

Лаборатория
Вычислительных
Комплексов

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ

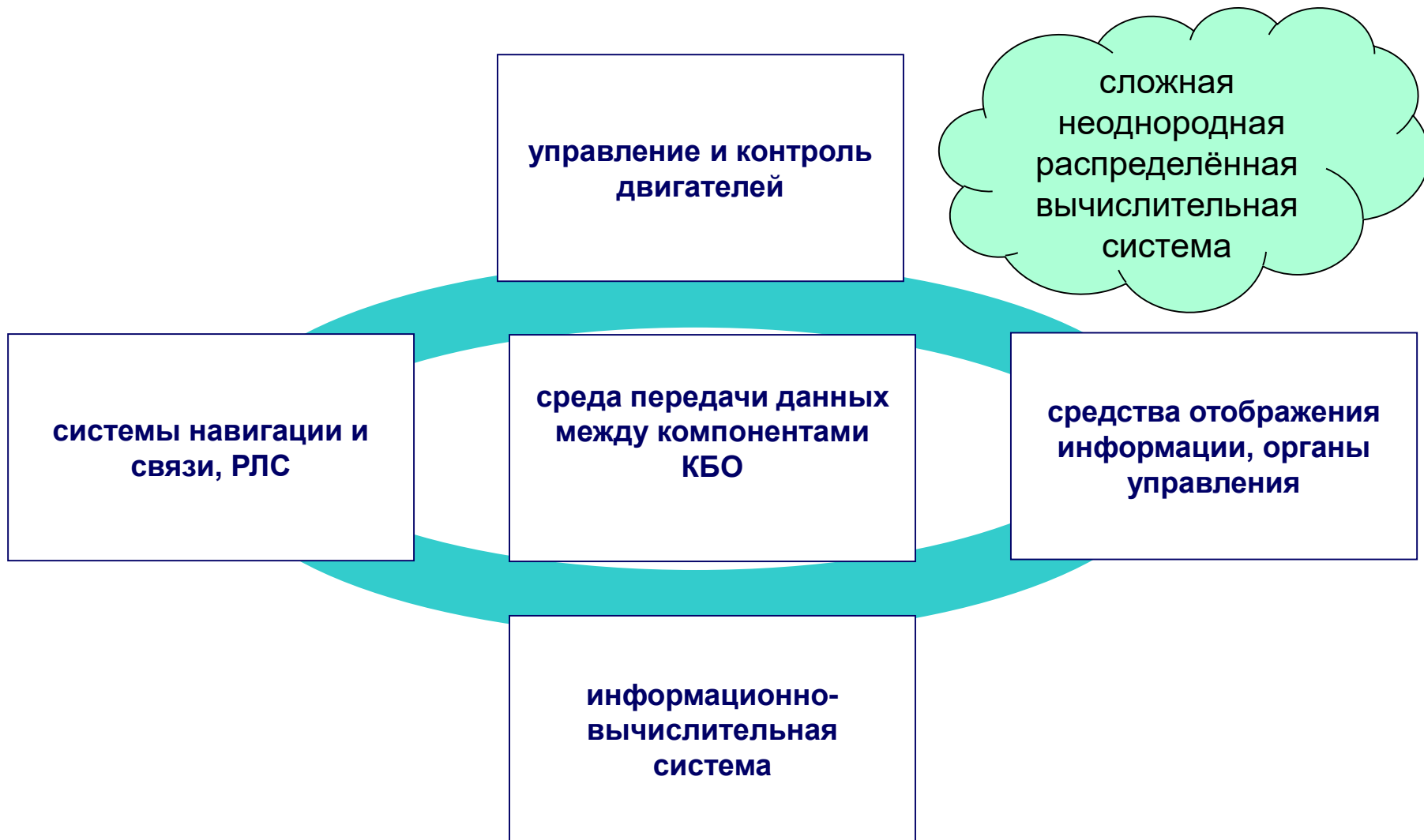
Лекция 13
Жизненный цикл ПО ИУС РВ и
инструментальные средства его поддержки

Кафедра АСВК,
Лаборатория Вычислительных Комплексов
Балашов В.В.

РАЗРАБОТКА ПО ИУС РВ – ТЕХНОЛОГИЧЕСКИ И ОРГАНИЗАЦИОННО СЛОЖНЫЙ ПРОЦЕСС

На примере разработки бортовых ИУС РВ

Комплекс бортового оборудования



Специфика бортовых ИУС РВ

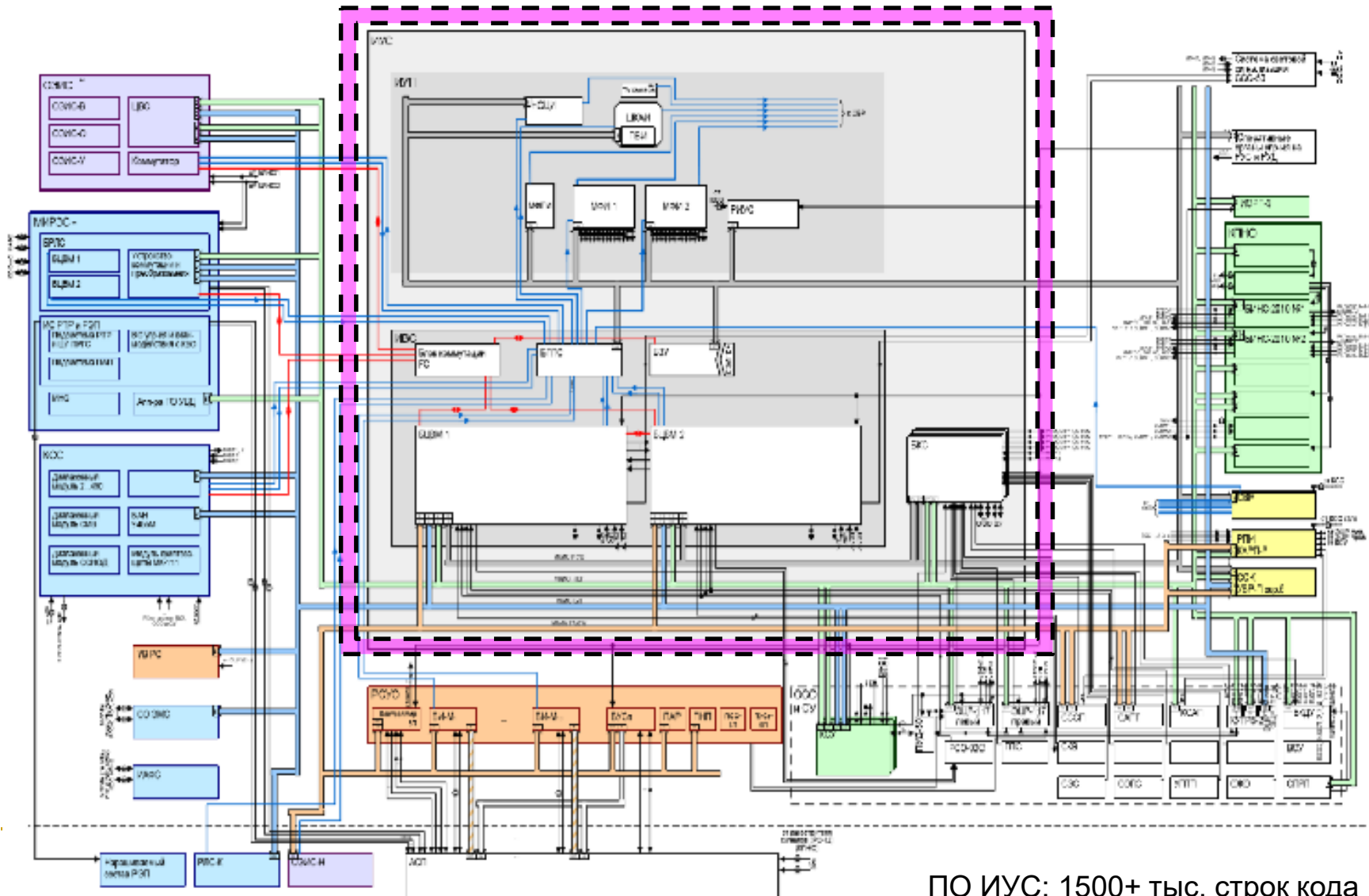
- Высокая сложность
- Функционирование в реальном времени
 - Вычисления
 - Информационный обмен
- Требования
 - Функциональность ← разнообразие
 - Надёжность
 - Реальное время
- Критичность
- Неоднородность
 - Каналы: точка-точка, шина, коммутатор; 12 kbps, 1 Mbps, 1 Gbps
 - Устройства: датчики, индикаторы, вычислители, органы управления, исполнительные устройства
 - Данные: аналоговые, цифровые; числовые массивы, видеопотоки

Бортовые ИУС РВ и их ПО требуют систематического подхода к проектированию, реализации и тестированию

Бортовые устройства



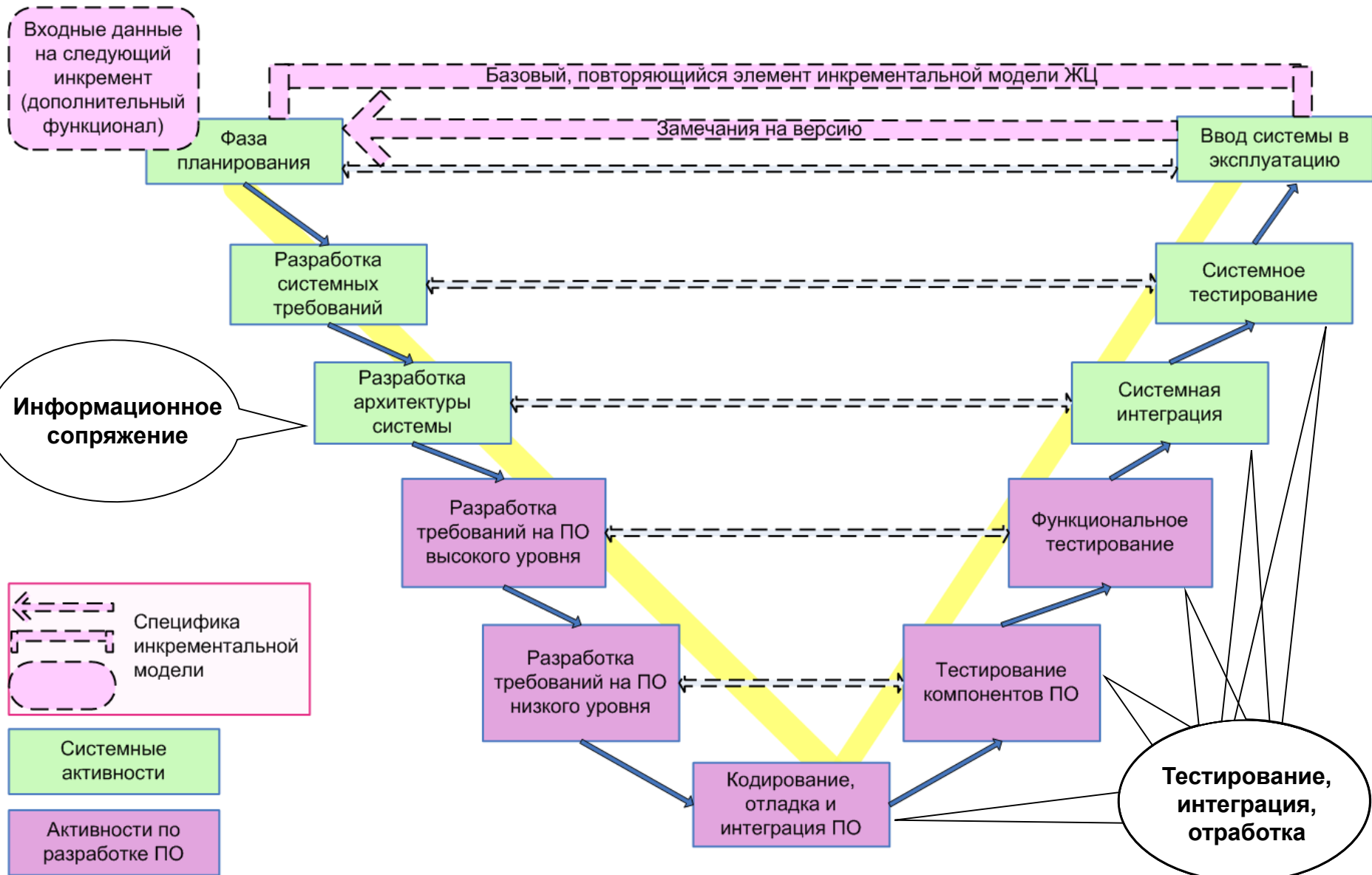
Уровень сложности комплекса бортового оборудования



ПО ИУС: 1500+ тыс. строк кода

Жизненный цикл бортового ПО

Жизненный цикл бортового ПО



– тестирование на каждом уровне

Процессы ЖЦ по стандарту DO-178B

Software Considerations in Airborne Systems and Equipment

- Процесс планирования и управления проектом
- Процесс разработки ПО
 - Разработка требований
 - Проектирование
 - Кодирование, отладка и интеграция
- Интегральные процессы
 - Управление конфигурацией
 - Верификация
 - Управление качеством
 - Сертификационное взаимодействие
(в данном курсе не рассматривается)

Влияние на состав фаз ЖЦ

Процессы

Процесс	Подпроцессы	Основные цели	Основные артефакты
Планирование и управление проектом		Определение основной методологии разработки, написание планов разработки, определение инструментов разработки, определение и проведение необходимых тренингов, определение методов мониторинга проекта. Выпуск необходимых шаблонов.	Планы (разработки, тестирования ПО, конфигурационного управления, управления качеством и т.д.); отчеты о рецензировании планов; графики; стандарты; шаблоны артефактов (документов) проекта; протоколы совещаний; официальные письма; рабочая переписка (e-mail); метрики и индикаторы, отражающие состояние процессов и прогресс, презентации; списки мероприятий.

Процессы

Процесс	Подпроцессы	Основные цели	Основные артефакты
Разработка программного обеспечения	Процесс разработки требований к ПО и процесс проектирования ПО	Создание и контроль требований верхнего и нижнего уровня на разработку программного обеспечения. Контроль полноты и непротиворечивости требований, прослеживаемости и тестируемости.	Спецификации требований в текстовом формате. Если требования разрабатываются с использованием инструментальных средств, то модули в формате инструментального средства. Формальные спецификации (данные, подготовленные для автоматической генерации кода) в формате их редакторов; формальные спецификации в формате HTML, или текстовом формате; отчеты о рецензировании формальных спецификаций; описание графической информации в необходимом объеме; Результаты автоматического контроля формальных спецификаций; матрица соответствия элементов формальных спецификаций и требований к ним с отчетом о рецензировании; сгенерированный код.

Процессы

Процесс	Подпроцессы	Основные цели	Основные артефакты
Разработка программного обеспечения	Процесс кодирования ПО и процесс интеграции	<p>Разработка программного кода в соответствии со спецификациями, получение загрузочных файлов.</p> <p>В процессе кодирования исходный код составляется на основе архитектуры ПО и требований нижнего уровня.</p> <p>Формирование исполняемого кода, загрузка исполняемого объектного кода на целевой вычислитель для интеграции ПО и аппаратных средств.</p>	<p>Код, разработанный вручную; служебные файлы, необходимые для компиляции и сборки кода; матрица покрытия требований кодом; код, сгенерированный автоматическими средствами генерации.</p> <p>Объектные файлы.</p> <p>Загрузочный файл</p>

Процессы

Процесс	Основные цели	Основные артефакты
Верификация программного обеспечения	<p>Интегральный процесс, целью которого является обнаружение и регистрация ошибок, которые могли появиться в процессе разработки ПО. Эти цели достигаются через ряд мероприятий: написание тестовых процедур в соответствии с требованиями, проведение анализа кода и тестовых процедур, проведение формального тестирования, анализ полученных результатов, написание сообщений о проблемах, найденных в процессе различных верификационных активностях, создание и выпуск отчетов о проведенном тестировании.</p>	<p>Отчет об автоматическом анализе кода, отчеты о рецензировании кода.</p> <p>Тестовые процедуры и скрипты в формате редакторов тестов и текстовом формате, в случае использования инструментальных средств разработки требований и тестовых процедур – модули в формате инструментального средства; отчеты о рецензировании тестов; отчеты о рецензировании результатов тестирования; матрица покрытия требований тестами с отчетом о её рецензировании; матрица результатов тестирования с отчетом о её рецензировании; реестр со списком всех верификационных документов для версии ПО.</p>

Процессы

Процесс	Основные цели	Основные артефакты
Управление качеством	Интегральный процесс, целью которого является контроль качества продукта и контроль соблюдения процессов.	Отчеты (протоколы) проверки качества продукции; отчеты (протоколы) проверки процессов; планы проверки.

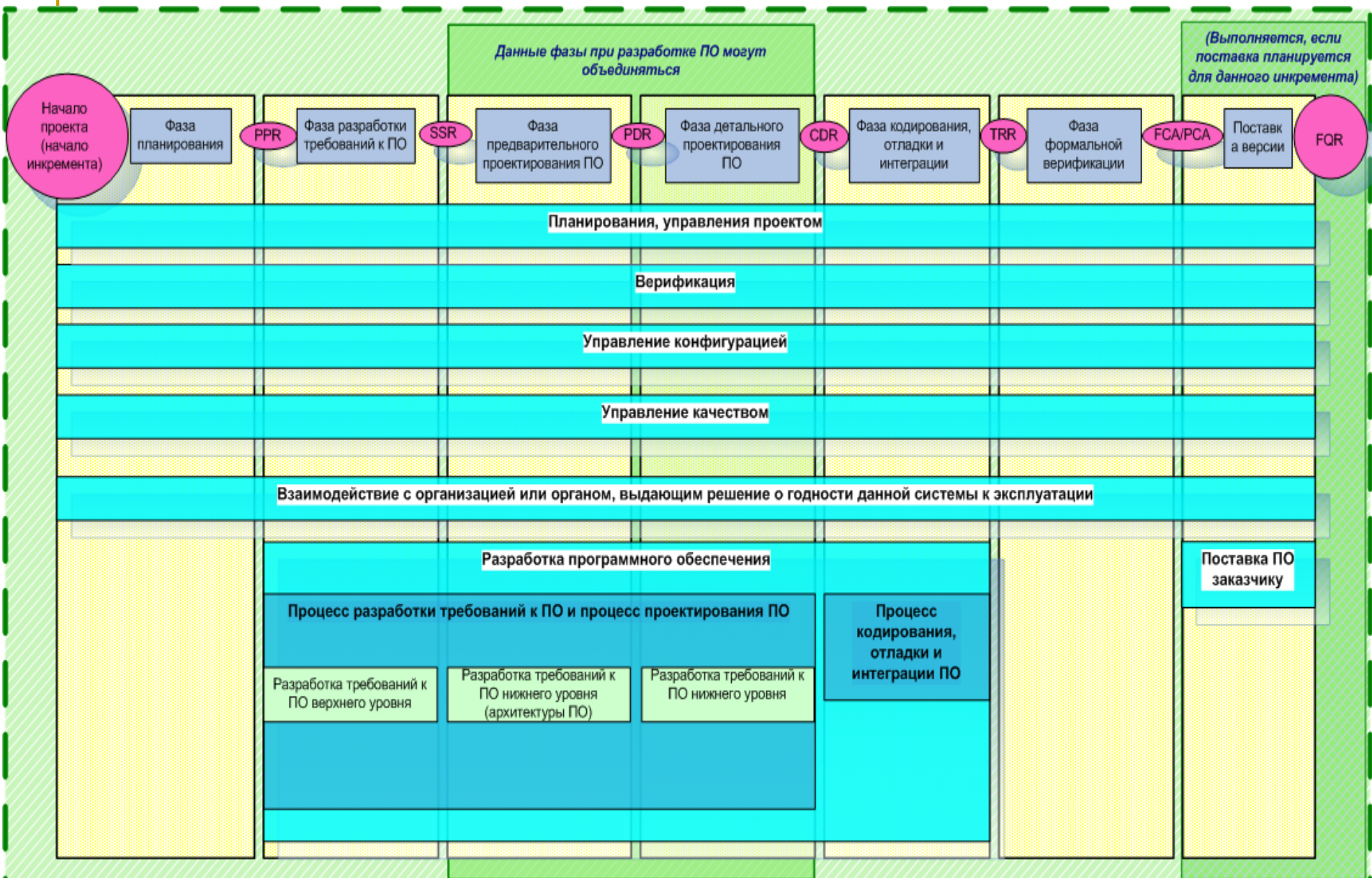
Процессы

Процесс	Основные цели	Основные артефакты
Конфигурационное управление, включая управление изменениями	<p>Интегральный процесс, целью которого является:</p> <ul style="list-style-type: none">определение и идентификация продукта,контроль выпуска и изменения версий в течение жизненного цикла,контроль корректности и полноты версии,версионный контроль всех артефактов проекта. <p>Определение конфигурационной структуры, обеспечение однозначной конфигурационной идентификации артефактов проекта.</p> <p>Установление и проведение процесса управления изменениями.</p>	<p>Поставки (файлы-архивы) и сопровождающие файлы-отчеты по каждому из процессов; описание версии ПО; сопровождающая документация; носитель версии (оптический диск), если это оговорено – с необходимыми атрибутами (наклейками или другой идентифицирующей информацией); конфигурационный перечень артефактов проекта.</p> <p>Базы данных проблем - описания проблем (в формате, пригодном для импорта в средство управления изменениями; в текстовом, или ином удобном для чтения формате), протоколы совещаний по управлению конфигурацией. В случае наличия независимых баз данных у различных участников кооперации, предусмотреть наличие процесса и артефактов для поддержки синхронизации информации в этих базах данных.</p>

Процессы

Процесс	Основные цели	Основные артефакты
Поставка ПО заказчику	Передача разработанного продукта заказчику.	Описание версии ПО; финальная версия поставляемого ПО; заключение о годности версии ПО к натурным испытаниям на целевой системе.

Соотношение фаз и процессов ЖЦ ПО



Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Фаза планирования	Определение методов и инструментов для производства программного продукта в процессе написания планов разработки ПО, написание стандартов разработки и создание необходимых шаблонов.	Инструменты, позволяющие: <ul style="list-style-type: none">- осуществлять планирование (в т.ч. создавать графики разработки),- создавать планы разработки и тестирования,- создавать шаблоны разработки,- отслеживать ход выполнения проекта,- осуществлять коммуникацию, взаимодействие различных участников проекта.

Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Фаза разработки требований к ПО	<p>Анализ входных требований.</p> <p>Написание требований верхнего и нижнего уровня на разработку ПО.</p> <p>Определение бортовых интерфейсов, структуры и протоколов среды обмена.</p> <p>Построение матрицы прослеживаемости между требованиями различного уровня.</p> <p>Написание формальных спецификаций.</p> <p>Начало написания тестовых процедур, соответствующих требованиям верхнего уровня.</p>	<p>Инструменты, позволяющие:</p> <ul style="list-style-type: none">- создавать требования,- создавать связи между требованиями,- разрабатывать и поддерживать интерфейсы между различными компонентами ПО,- проводить необходимые обзоры артефактов,- создавать тестовые процедуры,- разрабатывать формальные спецификации.

Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Фаза проектирования (дизайна) ПО	<p>Определение архитектуры ПО, декомпозиция ПО, декомпозиция распределения памяти.</p> <p>Определение структуры ПО с учетом требований реального времени.</p> <p>Написание (детализация) тестовых процедур в соответствии с уточнениями и детализацией требований.</p> <p>Построение предварительной матрицы покрытия требований тестовыми процедурами.</p>	<p>Инструменты, позволяющие:</p> <ul style="list-style-type: none">- создавать и поддерживать описание архитектуры ПО,- прослеживать связи между требованиями различного уровня,- задавать и прослеживать соответствие тестовых процедур и требований,- осуществлять необходимые обзоры артефактов,- создавать матрицу покрытия требований тестовыми процедурами.

Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Фаза кодирования, отладки и интеграции	<p>Генерация кода из формальных спецификаций.</p> <p>Разработка ручного кода.</p> <p>Интеграция всех разработанных программных компонент в единый модуль.</p> <p>Загрузка программного модуля на целевой вычислитель.</p> <p>Отладка разработанного кода и, параллельно, тестовых процедур.</p> <p>В конце фазы происходит «заморозка» программного кода и тестовых процедур.</p>	<p>Инструменты, позволяющие:</p> <ul style="list-style-type: none">- генерировать код на языке высокого уровня из формальных спецификаций;- разрабатывать ручной код;- проводить компиляцию исходного кода;- проводить компоновку двоичного кода;- проводить загрузку на целевой вычислитель и интеграцию;- проводить отладку на целевых вычислителях;- выполнять интеграционные тесты с имитацией данных на входах целевых вычислителей;- проводить коррекцию программного кода и тестовых процедур в случае обнаружения ошибок;- разрабатывать и выпускать необходимую документацию.

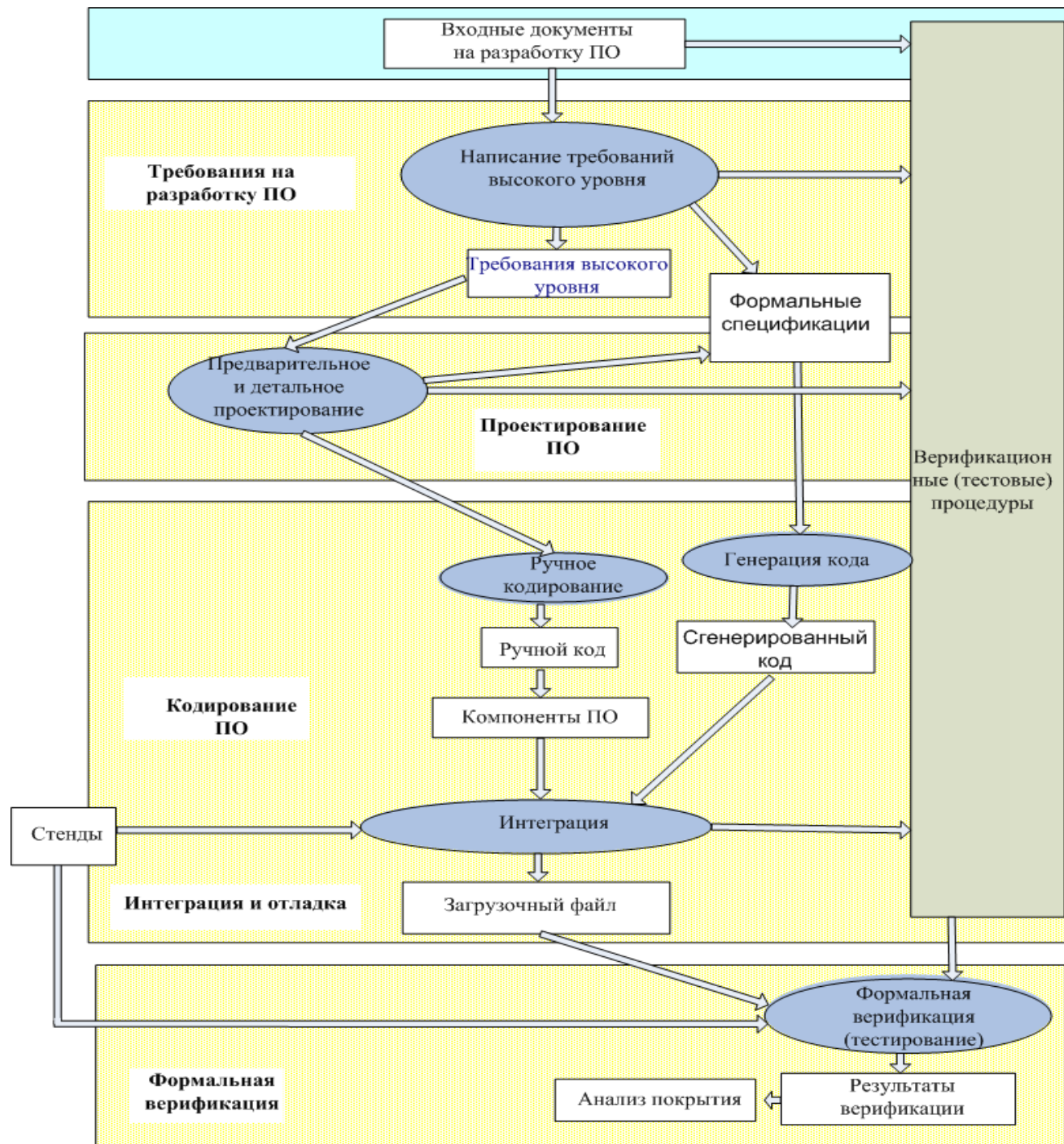
Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Формальная верификация	<p>Проведение прогона разработанных тестовых процедур на «замороженной» версии ПО (тестирование).</p> <p>Анализ результатов тестирования.</p> <p>Создание сообщений о проблемах по результатам тестирования.</p> <p>Финализация матрицы покрытия.</p>	<p>Инструменты и средства, позволяющие:</p> <ul style="list-style-type: none">- осуществлять прогон тестируемого ПО на целевом вычислителе,- выполнять тесты с имитацией данных на входах целевых вычислителей,- собирать результаты тестирования, в т.ч. формировать протоколы тестирования,- проводить необходимые обзоры артефактов,- оформлять сообщения о проблемах,- модифицировать матрицу покрытия.

Фазы

Название фазы	Основные активности внутри фазы	Применяемые инструментальные средства
Подготовка финальной поставки	Выпуск итоговых документов по составу версии.	Инструменты, позволяющие создавать, модифицировать и выпускать итоговые документы проекта

Активности по фазам ЖЦ ПО



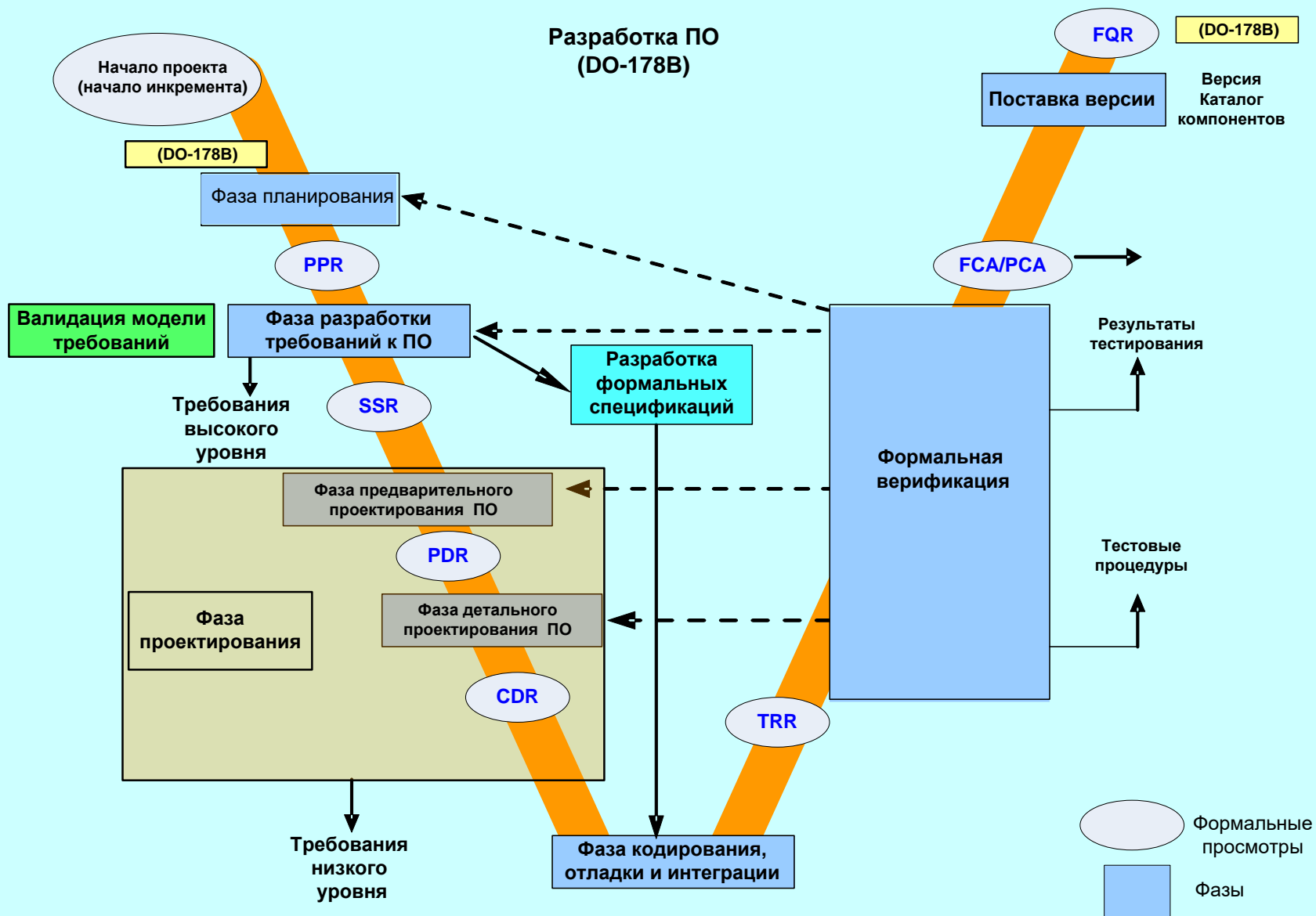
Вехи

Вехи		Аббревиатура	Описание
Русское название	Английское название		
Обзор планов	Plan Process Review	PPR	Одобрение, официальный выпуск планов разработки
Обзор спецификаций	Software Specification Review	SSR	Одобрение, официальный выпуск требований на разработку ПО и начало фазы предварительного дизайна
Предварительный обзор архитектуры	Preliminary Design Review	PDR	Одобрение, официальный выпуск предварительного дизайна (архитектуры ПО) и начало активностей по определению детального дизайна
Критический обзор архитектуры	Critical Design Review	CDR	Одобрение, официальный выпуск детального дизайна ПО
Обзор готовности тестов	Test Readiness Review	TRR	Начало выполнения формальной верификации (формального тестирования)

Вехи

Вехи		Аббревиатура	Описание
Русское название	Английское название		
Функциональный конфигурационный аудит (Функциональный аудит программных конфигураций)	Functional Configuration Audit	FCA	Проверка того, что элементы конфигурации отвечают всем функциональным требованиям, в том числе требованиям по производительности. FCA представляет обзор функционирования элементов, не только для обеспечения соответствия спецификации, но и для выявления не заданных требованиями функциональных характеристик.
Физический конфигурационный аудит (Физический аудит программных конфигураций)	Physical Configuration Audit	PCA	PCA обеспечивает гарантию того, что поставляемые результаты физически соответствуют перечисленным в документации пунктам и представлены в поставляемой версии.
Формальный квалификационный обзор	Formal Qualification Review	FQR	Успешное завершение этих аудитов может быть обязательным требованием для финального фиксирования версии.

Фазы и вехи на V-цикле



Инструментальные средства поддержки ЖЦ бортового ПО

Средства поддержки разработки требований

- Функциональность бортового ПО описывается десятками тысяч требований (системных; собственно к ПО)
- Необходимая функциональность:
 - создание и хранение требований, отслеживание истории
 - связывание требований с версиями документов и ПО
 - прослеживаемость требований на:
 - Низкоуровневые требования
 - Формальные спецификации
 - Код
 - Тесты
- Примеры средств: IBM DOORS, Borland CaliberRM, SyBase PowerDesigner

Примеры требований к подсистеме ИУС РВ

BCS.INIT.SELF

Встроенная самопроверка БЦВМ

После включения питания БЦВМ должна выполнять начальную встроенную самопроверку. Взаимодействие по каналам КБИ при этом не выполняется. Продолжительность встроенной самопроверки не менее 2 сек. и не более 8 сек. По результатам встроенной проверки загорается светодиод исправности БЦВМ.

BCS.INIT.MILS

Контроль МКИО

После встроенной самопроверки БЦВМ должна выполнять контроль обменов по МКИО:

- передачу групповой команды "Установить ОУ в исходное состояние";
- передачу групповой команды "Начать самоконтроль ОУ";
- опрос векторного слова самоконтроля у абонентов МКИО;

Продолжительность встроенной самопроверки не более 3 сек.

BCS.SKV.TEST.START

Включение режима 'Тест'

Не позднее, чем через 10 сек. после включения питания БЦВМ должна перевести СКВ в режим 'Тест'.

BCS.SKV.TEST.DELAY

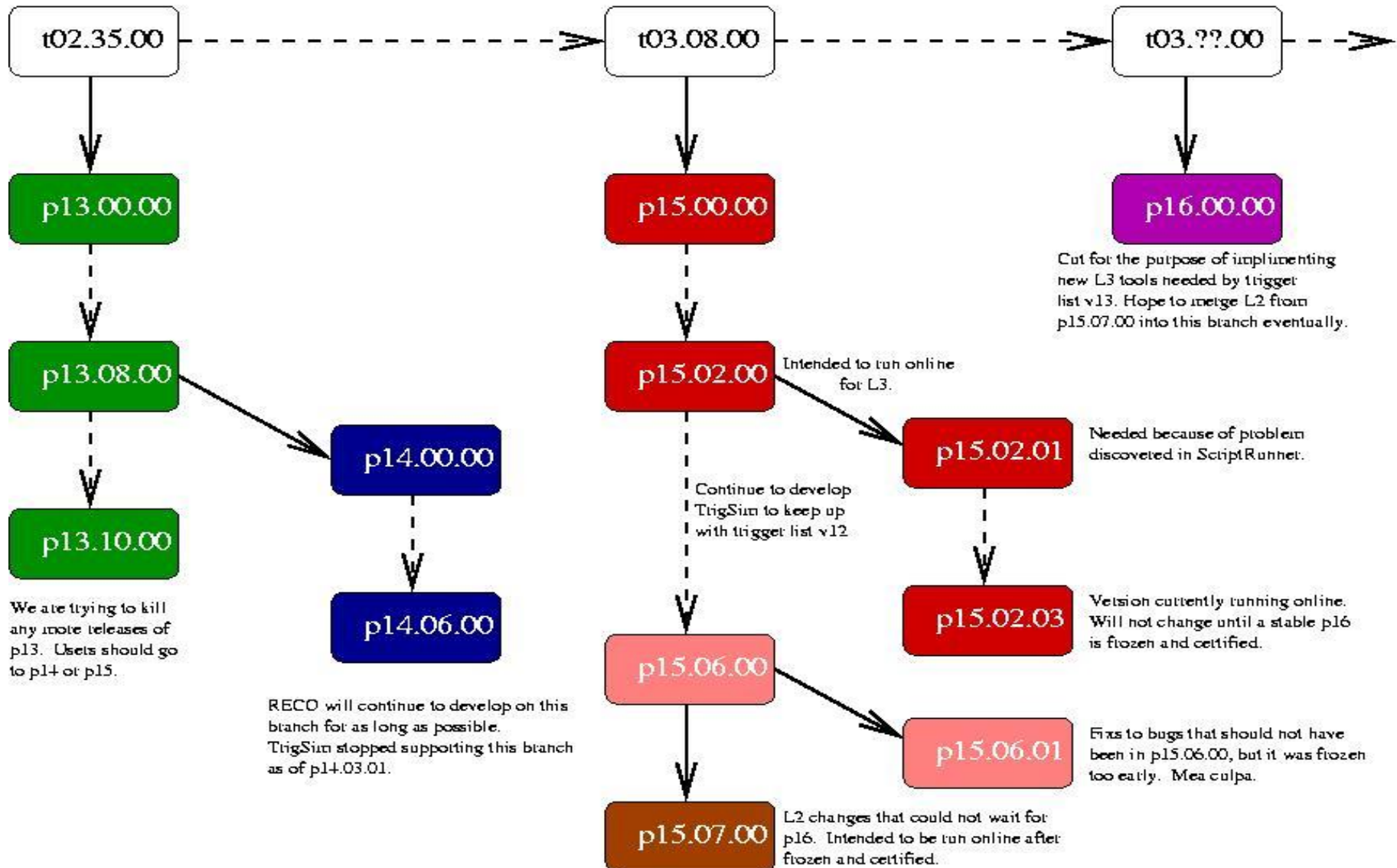
Продолжительность режима 'Тест'

Продолжительность режима 'Тест' СКВ должна быть не менее 2 сек.

Средства версионного/ конфигурационного контроля

- При разработке ПО формируется множество документов, группируемых в версии
 - Промежуточные («инженерные») версии
 - Официальные версии
- Версия = вся совокупность документов
 - Требования, спецификации, код, тесты,...
 - Часть конфигурации борта
- Необходимая функциональность:
 - Версионирование совокупности документов (в т.ч. атомарность изменений)
 - Поддержка ветвей истории
 - Поддержка групповой разработки, в т.ч. разграничения доступа
 - Обновление документов в реальном времени
- Примеры: CVS, Subversion, git, IBM ClearCase

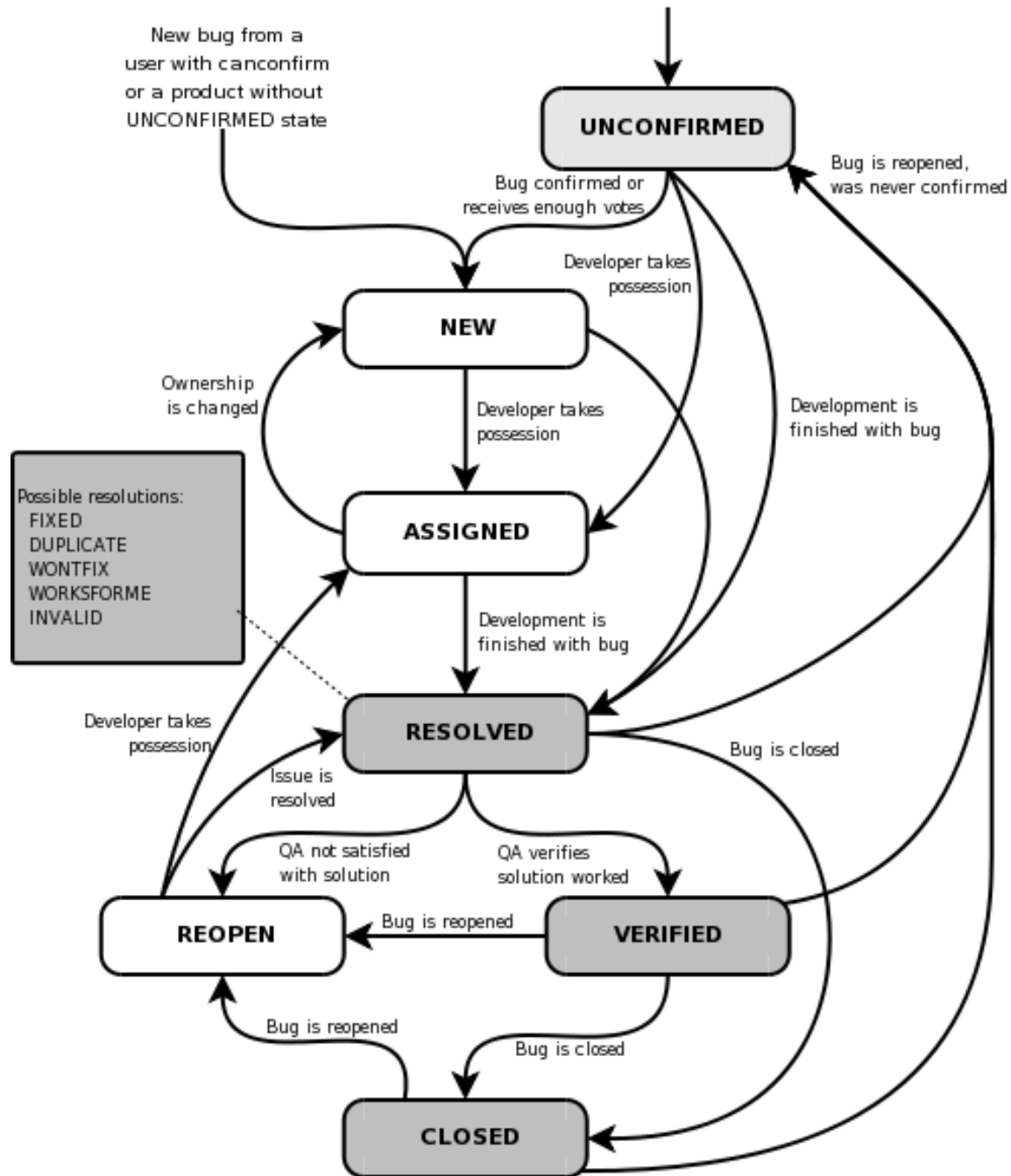
Дерево версий



Средства отслеживания проблем и изменений

- В процессе разработки ПО:
 - Дефекты и ошибки: зарегистрированы, проанализированы и устранены
 - Изменения: зарегистрированы, утверждены и реализованы
- Необходимая функциональность:
 - Поддержка структуры продукта и процесса
 - Настраиваемый формат сообщения о проблеме
 - Настраиваемый ЖЦ сообщения, поддержка согласования
 - Поддержка групповой разработки, втч разграничения доступа
 - Интеграция со средствами управления версиями
- Примеры: Bugzilla, Trac, IBM ClearQuest

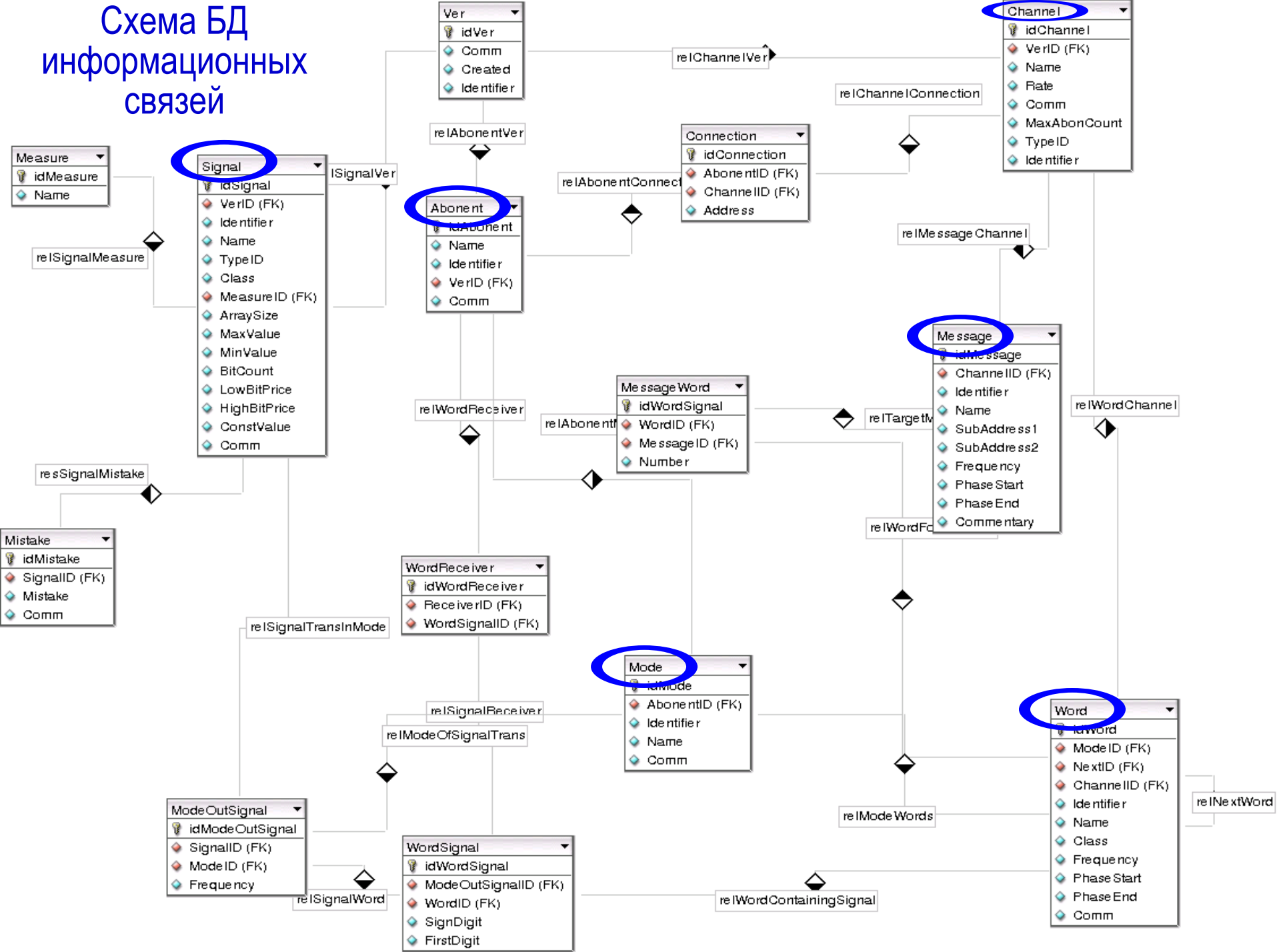
Жизненный цикл сообщения о проблеме



Средства поддержки сопряжения подсистем ПО

- Средства автоматизации проектирования бортовых интерфейсов
 - Балансировка загрузки каналов
 - Формирование набора сообщений
 - Построение расписаний обмена (канал с централизованным управлением)
 - Построение системы виртуальных каналов (сеть на основе коммутаторов)
- Средства автоматизации интеграции ПО
 - Использование унифицированных структурных компонентов ПО
 - По управлению: расписание выполнения СКПО
 - По данным:
 - БД информационных связей СКПО
 - Автоматическое формирование описания интерфейсов ПО (буфера, каналы, сообщения) для конфигурирования ОСРВ

Схема БД информационных связей



Средства автоматизации проектирования индикационных форматов

- Индикационный формат = набор графических элементов + правила поведения
 - Необходимая функциональность:
 - Редактирование в графической форме, WYSIWYG
 - Поддержка библиотеки элементов
 - Поддержка автономного тестирования
 - Генерация кода в формате для целевого устройства
 - Примеры: SCADE Display, VAPS, САПР ИФ
-

П/Т

НВГ

ОВО

БКС

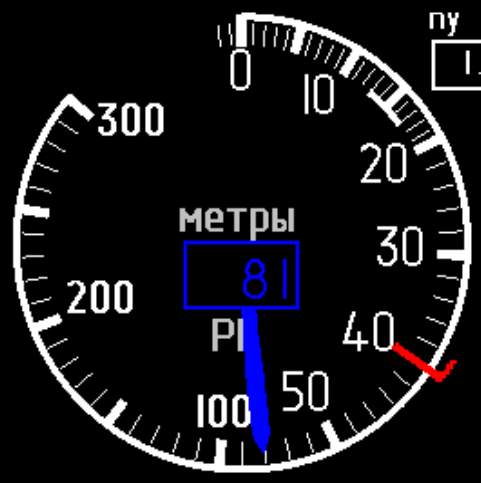
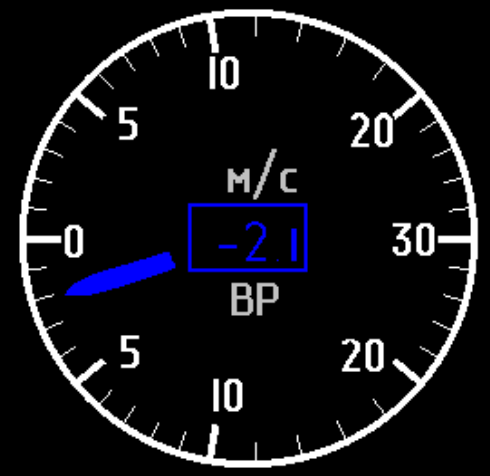
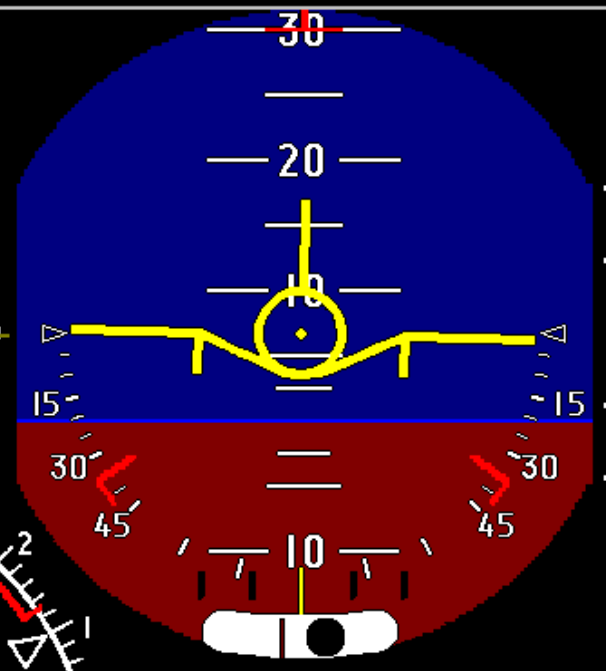
БКО

Р/С

ОПС

★

Н
7
6
0



ПЗ 9 ППМІ Д 91 км
Аз 313° ↑

И
С

И
Н

П
У

И
С
С
У

В
Л
О
К
И

А
П
Р

Т
А
Н
Г

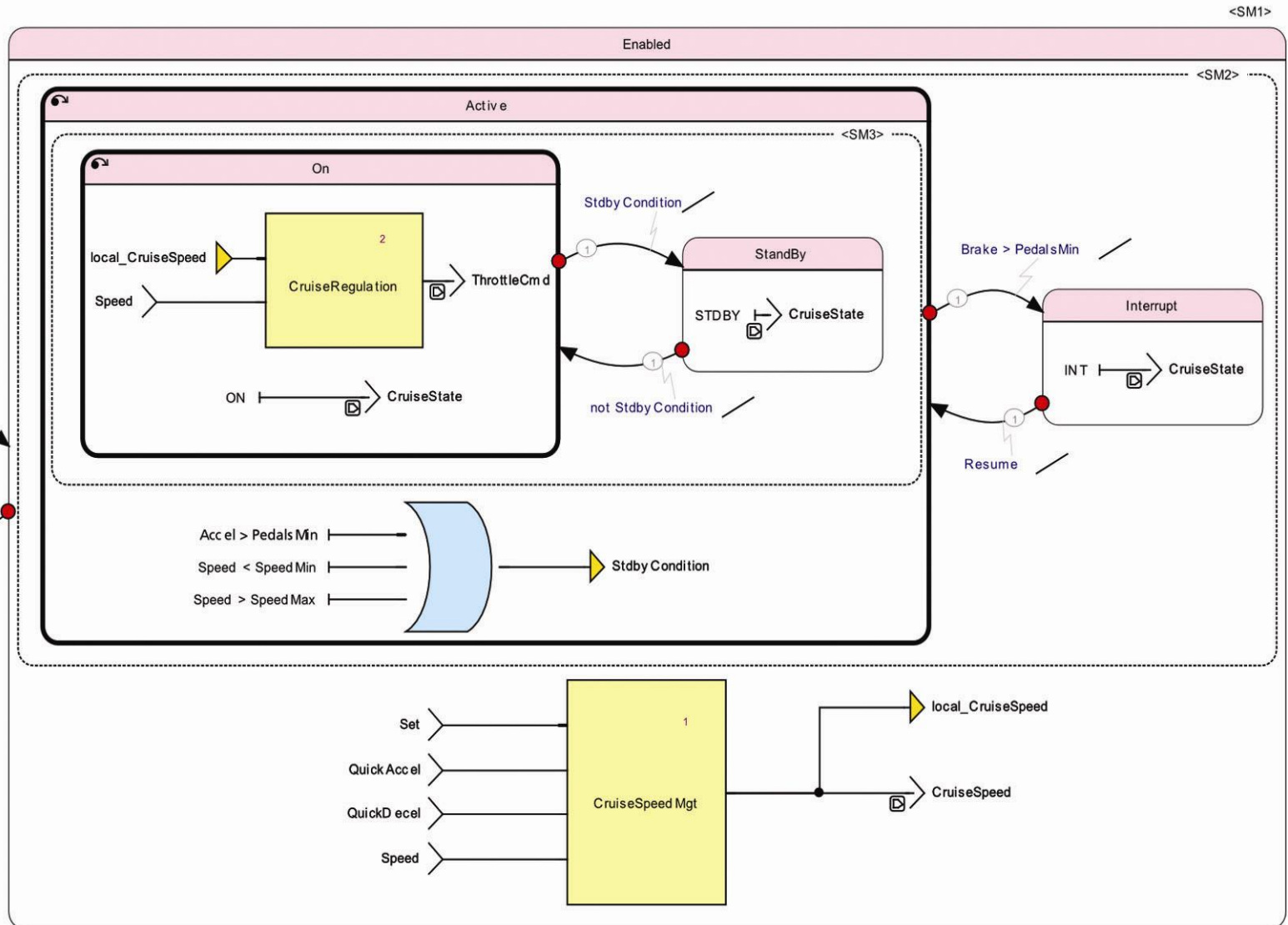
А
В
Т
О

П
У

Средства проектирования алгоритмов

- Алгоритмы бортового ПО хорошо формализуются
 - Поточковая обработка данных
 - Конечный автомат
- Проектирование/описание на формальном уровне позволяет формализовать проверку алгоритма
- Необходимая функциональность:
 - Поддержка обоих видов формального представления
 - Графическое описание
 - Тестирование и пошаговая отладка на уровне модели
 - Верификация на основе формальных методов
 - Сертифицированный кодогенератор
- Примеры: Telelogic Rhapsody, SCADE Suite, Simulink

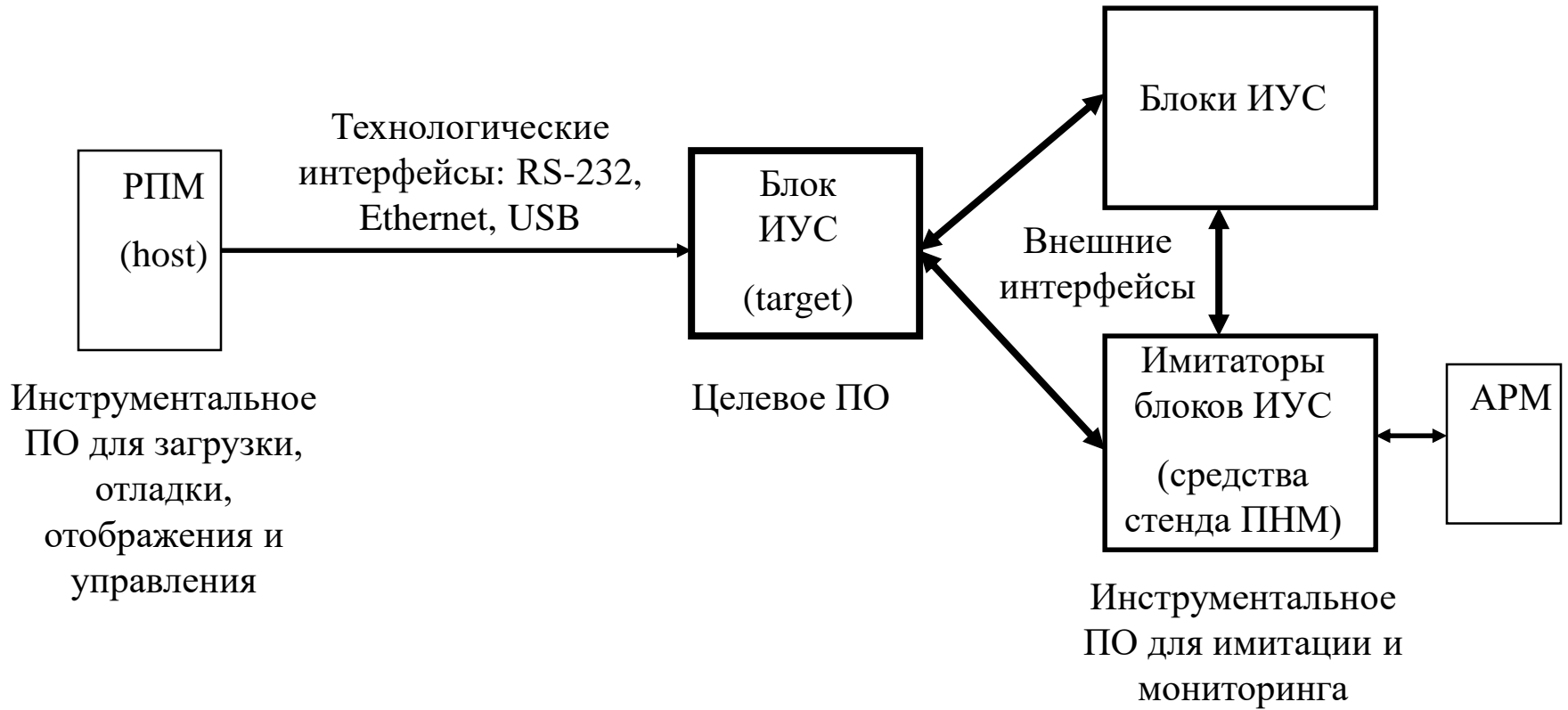
Диаграмма SCADE



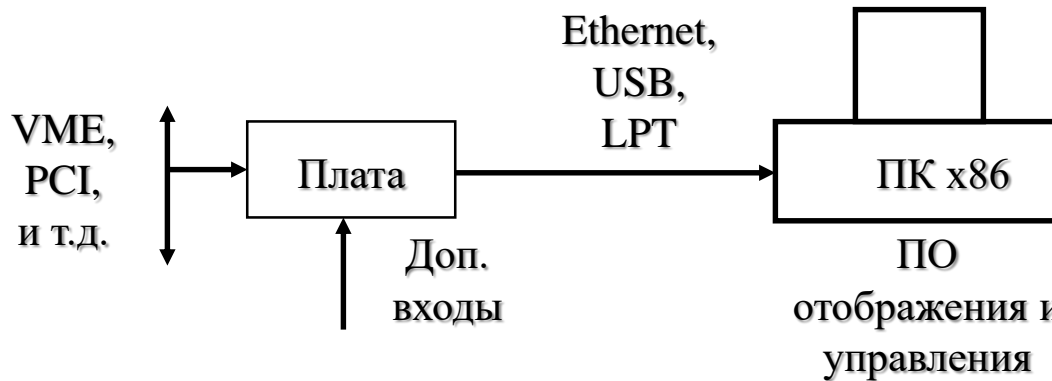
Технологический комплекс разработки программ

- Поддержка целевой ОС и аппаратной архитектуры
 - Поддержка редактирования/компиляции/компоновки программ
 - Поддержка отладки
 - В среде инструментального ПК
 - В среде эмулятора целевой системы
 - На целевой системе
 - Поддержка отладки в реальном времени
 - Мониторинг внешних каналов связи
 - Мониторинг внутренних данных программы
 - Мониторинг системных шин БЦВМ
-

Отладка ПО ИУС РВ на реальном блоке

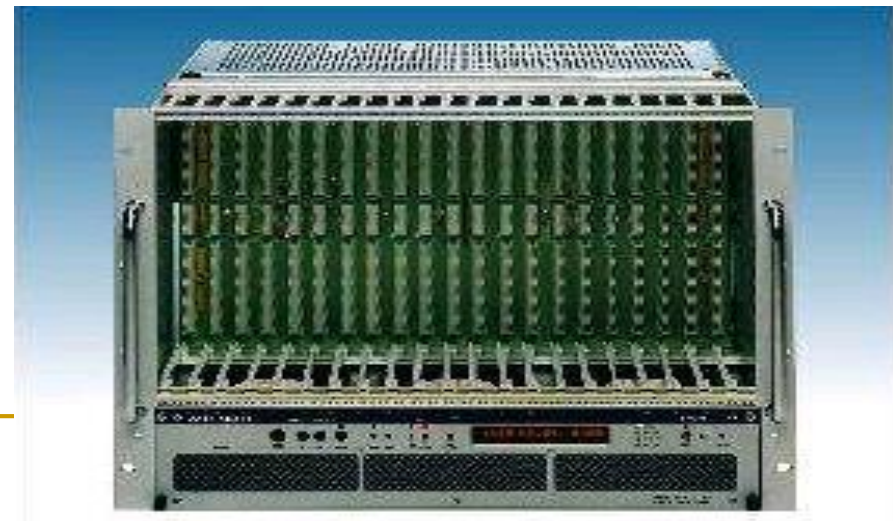


Анализаторы шин VME/PCI



Инструментальные средства анализа:

- Silicon Control Inc
- Curtiss-Wright Electronics systems / VMETRO
- LeCroy Inc
- Tektronix Inc
- Гранит-ВТ



Анализатор шины VME: таблица обменов

Right click to change signal properties

Trace Setups

Power Zoom

Start and Stop capture

Sample before and after trigger

Analyzer status

Marker Times

Trace - test2

62.40us 91.51us 29.11us

Setup address ?

Go Stop

0000100 ←→ 0000100 L1 SAVED

SAMPLE	COMMAND	ADDRESS32	DATA32	AM	TIME	DELTA	BUS DOCTOR	AM[5:0]	BR[3:0]	
-11	Write	0047FFEA	55555555	A32 SUPV DATA	10.54us			0D	F	5555
-10	Write	0047FFEC	55555555	A32 SUPV DATA	11.20us			0D	F	5555
-9	Write	0047FFEE	55555555	A32 SUPV DATA	12.01us	■		0D	F	5555
-8	Write	0047FFF0	55555555	A32 SUPV DATA	8.775us			0D	F	5555
-7	Write	0047FFF2	55555555	A32 SUPV DATA	9.405us			0D	F	5555
-6	Write	0047FFF4	55555555	A32 SUPV DATA	10.06us	62.40us		0D	F	5555
-5	Write	0047FFF6	55555555	A32 SUPV DATA	10.72us			0D	F	5555
-4	Write	0047FFF8	55555555	A32 SUPV DATA	11.38us			0D	F	5555
-3	Write	0047FFFA	55555555	A32 SUPV DATA	12.04us	▲		0D	F	5555
-2	Write	0047FFFC	55555555	A32 SUPV DATA	9.045us			0D	F	5555
-1	Write	0047FFFE	55555555	A32 SUPV DATA	9.705us			0D	F	5555
0	Write	00480000	55555555	A32 SUPV DATA	10.36us	┐		0D	F	5555
1	Write	00480002	55555555	A32 SUPV DATA	11.02us			0D	F	5555
2	Write	00480004	55555555	A32 SUPV DATA	11.68us			0D	F	5555
3	Write	00480006	55555555	A32 SUPV DATA	8.505us			0D	F	5555
4	Write	00480008	55555555	A32 SUPV DATA	9.165us			0D	F	5555
5	Write	0048000A	55555555	A32 SUPV DATA	9.975us			0D	F	5555

Signal scroll bar

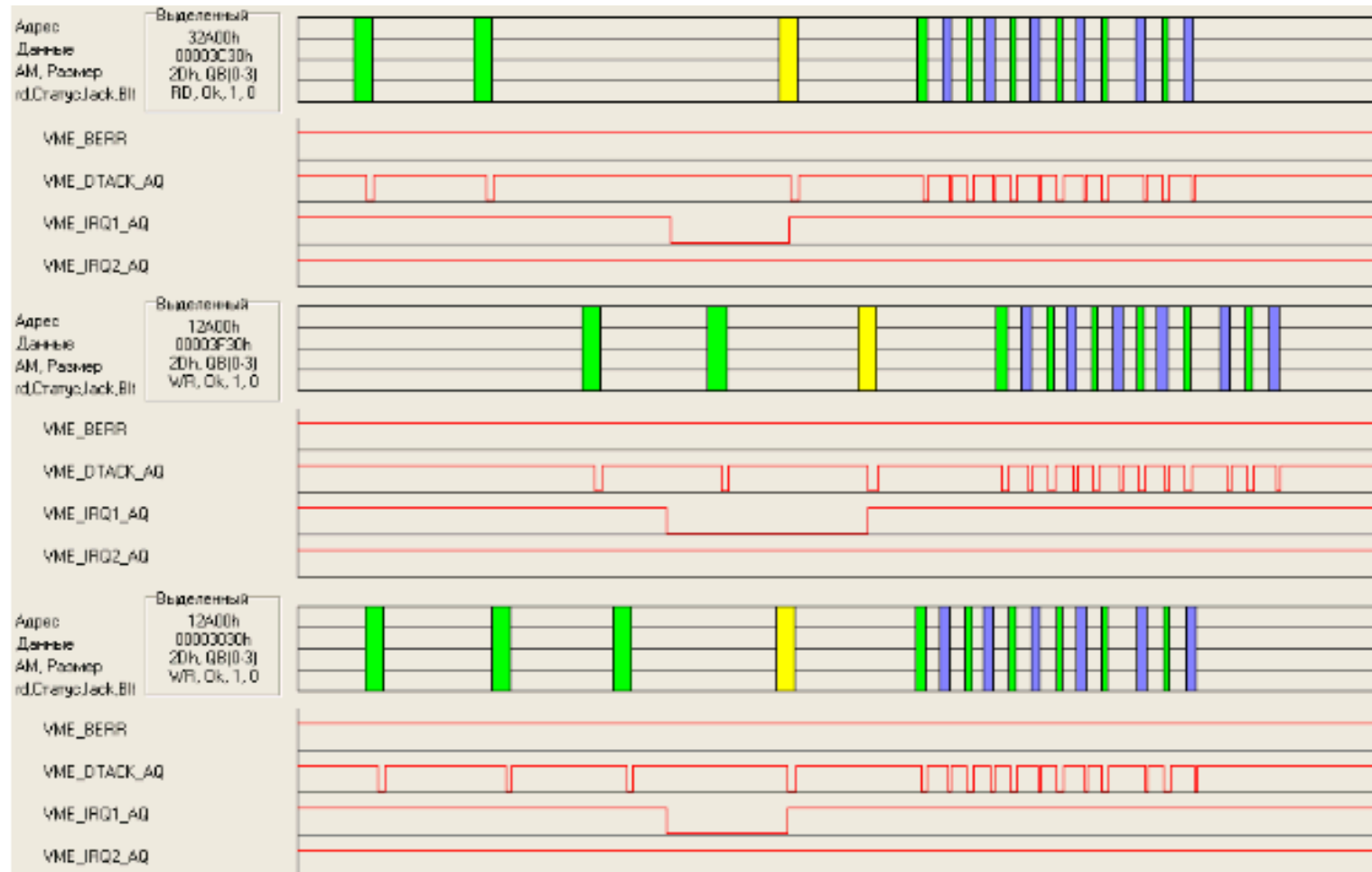
Trigger marker

Marker 1 Left click

Marker 2 Right Click

Sample scroll bar

Анализатор шины VME: временная диаграмма



Анализатор канала MIL STD-1553B

Текущий сеанс: "2012-05-05 14:31" <@standvm-jenny>

5 мая 2012 Время: 11:01.817125 - 12:01.817141 Длительность: 01:00.000016 Ч:М

Каналы

Имя	Тип	ОН	P	B	ОУ
mil	МКИО	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
mil1	МКИО	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
mil2	МКИО	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Свойство Фильтр

Имя	значение
Циклограмма	По умолчанию
Имя	mil
Тип	M1553
МК	<input checked="" type="checkbox"/>
Размещение	hamster:2
Фильтр	По умолчанию
КК	<input checked="" type="checkbox"/>
Размещение	hamster:2
ОУ	<input checked="" type="checkbox"/>
Размещение	hamster:2
Фильтр	По умолчанию

Добавить МК Добавить КК

Добавить ОУ Удалить

[МКИО] mils (180) Интервал: 15:07:20.817125 - 15:08:20.817141 Обмен Адрес ОУ Per Статистика КК

№	Время	Дата	Δt	Формат	Отправитель	Получатель	Размер
4600	15:07:53.815...	5 Май 2012	0:00:00.9991...	1 (КК->ОУ)	КК	0x01:0x01	32
4601	15:07:53.816...	5 Май 2012	0:00:00.0000...	2 (ОУ->КК)	0x01:0x01	КК	0
4602	15:07:53.816...	5 Май 2012	0:00:00.0000...	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	0
Пауза: 0:00:00.999144							
4603	15:07:54.815...	5 Май 2012	0:00:00.9991...	1 (КК->ОУ)	КК	0x01:0x01	32
4604	15:07:54.816...	5 Май 2012	0:00:00.0000...	2 (ОУ->КК)	0x01:0x01	КК	0
4605	15:07:54.816...	5 Май 2012	0:00:00.0000...	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	0
Пауза: 0:00:00.999119							
4606	15:07:55.815...	5 Май 2012	0:00:00.9991...	1 (КК->ОУ)	КК	0x01:0x01	32
4607	15:07:55.816...	5 Май 2012	0:00:00.0000...	2 (ОУ->КК)	0x01:0x01	КК	32
4608	15:07:55.817...	5 Май 2012	0:00:00.0000...	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	32
Пауза: 0:00:00.997816							
4609	15:07:56.815...	5 Май 2012	0:00:00.9978...	1 (КК->ОУ)	КК	0x01:0x01	32
4610	15:07:56.816...	5 Май 2012	0:00:00.0000...	2 (ОУ->КК)	0x01:0x01	КК	32
4611	15:07:56.817...	5 Май 2012	0:00:00.0000...	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	32

[МКИО] mils (181) 0.28% Интервал: 15:11:01.817125 - 15:12:01.817141 Обмен Адрес ОУ Per Статистика КК

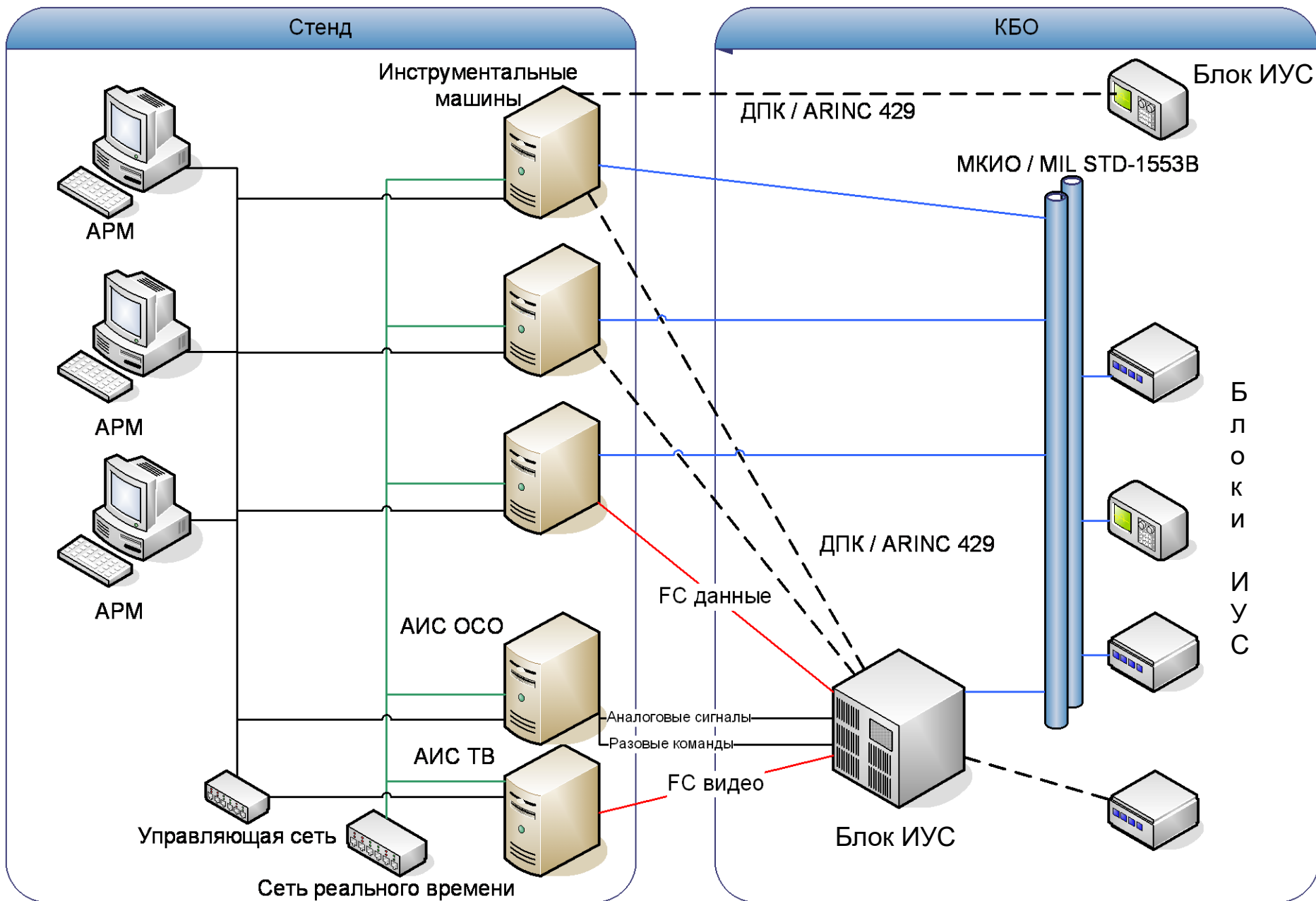
№	Время	Дата	Δt	Формат	Отправитель	Получатель	Размер
5166	15:11:01.817161	5 Май 2012	0:00:00.000000	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	
Пауза: 0:00:00.997801							
5167	15:11:02.815692	5 Май 2012	0:00:00.997801	1 (КК->ОУ)	КК	0x01:0x01	
5168	15:11:02.816413	5 Май 2012	0:00:00.000036	2 (ОУ->КК)	0x01:0x01	КК	
5169	15:11:02.817140	5 Май 2012	0:00:00.000042	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	
Пауза: 0:00:00.997821							
5170	15:11:03.815691	5 Май 2012	0:00:00.997821	1 (КК->ОУ)	КК	0x01:0x01	
5171	15:11:03.816413	5 Май 2012	0:00:00.000037	2 (ОУ->КК)	0x01:0x01	КК	
5172	15:11:03.817140	5 Май 2012	0:00:00.000042	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	
Пауза: 0:00:00.997822							
5173	15:11:04.815692	5 Май 2012	0:00:00.997822	1 (КК->ОУ)	КК	0x01:0x01	
5174	15:11:04.816413	5 Май 2012	0:00:00.000036	2 (ОУ->КК)	0x01:0x01	КК	
5175	15:11:04.817151	5 Май 2012	0:00:00.000053	3 (ОУ->ОУ)	0x00:0x01	0x01:0x01	
Пауза: 0:00:00.997811							
5176	15:11:05.815692	5 Май 2012	0:00:00.997811	1 (КК->ОУ)	КК	0x01:0x01	

14:27:26: Сеанс "Default" успешно загружен.

Средства поддержки верификации и тестирования бортового ПО

- Тестирование на целевой платформе
 - Недопустимость инструментирования
 - Тестирование через каналы бортовых интерфейсов
 - Тестирование требований реального времени
 - Интерактивное тестирование индикационных форматов
- Многоэтапное тестирование
 - Сопровождение интеграции подсистем КБО
- Необходимая функциональность:
 - Поддержка стандартов бортовых интерфейсов
 - Многомашинные конфигурации
 - Выполнение тестов в реальном времени
 - Автоматическое и интерактивное тестирование
 - Пакетный режим
 - Формирование отчётов, прослеживаемость требований
- Примеры средств: Rational Test RealTime, VectorCast, средства разработки ЛВК

Архитектура стенда тестирования ИУС

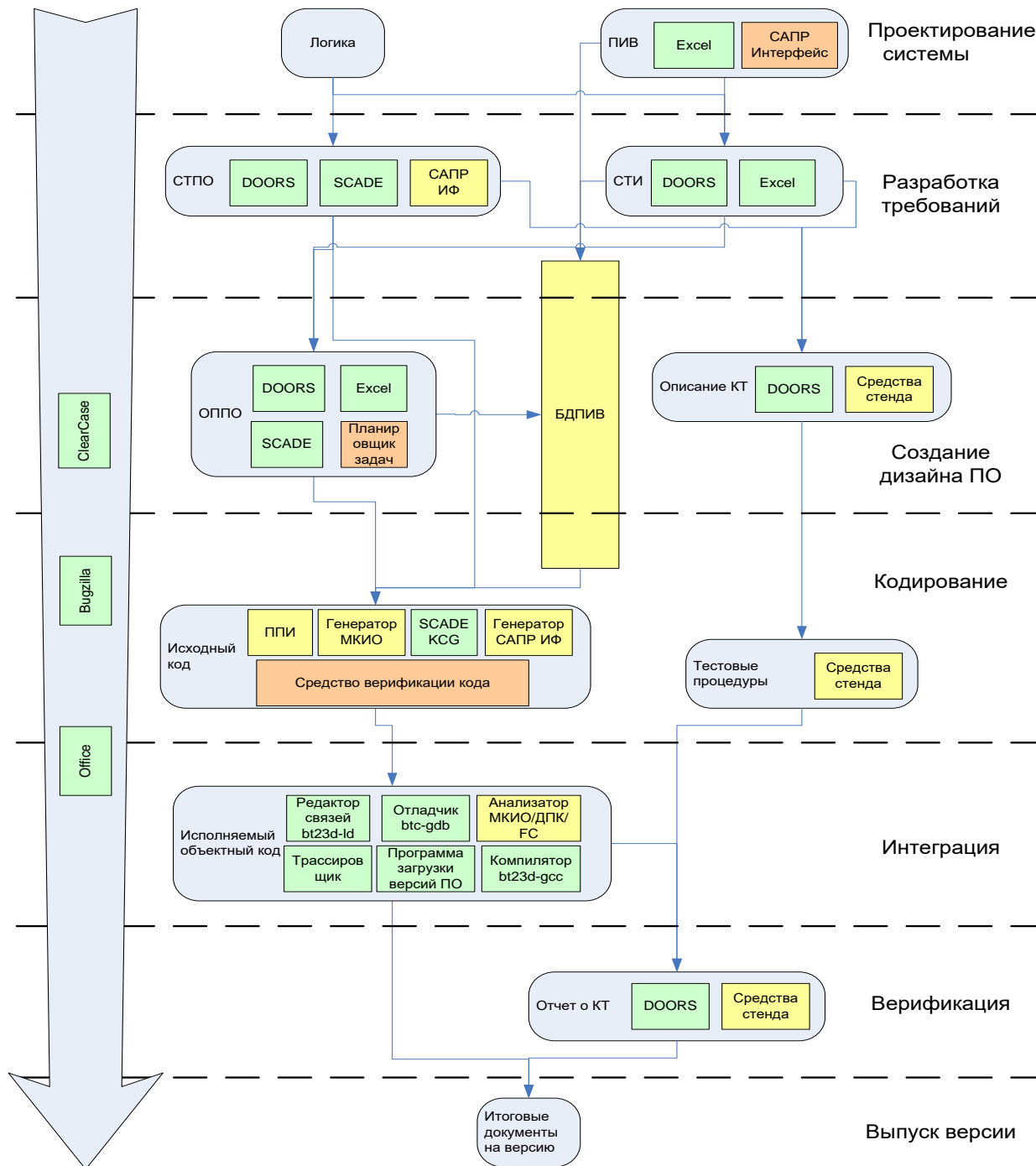


Технологическая цепочка средств разработки бортового ПО

Принцип построения технологической цепочки

- Сквозная поддержка ЖЦ, включая активности на всех фазах
- Сопряжение «вход-выход» с обеспечением совместимости форматов данных
- Особое внимание на переходы между фазами
 - Требуется фиксация выходных артефактов

Цепочка средств разработки бортового ПО



Уровни критичности бортового ПО и квалификация инструментария

Уровни критичности бортового ПО

Уровень	Последствия отказа
A	может привести к катастрофическому отказному состоянию системы и воздушного судна (выход из строя систем управления рулями высоты, двигателями, ...)
B	может привести к аварийному отказному состоянию системы и воздушного судна (выход из строя основных дисплеев кабины, нарушение герметичности, ...)
C	может привести к сложному отказному состоянию системы и воздушного судна (выход из строя систем навигации, радиосвязи, ...)
D	может привести к отказному состоянию типа «усложнение условий полета» воздушного судна (выход из строя систем освещения, внутреннего оповещения, ...)
E	может привести к отказному состоянию без влияния на эксплуатационные возможности воздушного судна (выход из строя развлекательных подсистем, средств доступа в Интернет, ...)

Уровни критичности программных инструментов

- Инструмент, выходные данные которого являются частью бортового ПО (следовательно, он может внести ошибку)
- Инструмент, который автоматизирует процесс(ы) верификации, который может не выявить ошибки, и выходные данные которого используются для подтверждения исключения или усечения [некоторых]:
 - процессов верификации бортового ПО, *помимо* непосредственно автоматизируемых данным инструментом
 - процессов разработки бортового ПО
- Инструмент, который в рамках своего целевого назначения может не выявить ошибку в бортовом ПО

Зависимость уровня квалификации от критичности бортового ПО и инструмента

Уровень ПО	Уровень критерия		
	1	2	3
A	TQL-1	TQL-4	TQL-5
B	TQL-2	TQL-4	TQL-5
C	TQL-3	TQL-5	TQL-5
D	TQL-4	TQL-5	TQL-5

Хорошие новости

- ПО ИУС РВ требует систематического подхода к проектированию, разработке и тестированию
- Сложность и критичность ПО ИУС РВ определяет организационную и технологическую сложность процесса его разработки
- Выполнено организационное структурирование жизненного цикла ПО ИУС РВ по фазам, процессам, активностям
- Построена и внедрена технологическая цепочка средств разработки ПО ИУС РВ, применяемая в соответствии со структурой ЖЦ
- Разработана и внедрена линейка стендов отработки ПО ИУС РВ

Спасибо за внимание

Балашов В.В.
к.ф.-м.н., с.н.с.
hbd@cs.msu.su