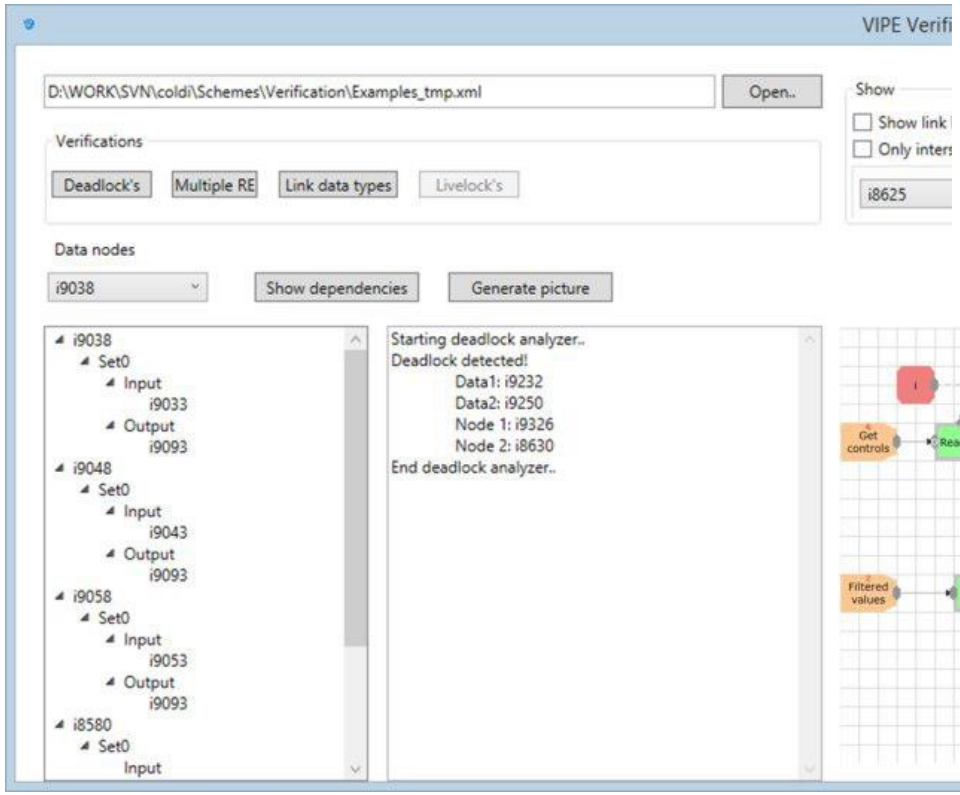
Проверка:

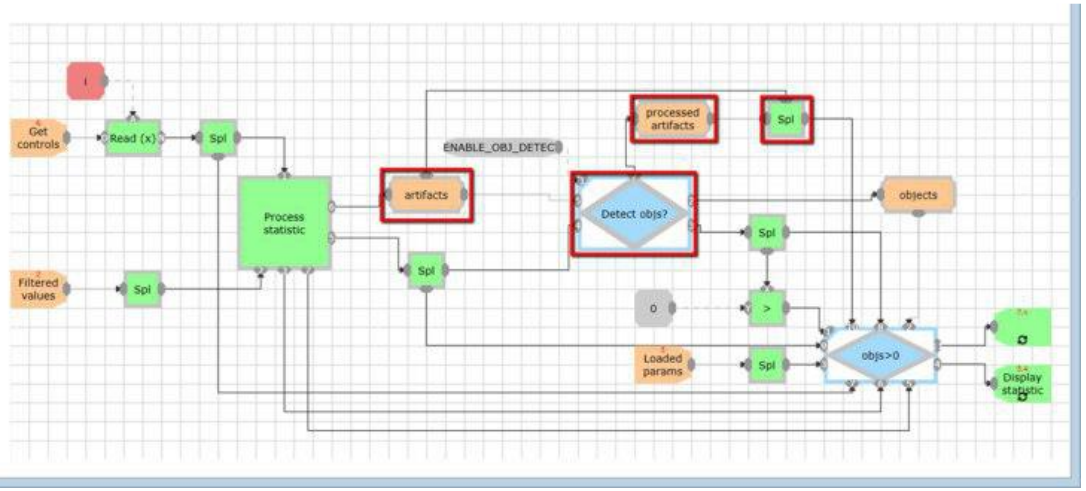
- синтаксиса
- типов данных
- корректности связей
- и т.д.

Интерактивные инструменты

- Валидация
- Верификация

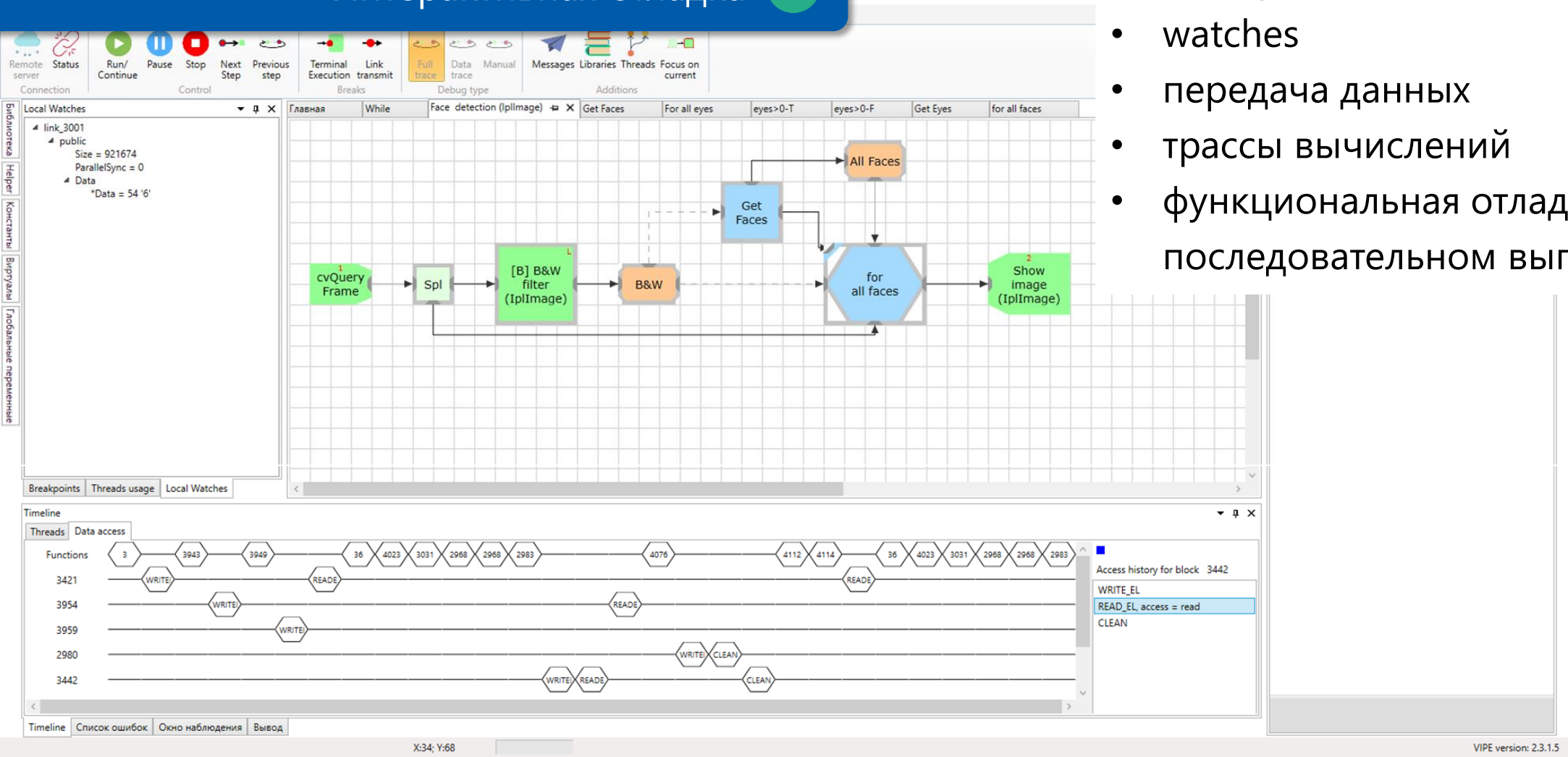
- гонки
  - однозначность результатов вычислений
  - гонки данных
- тупиковые ситуации
  - deadlocks
  - livelocks
- завершаемость





Интерактивные инструменты

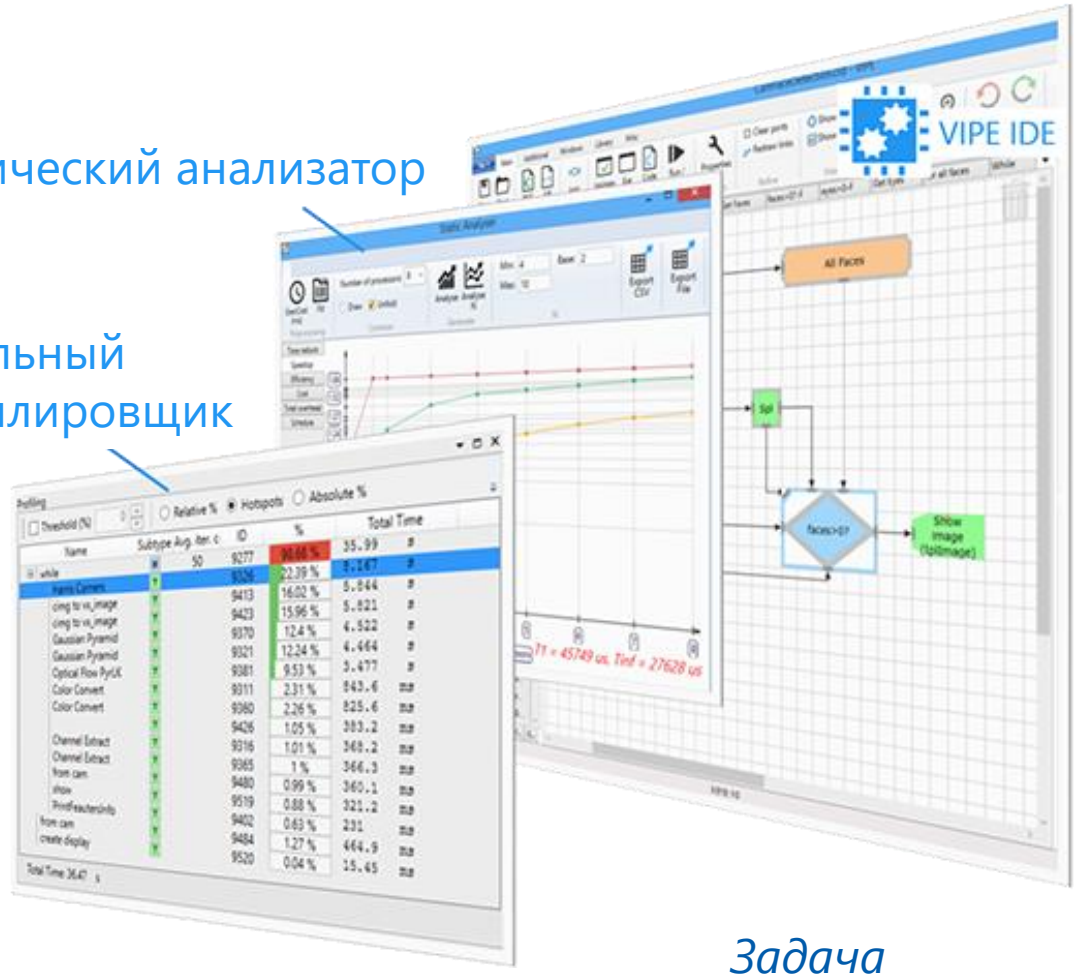
- Валидация
- Верификация
- Интерактивная отладка



- пошаговая отладка
- breakpoints
- watches
- передача данных
- трассы вычислений
- функциональная отладка при последовательном выполнении

Поток разработки встраиваемого ПО в VIPE

Использование инструментов анализа на различных стадиях готовности программ



Сопровождение процесса разработки

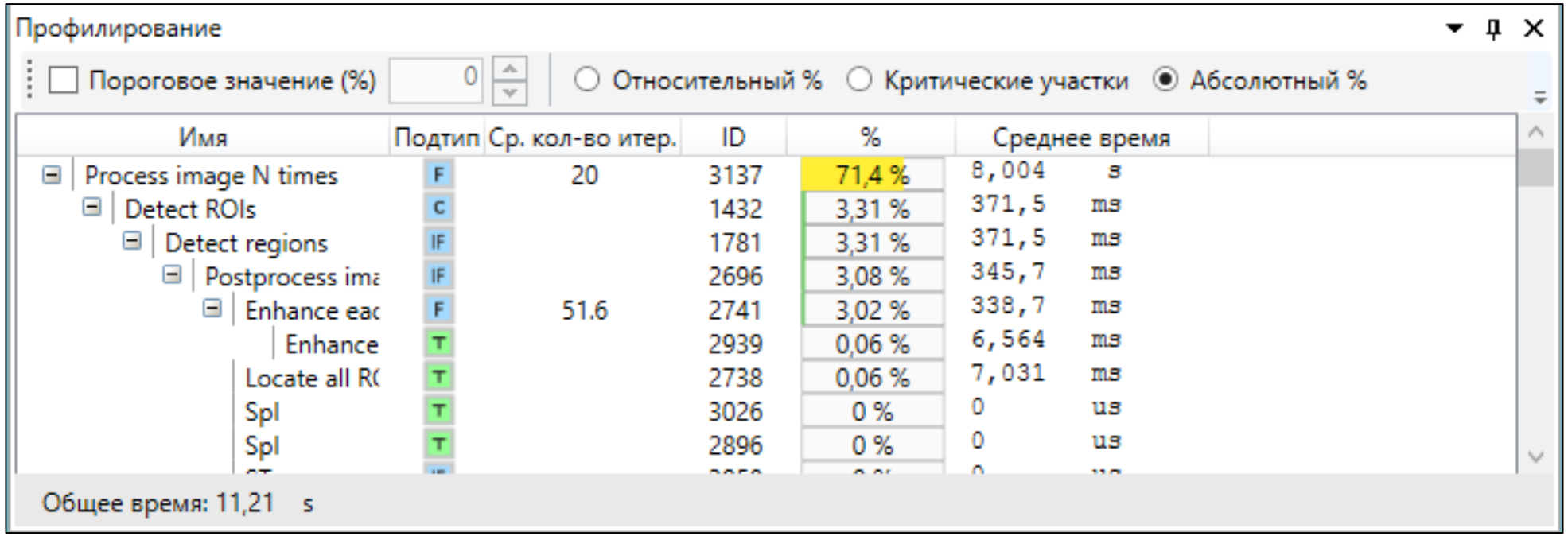
Процесс улучшения характеристик

Задача
корректировка хода проектирования для увеличения вероятности достижения требуемых характеристик к финалу разработки

Задача
добиться требуемых характеристик работы ФПО на выбранной платформе КБО

Инструмент анализа визуальный профилировщик

Поиск критических участков (hotspots) в программе



Позволяет

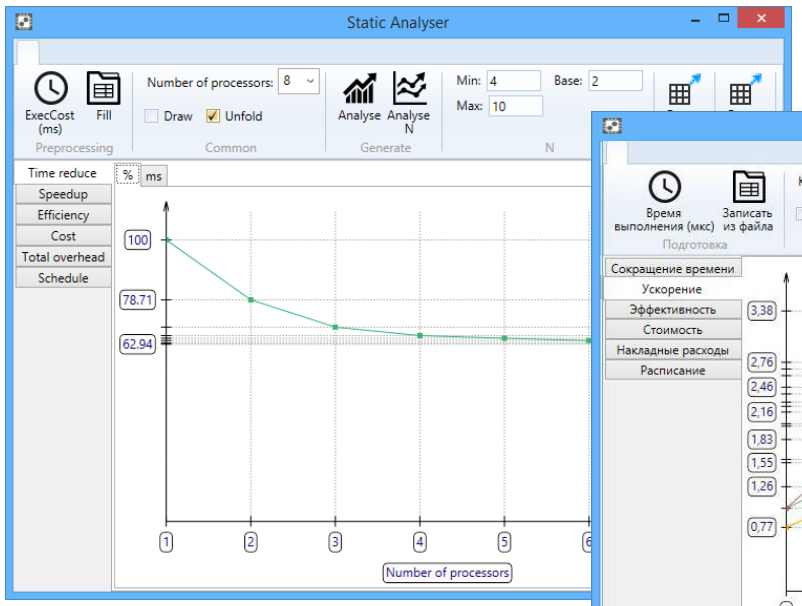
- выявить узкие места (bottlenecks)
- определить операции, требующие оптимизации или декомпозиции
- определить пути повышения скорости работы программы

Различные режимы работы

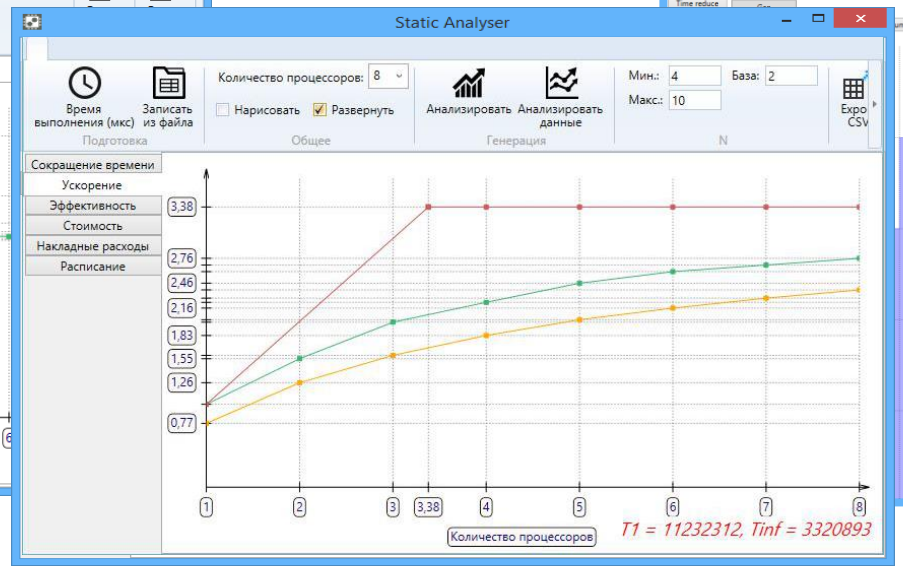
- абсолютные времена выполнения операторов схемы
- относительные времена согласно структуре вложенностей
- критические участки с фильтрацией малых величин

Инструмент статического анализа

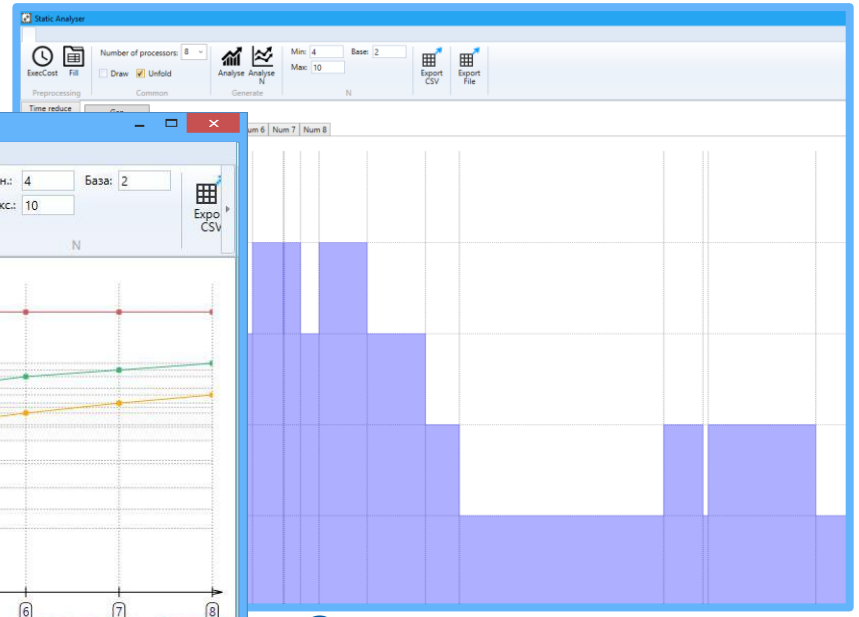
Быстрая оценка характеристик работы программы на многоядерной платформе



Время



Ускорение



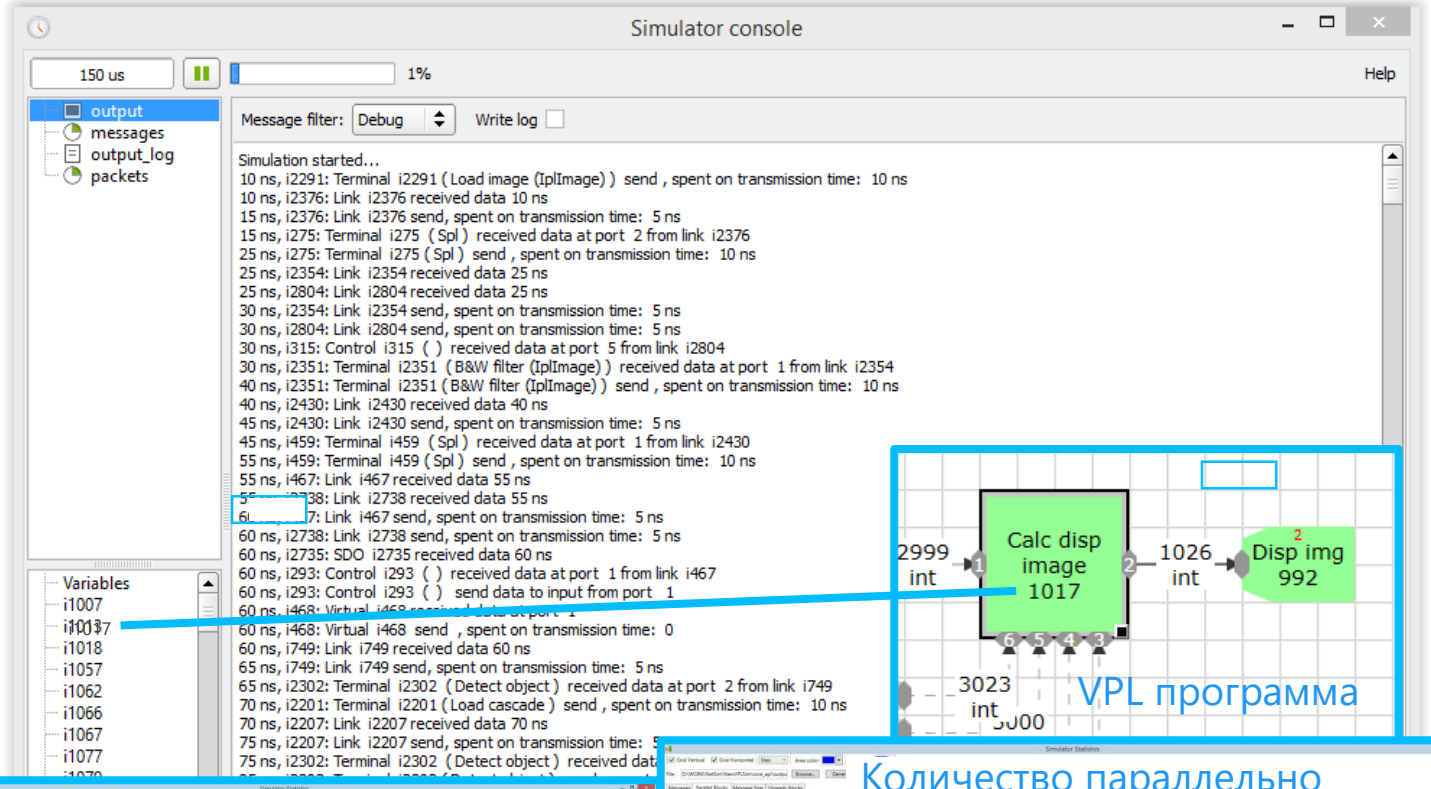
Одновременно выполняющиеся процессы

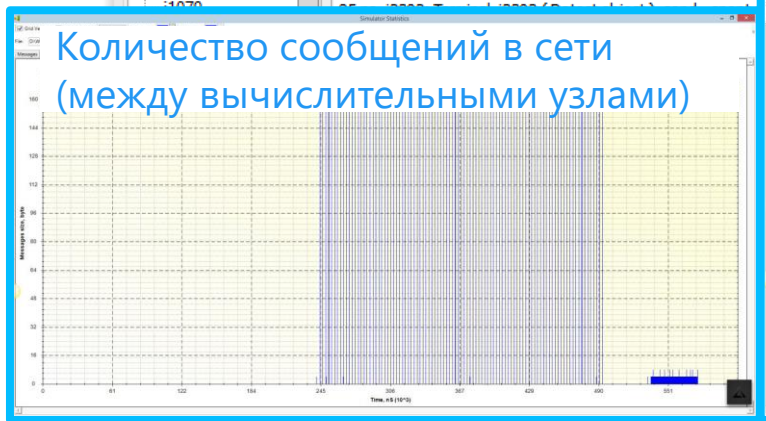
Позволяет

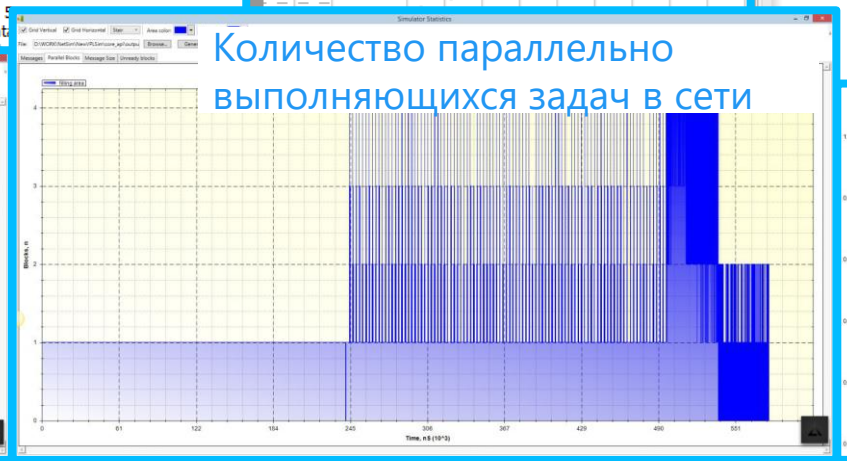
- оценить уровень параллелизма алгоритма и программы
- оценить потенциальную эффективность использования ресурсов платформы
- подготовить ПО для работы на многоядерной платформе

VPL-симулятор

Моделирование работы программы на крупноблочной модели вычислительной платформы







Позволяет оценить:

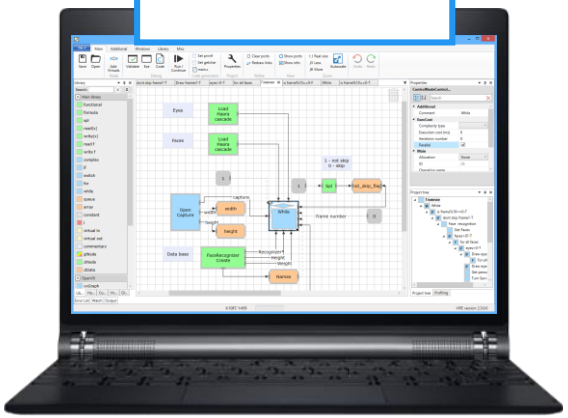
1. требования к производительности процессоров системы
2. требования к локальной памяти узлов системы
3. загруженность процессоров для различных вариантов размещения и сбалансированность загрузки
4. количество и интенсивность обменов данными
5. узкие места программы, распределения задач на аппаратной платформе

Поток разработки встраиваемого ПО в VIPE

Развертывание на целевой вычислительной платформе

Генерация кода

ФПО



Сгенерированный исходный код

Инструментальные программы, поставляющиеся с платформой

C/C++
API ОС

Распределенный (обмен сообщ.)

Параллельный OpenMP Pthreads

Custom runtime

lib

lib

lib

Компилятор

Загрузчик

ОС
ФПО

Многоядерная платформа

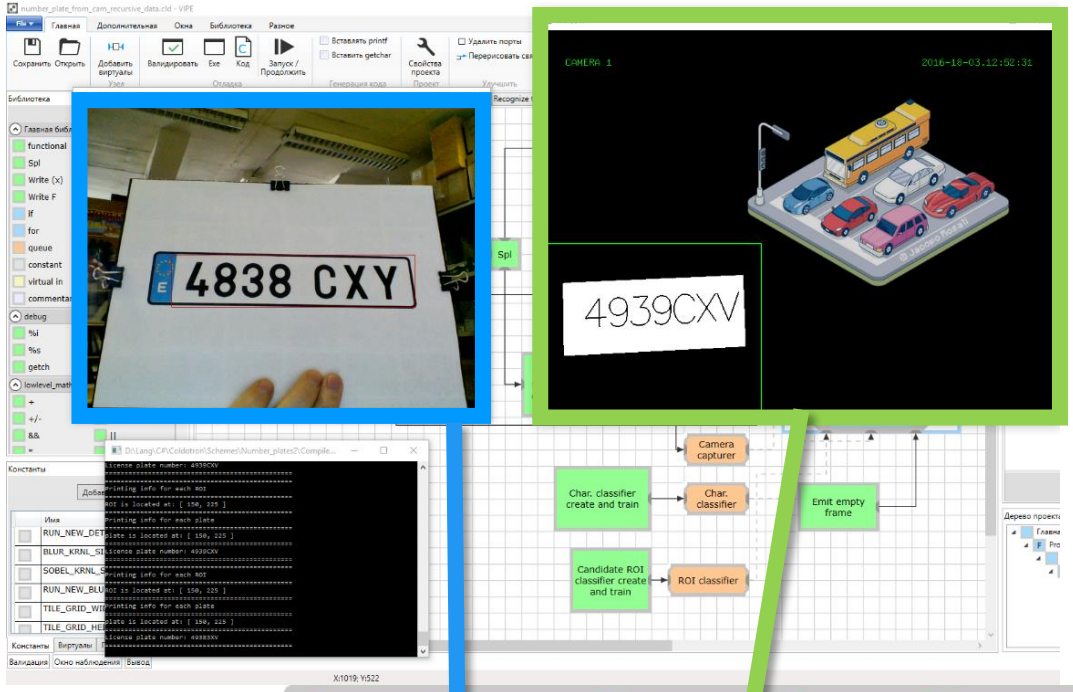
ОС
ФПО

Многопроцессорная платформа

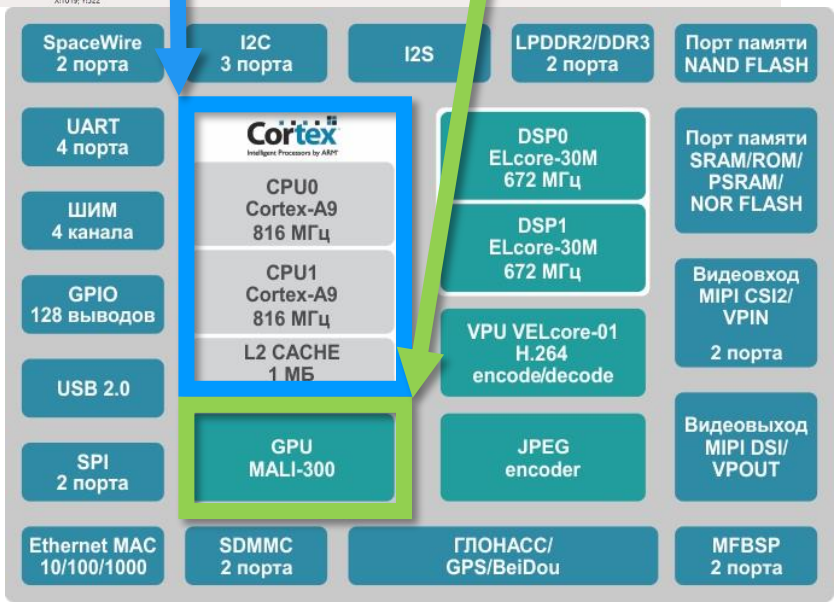
ОС
ФПО

Распределенная платформа

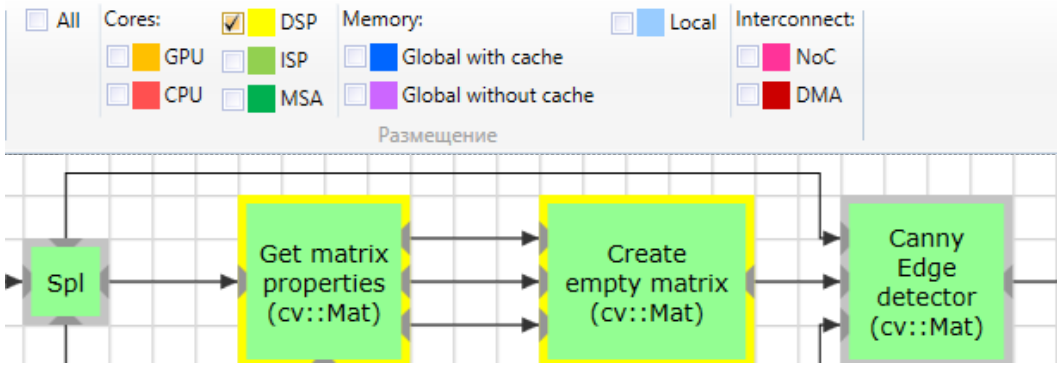
Поддержка неоднородных многоядерных систем в VIPE



Система на кристалле 1892ВМ14Я



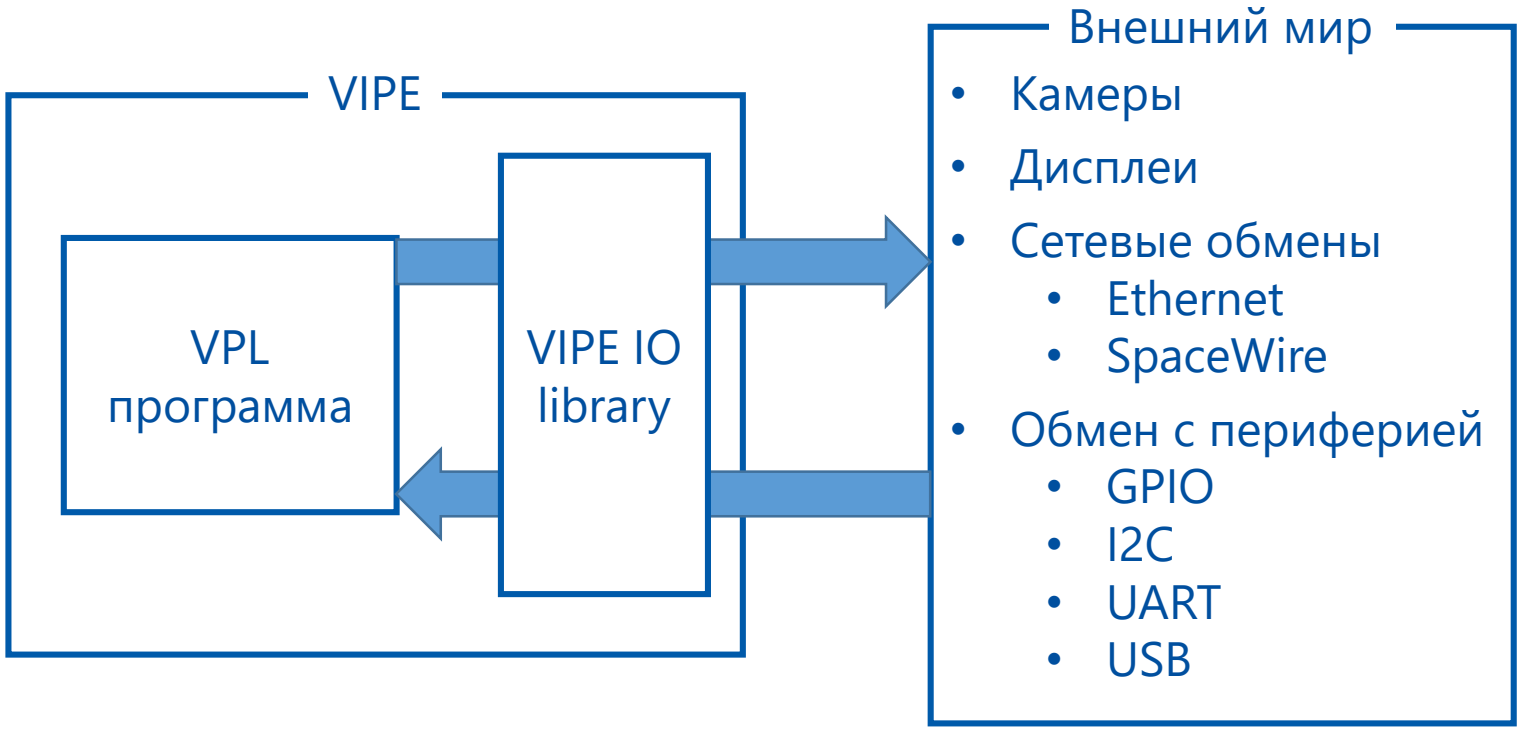
ЭЛВИС



- Размещение объектов на одном или нескольких типах ядер (CPU, GPU, DSP)
  - Операторы на разные типы ядер
  - Данные на разные типы памяти
  - Обмены данными на разные типы каналов
- Выбор реализации для операторов обработки данных
- Подготовка начальных данных и результатов работы операторов программы, принимая во внимание особенности различных механизмов связей

VIPE IO library

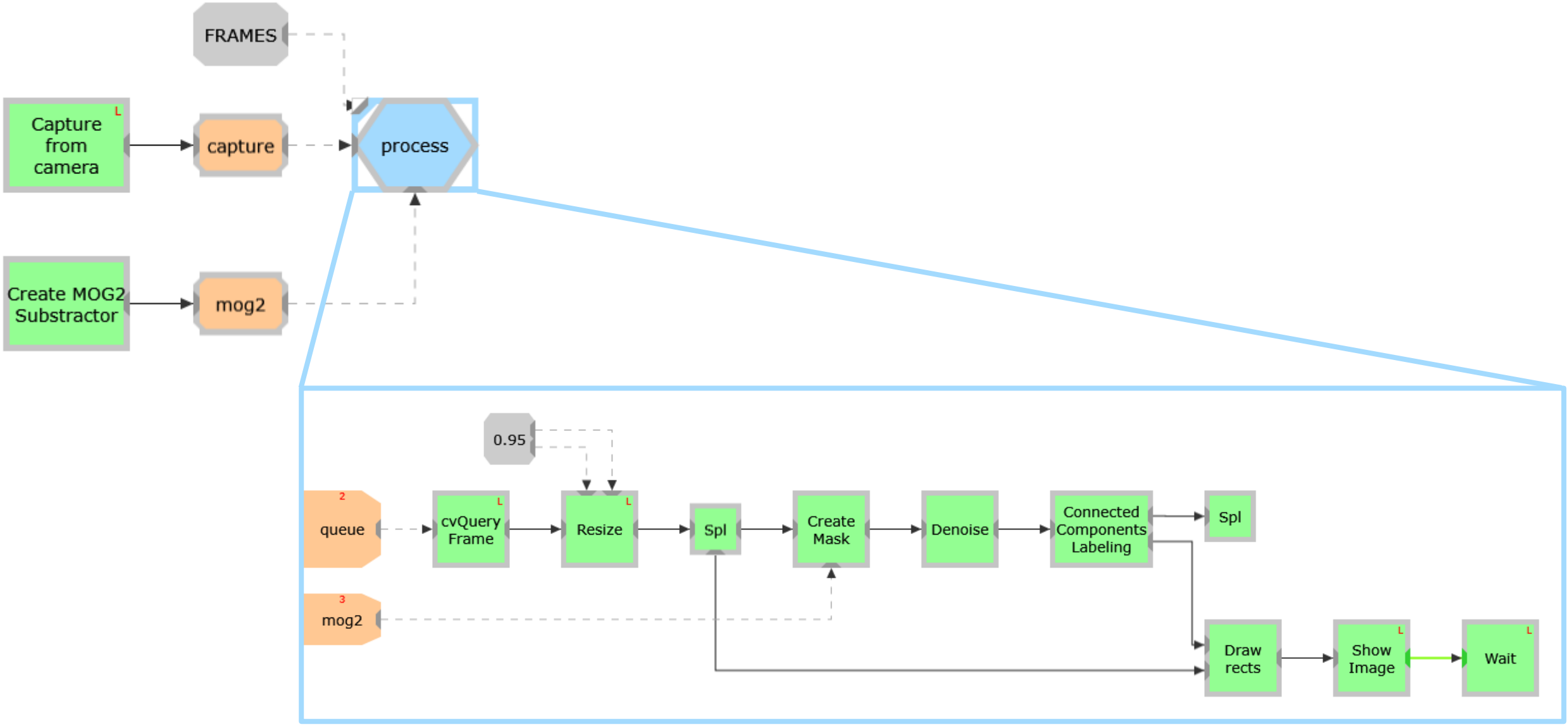
Предназначена для взаимодействия VPL-программ с внешним миром.



VIPE IO library

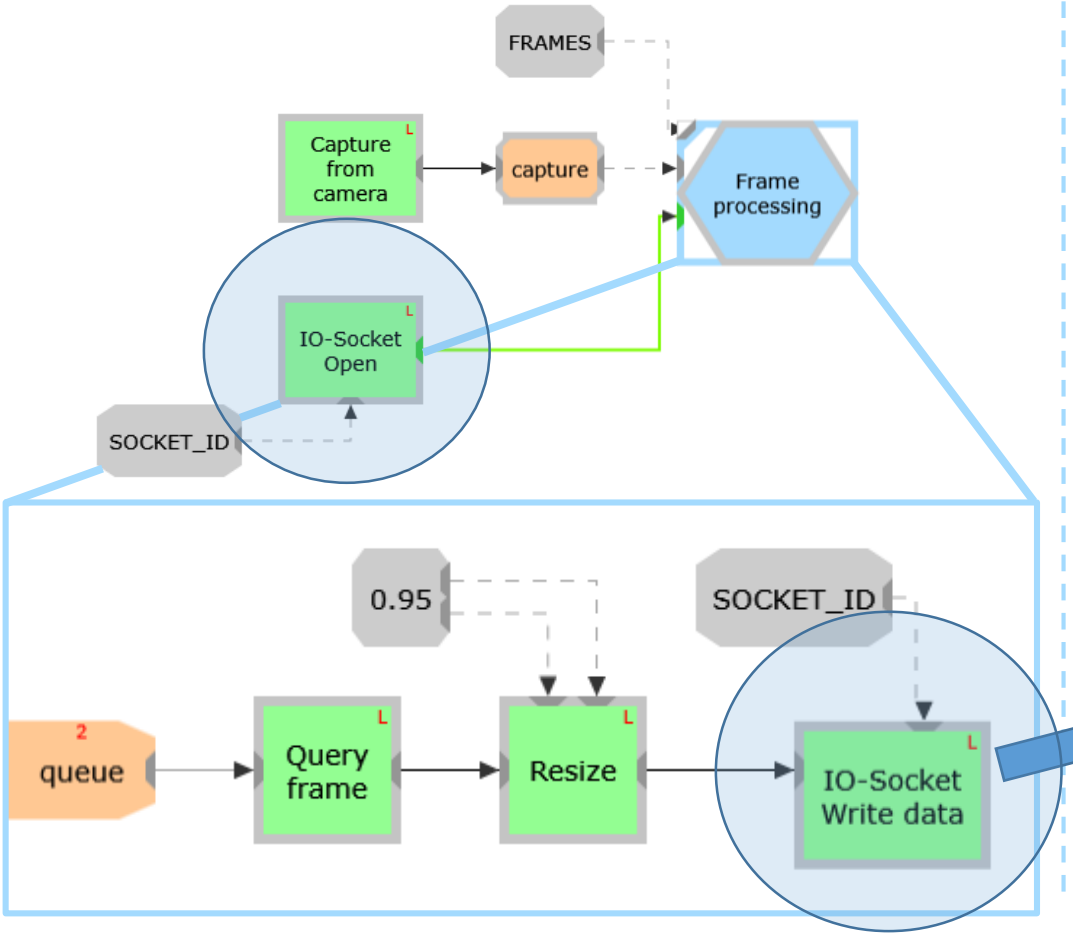
- единый принцип взаимодействия с внешними устройствами
- визуальная библиотека работы с внешними устройствами
  - считывание и вывод видео
    - USB, MIPI, HDMI
  - работа с периферийными интерфейсами
    - Serial, I2C
  - сетевое взаимодействие
    - Ethernet, SpaceWire
- программная библиотека реализаций для разных платформ
- генерация исполняемого кода для запуска на разных платформах

Пример монолитной программы обработки видео-потока

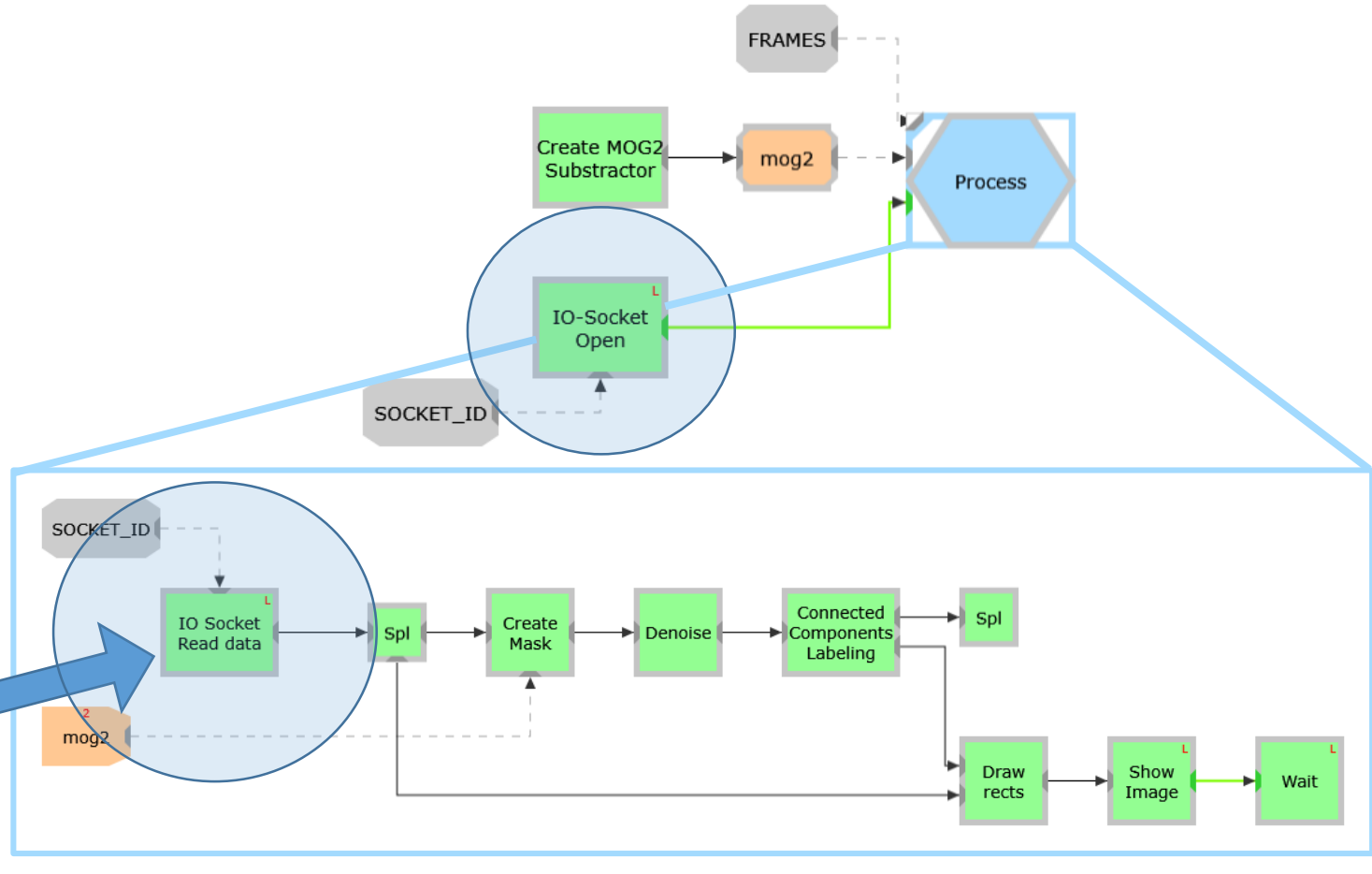


Распределенная обработка для ИМА-системы

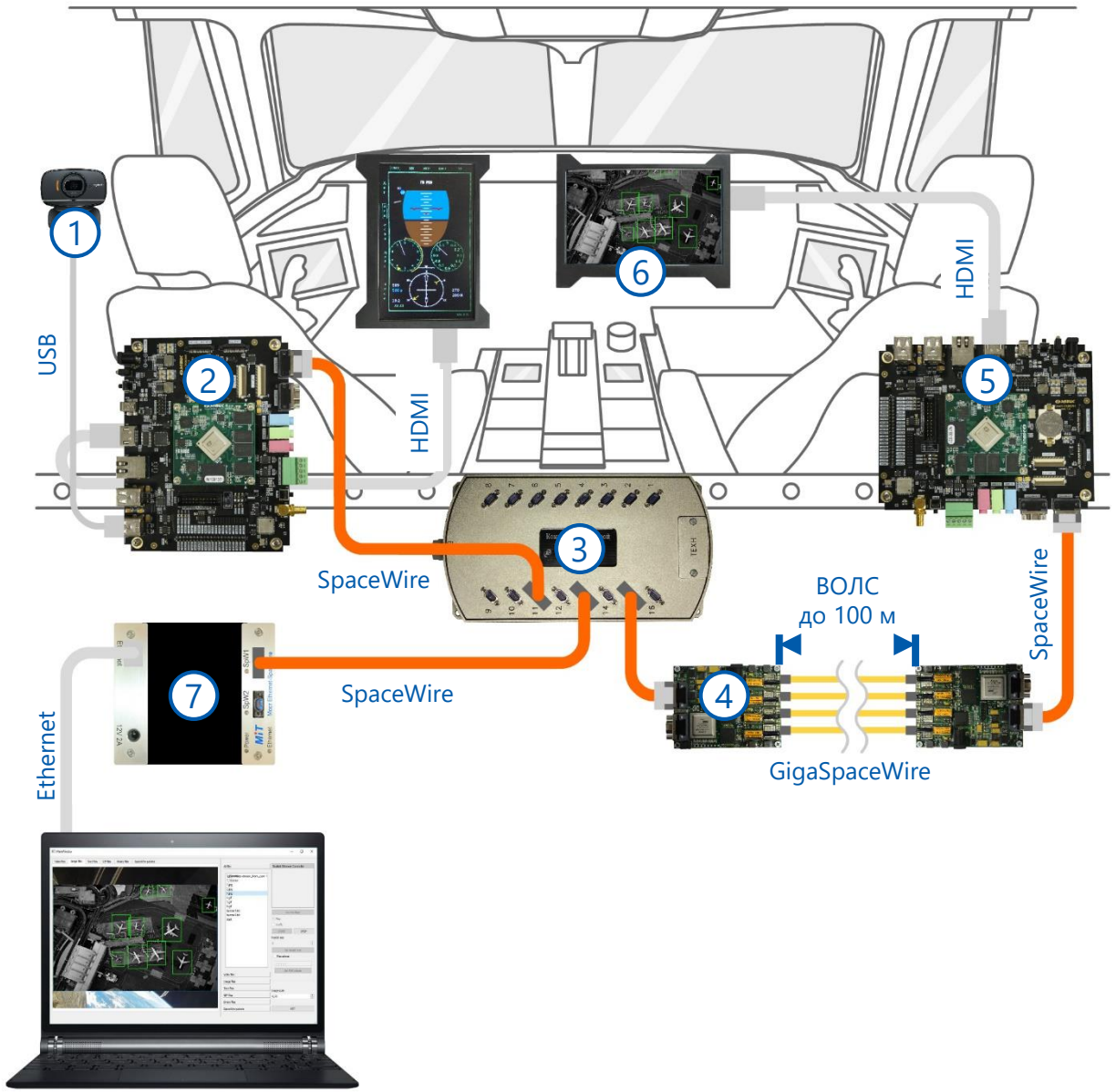
Задача системы с цифровыми датчиками



Задача ИМА-системы с обработкой и отображением



VIPE на МАКС-2019

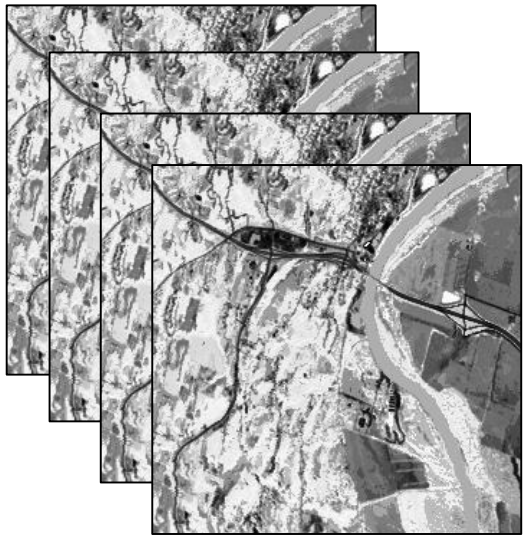


Стенд демонстрирует возможность высокопроизводительной распределенной обработки информации в сегменте сети SpaceWire на примере передачи и анализа видеопотока.

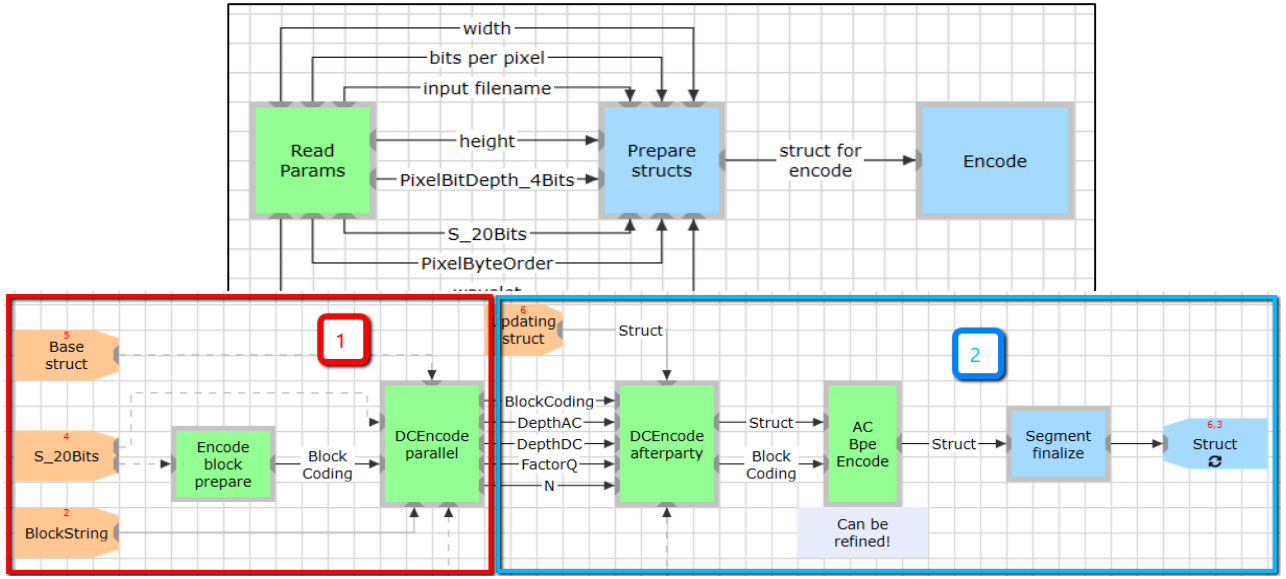
- Процессорный модуль Салют ② ⑤ для обработки и видеоанализа видео потока
- Коммутатор SpW ③ дублирует входящий поток двум получателям
- Мост SpW-ВОЛС ④ позволяет по ВОЛС передавать данные на расстояния до 100 м
- Мост Ethernet-SpaceWire ⑦ передает поток по Ethernet на ноутбук для отладки и анализа

Стенд на МАКС-2019

Use case: сжатие изображений по CCSDS 122.0



Source: http://hyperspectral.unl.edu



Продемонстрирована портируемость VPL программы на 3 платформы:

- x86
- ARM
- Leon 4

Программа сжатия изображений по CCSDS 122.0 разрабатывалась в проекте PaPP Европейской технической платформы ARTEMIS/ECSEL.

Joint Technology Initiatives - Collaborative Project (ARTEMIS)

ARTEMIS-2011-1

Grant Agreement Number 295440

PaPP

Portable and Predictable Performance on Heterogeneous Embedded Manycores

Программный комплекс разработки ПО

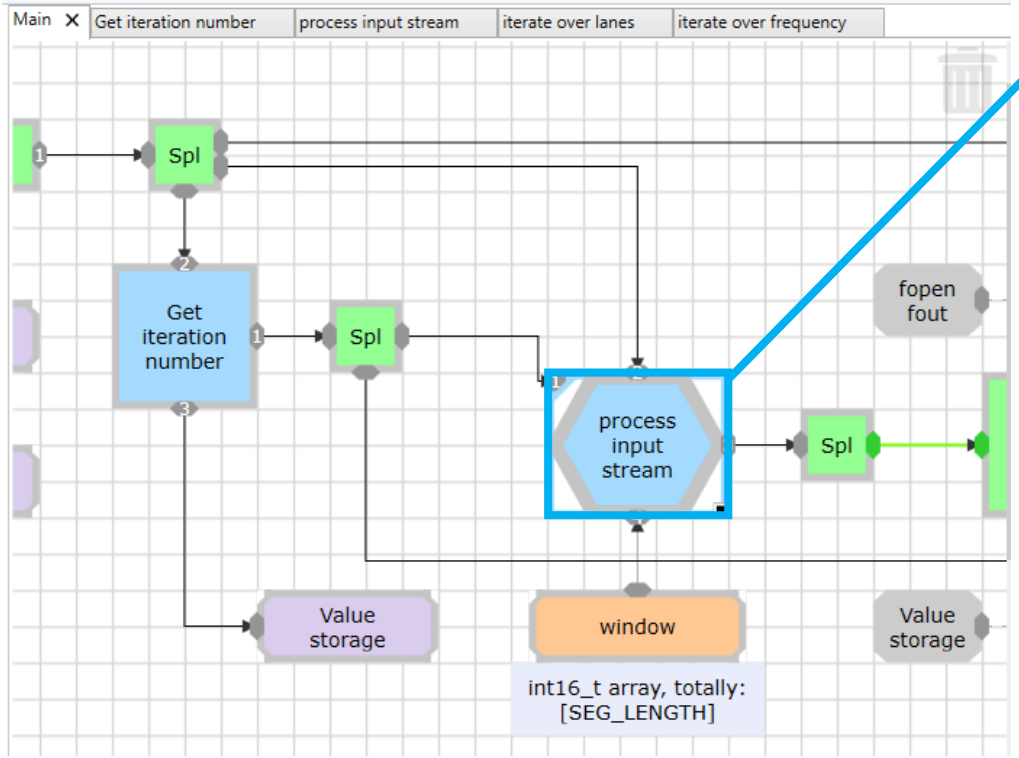
Индустриальный партнер, владелец use case и разработчик аппаратной платформы

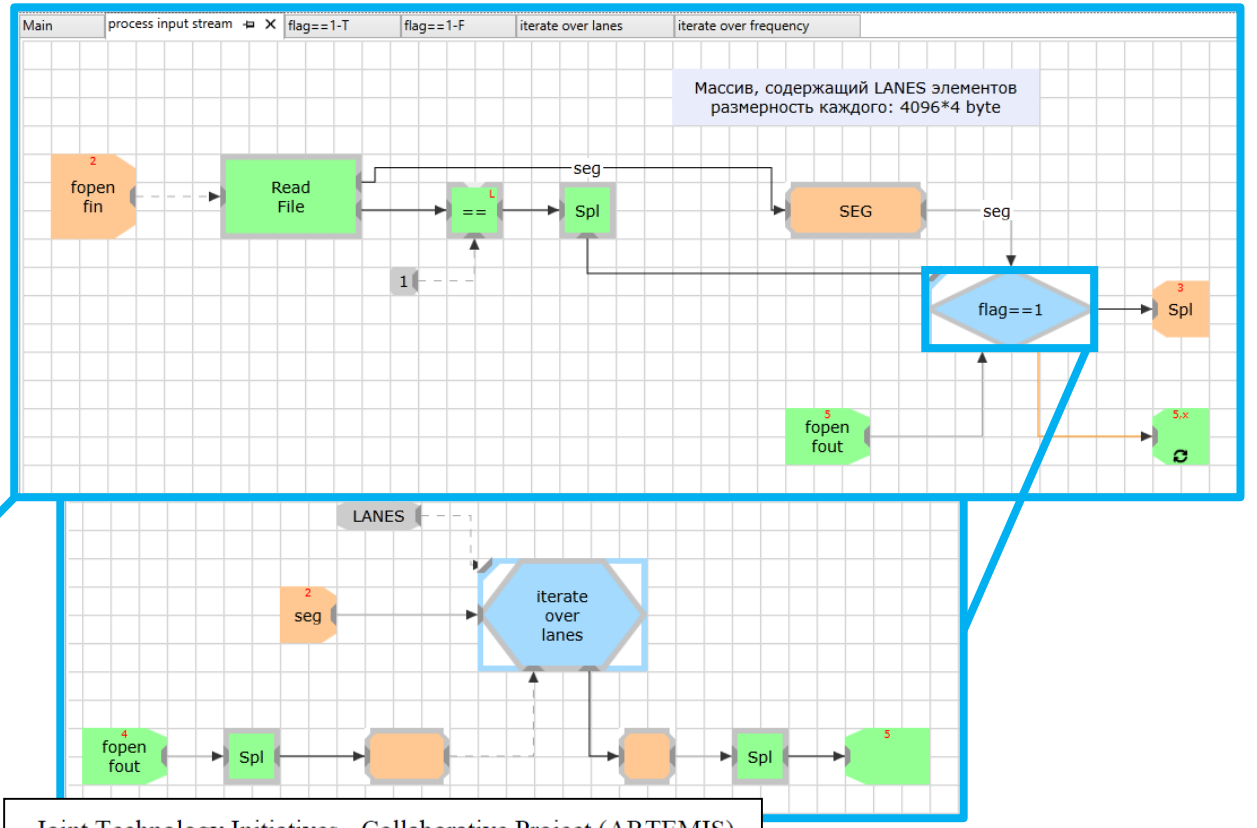
SUAI

ThalesAlenia Space
a Thales / Leonardo company

Use case: отслеживание движения объектов радаром

Программа отслеживания движения объектов радаром разрабатывался в проекте PaPP Европейской технической платформы ARTEMIS/ECSEL.





Joint Technology Initiatives - Collaborative Project (ARTEMIS)

ARTEMIS-2011-1

Grant Agreement Number 295440

PaPP

Portable and Predictable Performance on Heterogeneous Embedded Manycores

SUAI

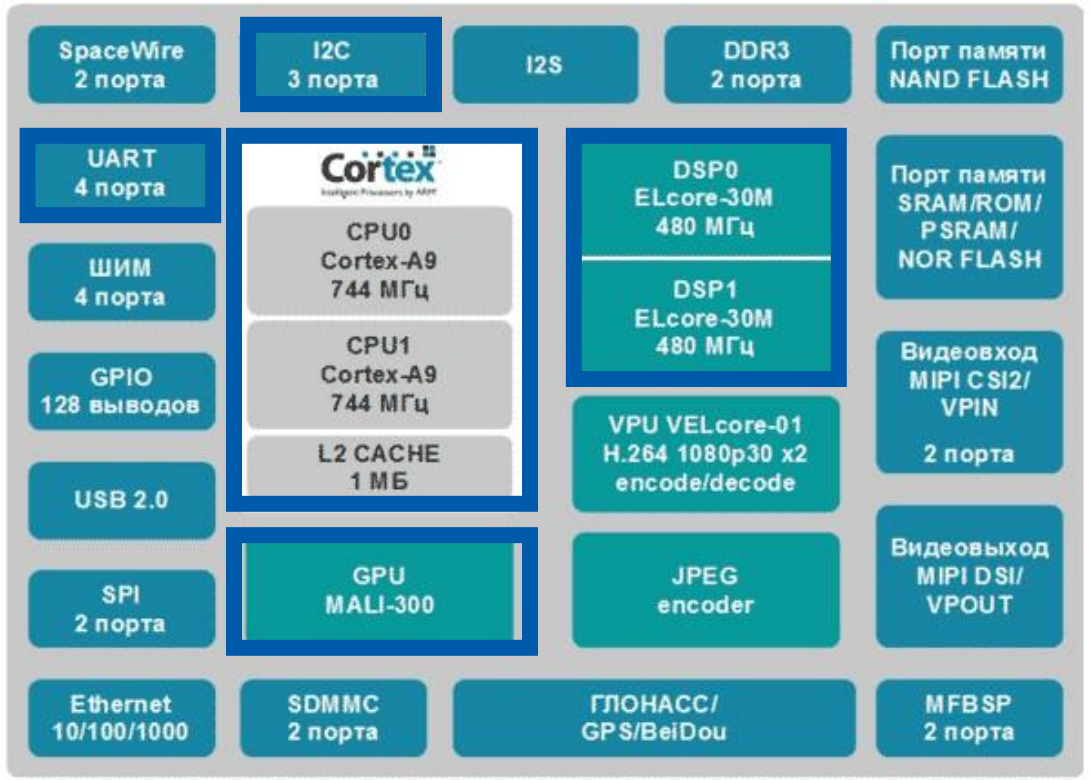
tecnalia

VIPE для ВМ14Я/Салют

Реализована полная интеграция VIPE с ВМ14Я/Салют*

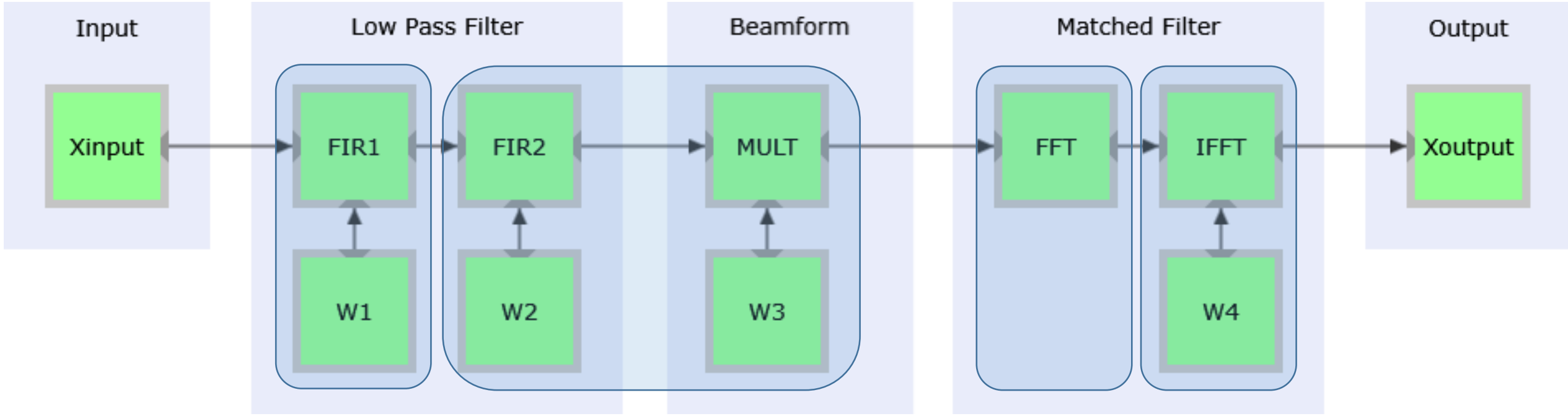
- Удаленная кросс-сборка «одной кнопкой»
- Поддержка:
  - Кластер CPU
  - DSP (OpenVX)
  - GPU (OpenGL)
  - Периферии: UART, I2C
  - Сетевые контроллер: Ethernet, SpaceWire

VIPE для ВМ14Я/Салют готов к использованию.

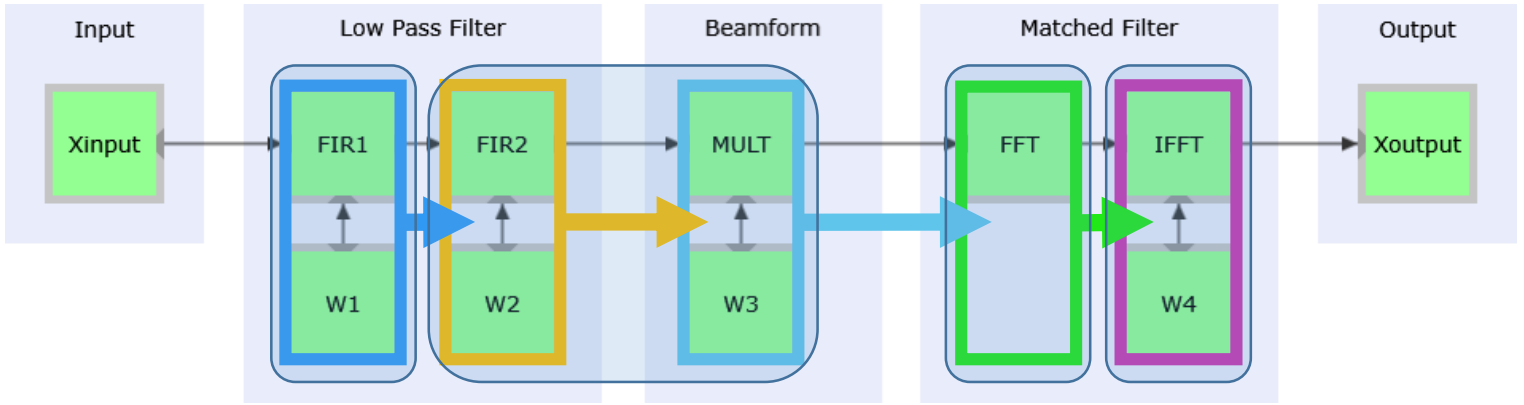


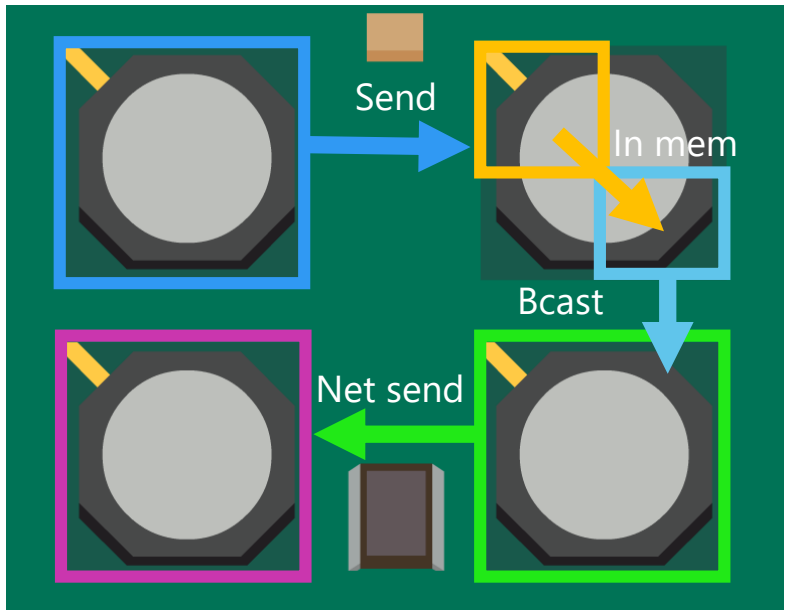
** Отечественный гетерогенный малопотребляющий многоядерный сигнальный микропроцессор 1892ВМ14Я для связных, навигационных, мультимедийных, встраиваемых, мобильных приложений.*
АО НПЦ «ЭЛВИС».

Поддержка partitioning в VIPE



Поддержка многопроцессорных систем в VIPE



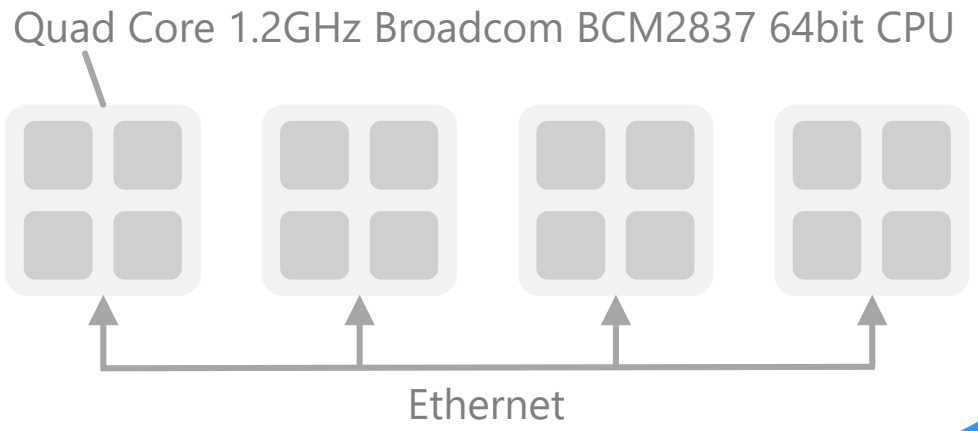


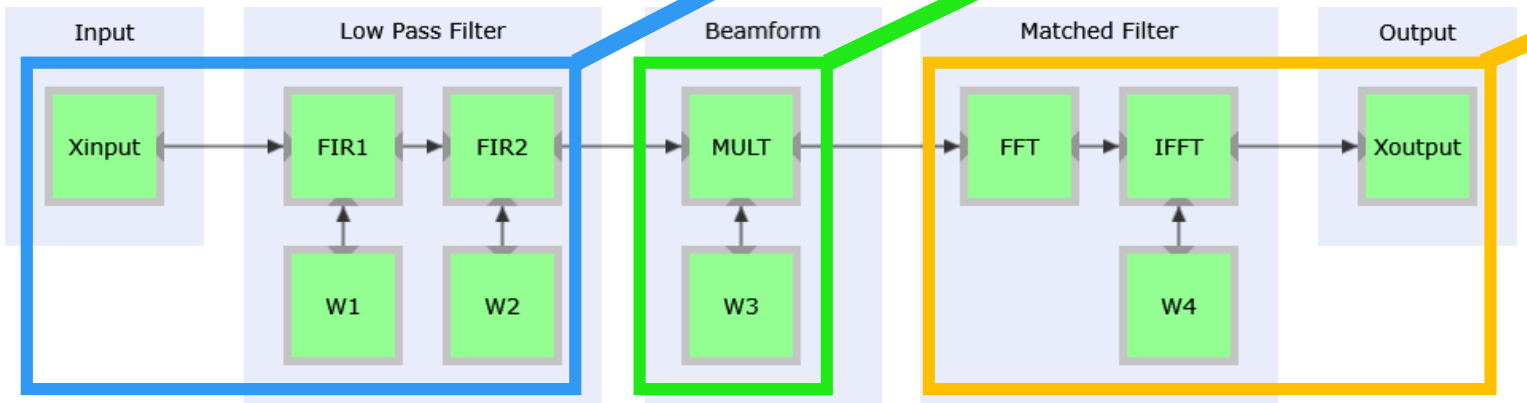
При анализе и генерации кода учитывается:

- Производительность процессоров
- Пропускная способность сетевых каналов
- Задержки, требования и т.д.

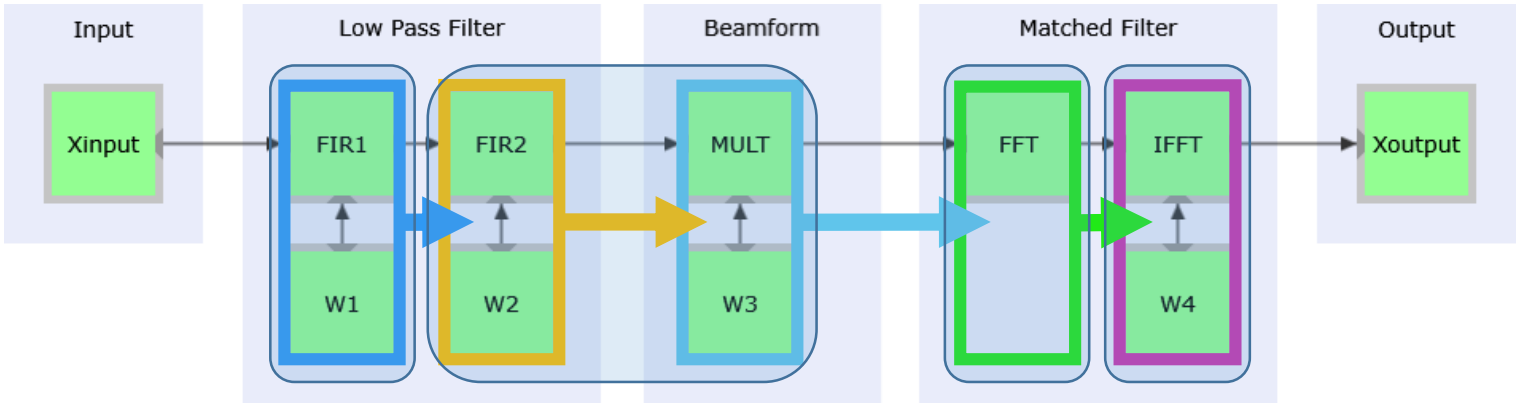
Поддержка распределенных систем в VIPE

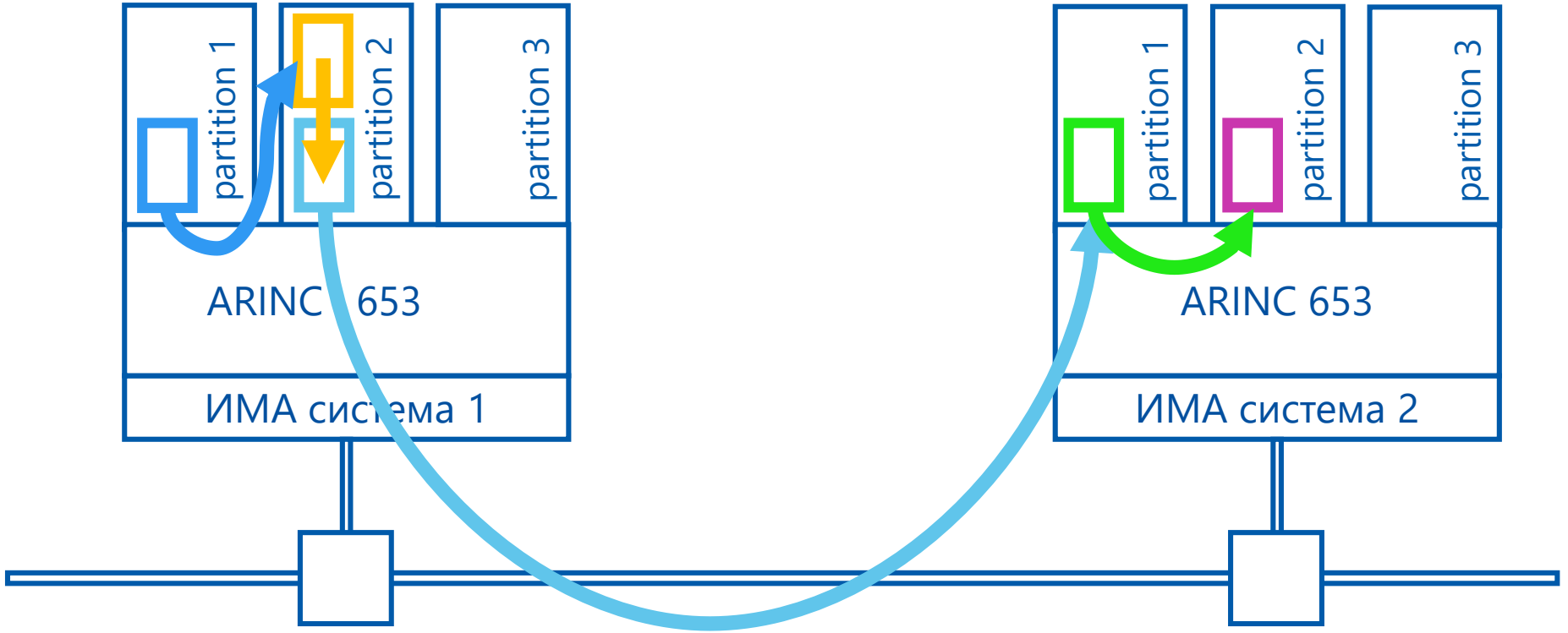
Распределенная система на основе четырех многоядерных вычислительных модулей





Поддержка IMA / DIMA систем в VIPE





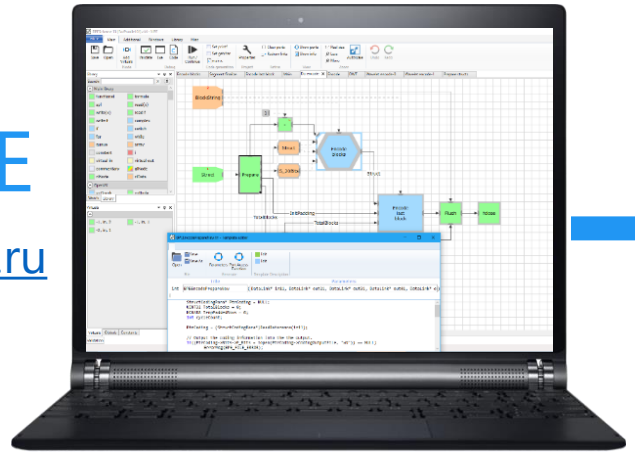
Выводы

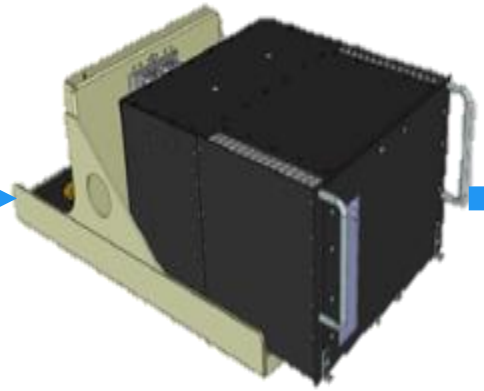
Проектирование и программирование в VIPE

Развертывание на целевой платформе

Интеграционное тестирование

VIPE
www.vipetech.ru





VIPE обеспечивает:

- поддержку создания и сопровождения ФПО на этапах жизненного цикла на основе общей программной модели
- совместное проектирование ФПО различными специалистами, сопровождение ФПО
- унификацию разработки ФПО для различных предметных областей
- эффективное программирование распределенных, многопроцессорных, многоядерных, неоднородных встроенных систем
- переносимость и адаптация ПО под различные платформы КБО

VIPE – инструмент для разработки ФПО для ИМА