# Подходы и инструменты поддержки безопасной разработки ПО реального времени для КВО

Сергей Зыль

Заместитель генерального директора ООО «СВД ВС» по научной работе, к.т.н.

СВД
Встраиваемые Системы

# Модель уровней защиты в перерабатывающей промышленности (МЭК 61511)

Процесс или УО

Мониторинг и управление (РСУ)

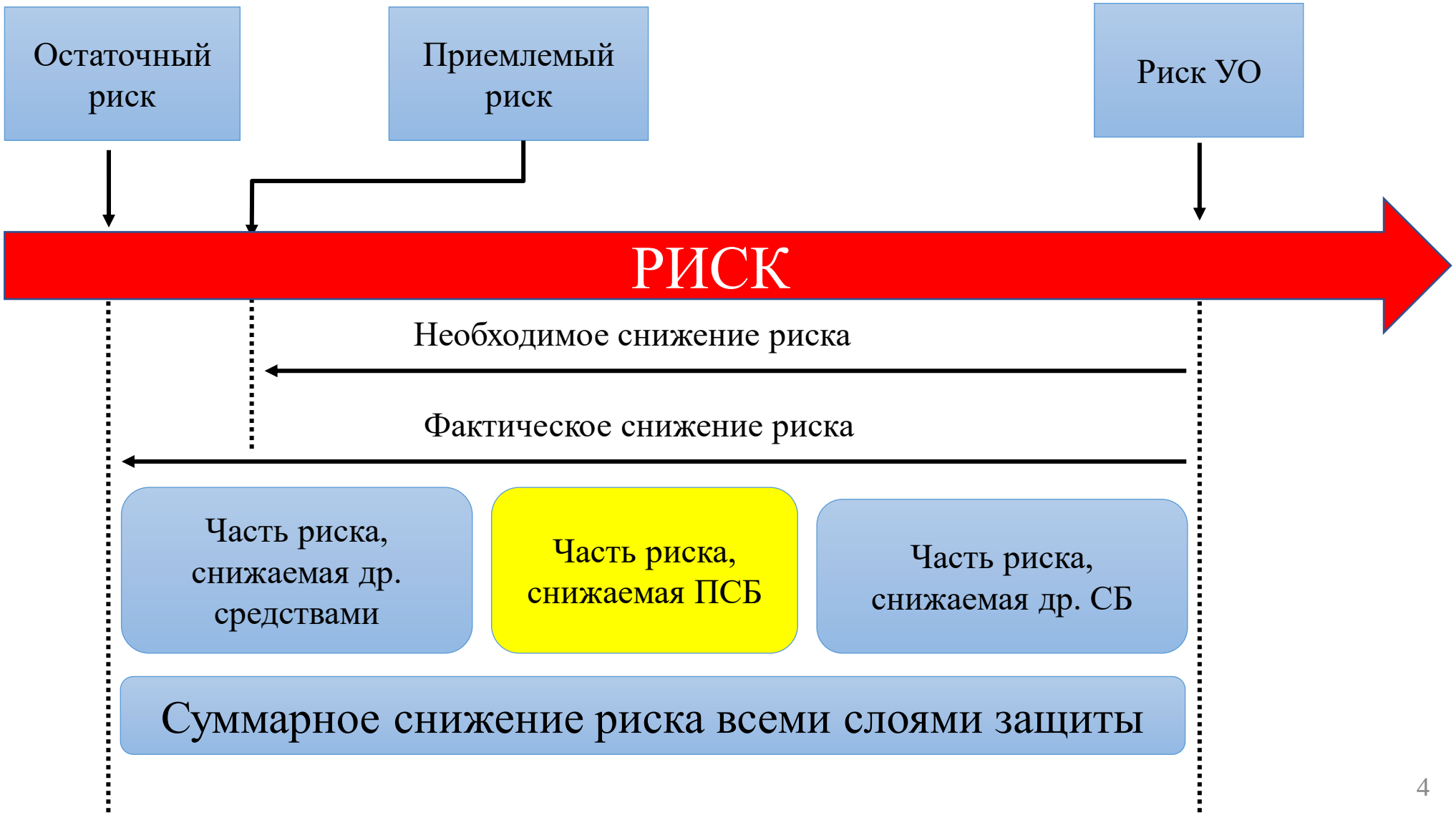
Защита (ПСБ)

Ограничение последствий

Спасательные меры

СВД
Встраиваемые Системы

# Функциональная безопасность – инструмент снижения риска




СВД
Встраиваемые Системы

# Базовый стандарт и разнообразие областей применения

## ГОСТ Р МЭК 61508 – базовый стандарт для ПСБ

СВД
Встраиваемые
Системы

# Уровни полноты безопасности (SIL)

SIL4 – поезд, самолёт

SIL3 – лифт, автомобиль

SIL2 – промышленный робот, станок

SIL1 – защита от перегрева двигателя, storm-position


Таблица 2 — Уровни полноты безопасности: целевая мера отказов для функции безопасности, работающей в режиме низкой интенсивности запросов

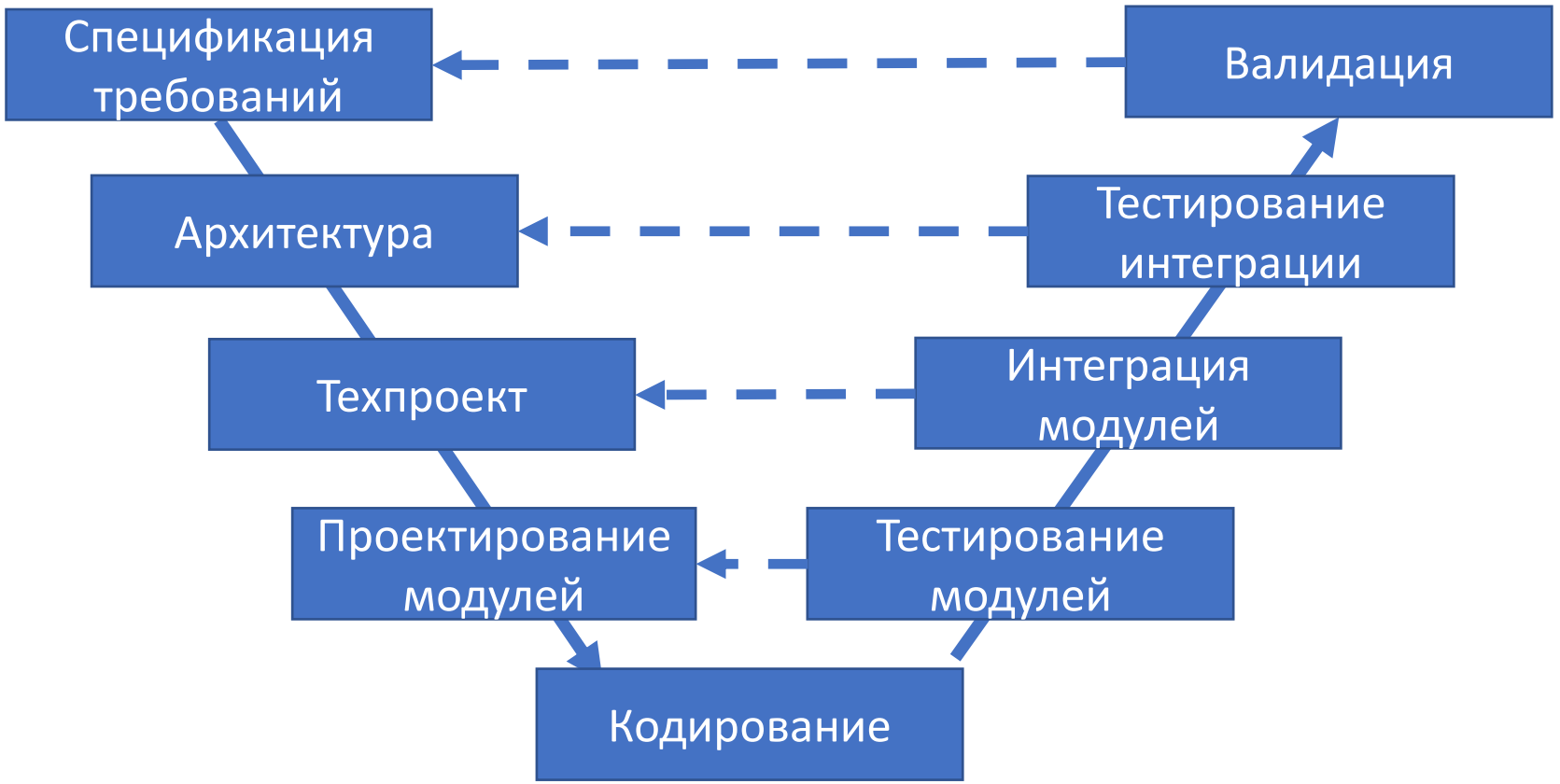
| Уровень полноты безопасности | Средняя вероятность опасного отказа функции безопасности по запросу ($PFD_{avg}$) |
|---|---|
| 4 | $> 10^{-5} — < 10^{-4}$ |
| 3 | $> 10^{-4} — < 10^{-3}$ |
| 2 | $> 10^{-3} — < 10^{-2}$ |
| 1 | $> 10^{-2} — < 10^{-1}$ |

Таблица 3 — Уровни полноты безопасности: целевая мера отказов для функции безопасности, работающей в режиме высокой интенсивности запросов или в режиме с непрерывным запросом

| Уровень полноты безопасности | Средняя частота опасных отказов функции безопасности [$h^{-1}$] (PFH) |
|---|---|
| 4 | $> 10^{-9} — < 10^{-8}$ |
| 3 | $> 10^{-8} — < 10^{-7}$ |
| 2 | $> 10^{-7} — < 10^{-6}$ |
| 1 | $> 10^{-6} — < 10^{-5}$ |

СВД
Встраиваемые
Системы

# V-модель разработки ПО




СВД
Встраиваемые Системы

# Методы спецификации требований

Таблица А.1 - Спецификация требований к программному обеспечению системы безопасности (см. 7.2)

| Метод/средство [1)] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1a Полуформальные методы | Таблица В.7 | R | R | HR | HR |
| 1b Формальные методы | В.2.2, С.2.4 | - | R | R | HR |
| 2 Прямая прослеживаемость между требованиями к системе безопасности и требованиями к программному обеспечению системы безопасности | С.2.11 | R | R | HR | HR |
| 3 Обратная прослеживаемость между требованиями к системе безопасности и предполагаемыми потребностями безопасности | С.2.11 | R | R | HR | HR |
| 4 Компьютерные средства разработки спецификаций для поддержки, перечисленных выше подходящих методов/средств | В.2.4 | R | R | HR | HR |

Таблица В.7 - Полуформальные методы (см. таблицы А.1, А.2 и А.4 приложения А)

| Метод/средство [1)] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Логические/функциональные блок-схемы | См. примечание 1 | R | R | HR | HR |
| 2 Диаграммы последовательности действий | См. примечание 1 | R | R | HR | HR |
| 3 Диаграммы потоков данных | С.2.2 | R | R | R | R |
| 4a Конечные автоматы/диаграммы переходов | В.2.3.2 | R | R | HR | HR |
| 4b Моделирование во времени сетями Петри | В.2.3.3 | R | R | HR | HR |
| 5 Модели данных сущность-связь-атрибут | В.2.4.4 | R | R | R | R |
| 6 Диаграммы последовательности сообщений | С.2.14 | R | R | R | R |
| 7 Таблицы решений и таблицы истинности | С.6.1 | R | R | HR | HR |
| 8 UML | С.3.12 | R | R | R | R |

СВД
Встраиваемые Системы

# Методы архитектурного проектирования

| | | | | | |
|---|---|---|---|---|---|
| 7 Модульный подход | Таблица B.9 | HR | HR | HR | HR |
| 8 Использование доверительных/проверенных элементов программного обеспечения (при наличии) | C.2.10 | R | HR | HR | HR |
| 9 Прямая прослеживаемость между спецификацией требований к программному обеспечению системы безопасности и архитектурой программного обеспечения | C.2.11 | R | R | HR | HR |
| 10 Обратная прослеживаемость между спецификацией требований к программному обеспечению системы безопасности и архитектурой программного обеспечения | C.2.11 | R | R | HR | HR |
| 11a Методы структурных диаграмм 2) | C.2.1 | HR | HR | HR | HR |
| 11b Полуформальные методы 2) | Таблица B.7 | R | R | HR | HR |
| 11c Формальные методы проектирования и усовершенствования 2) | B.2.2, C.2.4 | - | R | R | HR |
| 11d Автоматическая генерация программного обеспечения | C.4.6 | R | R | R | R |
| 12 Автоматизированные средства разработки спецификаций и проектирования | B.2.4 | R | R | HR | HR |
| 13a Циклическое поведение с гарантированным максимальным временем цикла | C.3.11 | R | HR | HR | HR |
| 13b Архитектура с временным распределением | C.3.11 | R | HR | HR | HR |
| 13c Управление событиями с гарантированным максимальным временем реакции | C.3.11 | R | HR | HR | - |
| 14 Статическое выделение ресурсов | C.2.6.3 | - | R | HR | HR |
| 15 Статическая синхронизация доступа к разделяемым ресурсам | C.2.6.3 | - | - | R | HR |

Таблица B.9 - Модульный подход (см. таблицу A.4 приложения A)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Ограничение размера программного модуля | C.2.9 | HR | HR | HR | HR |
| 2 Управление сложностью программного обеспечения | C.5.13 | R | R | HR | HR |
| 3 Ограничение доступа/инкапсуляция информации | C.2.8 | R | HR | HR | HR |
| 4 Ограниченное число параметров/фиксированное число параметров подпрограммы | C.2.9 | R | R | R | R |
| 5 Одна точка входа и одна точка выхода в каждой подпрограмме и функции | C.2.9 | HR | HR | HR | HR |
| 6 Полностью определенный интерфейс | C.2.9 | HR | HR | HR | HR |

1) Методы/средства следует выбирать в соответствии с уровнем полноты безопасности. Использование одного метода является, по-видимому, недостаточным. Следует рассматривать все подходящие методы.

Примечание 1 - См. таблицу C.19.
Примечание 2 - Ссылки (являющиеся справочными, а не обязательными) "B.x.x.x", "C.x.x.x" в столбце 2 указывают на подробные описания методов/средств, изложенных в приложениях B и C [5].

Таблица A.3 - Проектирование и разработка программного обеспечения: инструментальные средства поддержки и языки программирования (см. 7.4.4)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Выбор соответствующего языка программирования | C.4.5 | HR | HR | HR | HR |
| 2 Строго типизированные языки программирования | C.4.1 | HR | HR | HR | HR |
| 3 Подмножество языка | C.4.2 | - | - | HR | HR |
| 4a Сертифицированные средства и сертифицированные трансляторы | C.4.3 | R | HR | HR | HR |
| 4b Инструментальные средства, заслуживающие доверия на основании опыта использования | C.4.4 | HR | HR | HR | HR |

СВД
Встраиваемые Системы

# Методы технического проектирования

Таблица А.4 — Проектирование и разработка программного обеспечения: детальное проектирование (см. 7.4.5 и 7.4.6) (включает в себя проектирование системы программного обеспечения, проектирование модуля программного обеспечения и кодирование)

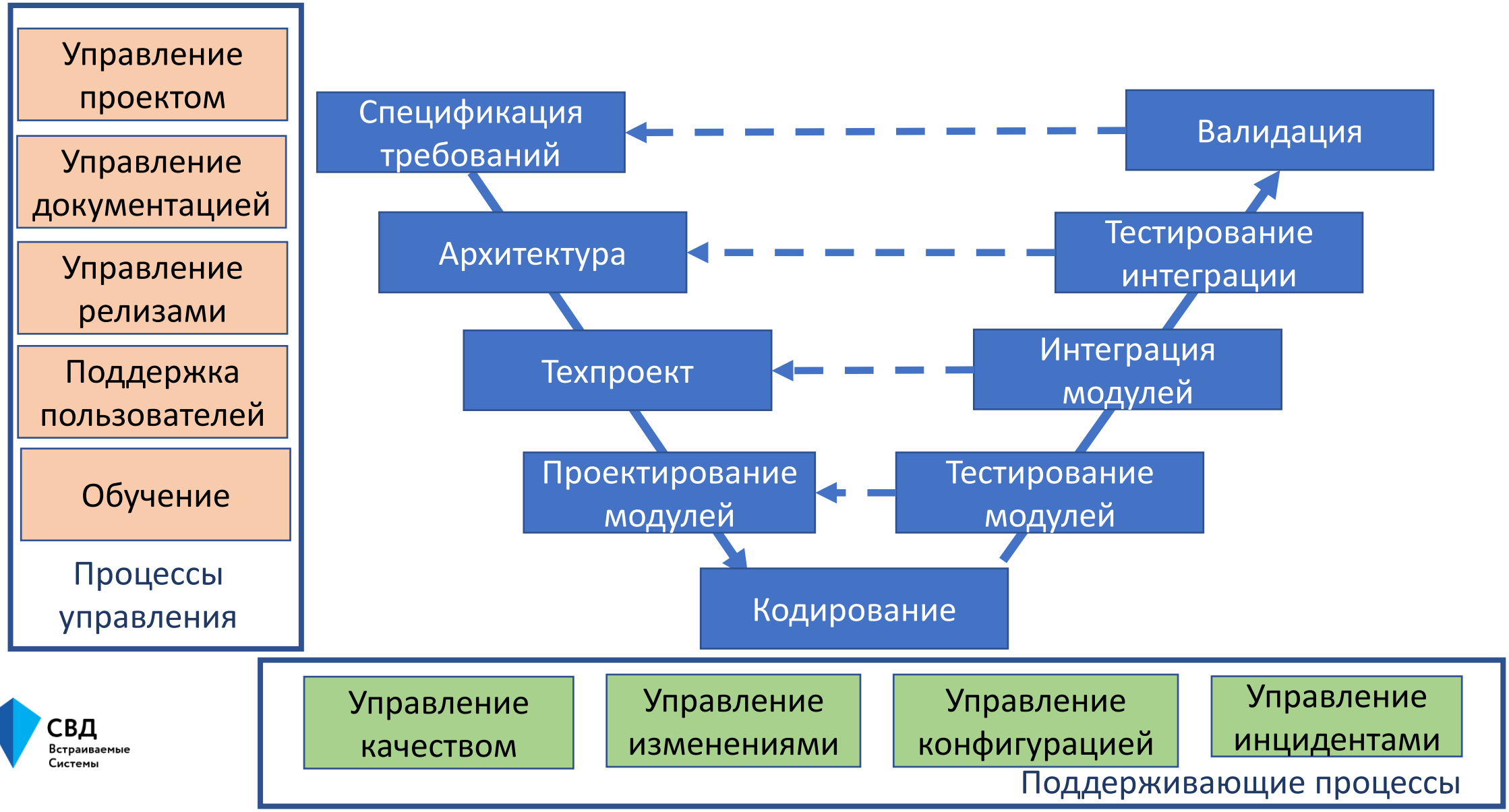
| Метод/средство[1)] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1a Методы структурных диаграмм[2)] | С.2.1 | HR | HR | HR | HR |
| 1b Полуформальные методы[2)] | Таблица В.7 | R | R | HR | HR |
| 1c Формальные методы проектирования и усовершенствования[2)] | В.2.2, С.2.4 | — | R | R | HR |
| 2 Средства автоматизированного проектирования | В.3.5 | R | R | HR | HR |
| 3 Программирование с защитой | С.2.5 | — | R | HR | HR |
| 4 Модульный подход | Таблица В.9 | HR | HR | HR | HR |
| 5 Стандарты для проектирования и кодирования | С.2.6, таблица В.1 | R | HR | HR | HR |
| 6 Структурное программирование | С.2.7 | HR | HR | HR | HR |
| 7 Использование доверительных/проверенных программных модулей и компонентов (по возможности) | С.2.10 | R | HR | HR | HR |
| 8 Прямая прослеживаемость между спецификацией требований к программному обеспечению системы безопасности и проектом программного обеспечения | С.2.11 | R | R | HR | HR |

СВД Встраиваемые Системы

# Стандарты кодирования

| Метод/средство[1)] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Использование стандартов кодирования для сокращения вероятности ошибок | C.2.6.2 | HR | HR | HR | HR |
| 2 Не использовать динамические объекты | C.2.6.3 | R | HR | HR | HR |
| 3a Не использовать динамические переменные | C.2.6.3 | — | R | HR | HR |
| 3b Проверка создания динамических переменных в неавтономном режиме | C.2.6.4 | — | R | HR | HR |
| 4 Ограниченное использование прерываний | C.2.6.5 | R | R | HR | HR |
| 5 Ограниченное использование указателей | C.2.6.6 | — | R | HR | HR |
| 6 Ограниченное использование рекурсий | C.2.6.7 | — | R | HR | HR |
| 7 Не использовать неструктурированное управление в программах, написанных на языках высокого уровня | C.2.6.2 | R | HR | HR | HR |
| 8 Не использовать автоматическое преобразование типов | C.2.6.2 | R | HR | HR | HR |

# V-модель разработки ПО




СВД
Встраиваемые Системы

# V-модель разработки ПО


СВД
Встраиваемые Системы

# Классы off-line инструментов МЭК 61508

- T1 – не генерирует выходные данные, которые явно или не явно влияют на исполняемый код системы, связанной с безопасностью
- T2 – инструменты тестирования или верификации, которые не могут напрямую вносить ошибки в исполняемый код
- T3 – генерирует выходные данные, которые явно или не явно влияют на исполняемый код системы, связанной с безопасностью

СВД
Встраиваемые Системы

# Рассмотрим два типа инструментов класса Т2

Таблица В.8 - Статический анализ (см. таблицу А.9 приложения А)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Анализ граничных значений | С.5.4 | R | R | HR | HR |
| 2 Таблица контрольных проверок | В.2.5 | R | R | R | R |
| 3 Анализ потоков управления | С.5.9 | R | HR | HR | HR |
| 4 Анализ потоков данных | С.5.10 | R | HR | HR | HR |
| 5 Предположение ошибок | С.5.5 | R | R | R | R |
| 6a Формальные проверки, включая конкретные критерии | С.5.14 | R | R | HR | HR |
| 6b Сквозной контроль (программного обеспечения) | С.5.15 | R | R | R | R |
| 7 Тестирование на символьном уровне | С.5.11 | - | - | R | R |
| 8 Анализ проекта | С.5.16 | HR | HR | HR | HR |
| 9 Статический анализ выполнения программы с ошибкой | В.2.2, С.2.4 | R | R | R | HR |
| 10 Временной анализ выполнения при наихудших условиях | С.5.20 | R | R | R | R |

Статический анализ –
без выполнения программы

Динамический анализ –
требуется выполнение кода

Таблица В.2 - Динамический анализ и тестирование (см. таблицы А.5 и А.9 приложения А)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Выполнение тестового примера, связанного с анализом граничных значений | С.5.4 | R | HR | HR | HR |
| 2 Выполнение тестового примера, связанного с предполагаемой ошибкой | С.5.5 | R | R | R | R |
| 3 Выполнение тестового примера, связанного с введением ошибки | С.5.6 | - | R | R | R |
| 4 Выполнение тестового примера, сгенерированного на основе модели | С.5.27 | R | R | HR | HR |
| 5 Моделирование реализации | С.5.20 | R | R | R | HR |
| 6 Разделение входных данных на классы эквивалентности | С.5.7 | R | R | R | HR |
| 7a Структурный тест со 100%-ным охватом (точки входа) 2) | С.5.8 | HR | HR | HR | HR |
| 7b Структурный тест со 100%-ным охватом (операторы) 2) | С.5.8 | R | HR | HR | HR |
| 7c Структурный тест со 100%-ным охватом (условные переходы) 2) | С.5.8 | R | R | HR | HR |
| 7d Структурный тест со 100%-ным охватом (составные условия, MC/DC) 2) | С.5.8 | R | R | R | HR |

СВД
Встраиваемые
Системы

# Рассмотрим два типа инструментов

Таблица B.8 - Статический анализ (см. таблицу A.9 приложения A)

| Метод/средство [1] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Анализ граничных значений | C.5.4 | R | R | HR | HR |
| 2 Таблица контрольных проверок | B.2.5 | R | R | R | R |
| 3 Анализ потоков управления | C.5.9 | R | HR | HR | HR |
| 4 Анализ потоков данных | C.5.10 | R | HR | HR | HR |
| 5 Предположение ошибок | C.5.5 | R | R | R | R |
| 6a Формальные проверки, включая конкретные критерии | C.5.14 | R | R | HR | HR |
| 6b Сквозной контроль (программного обеспечения) | C.5.15 | R | R | R | R |
| 7 Тестирование на символьном уровне | C.5.11 | - | - | R | R |
| 8 Анализ проекта | C.5.16 | HR | HR | HR | HR |
| 9 Статический анализ выполнения программы с ошибкой | B.2.2, C.2.4 | R | R | R | HR |
| 10 Временной анализ выполнения при наихудших условиях | C.5.20 | R | R | R | R |

Статический анализ –
без выполнения программы

Динамический анализ –
требуется выполнение кода

Таблица B.2 - Динамический анализ и тестирование (см. таблицы A.5 и A.9 приложения A)

| Метод/средство [1] | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Выполнение тестового примера, связанного с анализом граничных значений | C.5.4 | R | HR | HR | HR |
| 2 Выполнение тестового примера, связанного с предполагаемой ошибкой | C.5.5 | R | R | R | R |
| 3 Выполнение тестового примера, связанного с введением ошибки | C.5.6 | - | R | R | R |
| 4 Выполнение тестового примера, сгенерированного на основе модели | C.5.27 | R | R | HR | HR |
| 5 Моделирование реализации | C.5.20 | R | R | R | HR |
| 6 Разделение входных данных на классы эквивалентности | C.5.7 | R | R | R | HR |
| 7a Структурный тест со 100%-ным охватом (точки входа) [2] | C.5.8 | HR | HR | HR | HR |
| 7b Структурный тест со 100%-ным охватом (операторы) [2] | C.5.8 | R | HR | HR | HR |
| 7c Структурный тест со 100%-ным охватом (условные переходы) [2] | C.5.8 | R | R | HR | HR |
| 7d Структурный тест со 100%-ным охватом (составные условия, MC/DC) [2] | C.5.8 | R | R | R | HR |

СВД
Встраиваемые
Системы

# Статический анализ

Статические анализаторы способны обнаружить различные ошибки программирования на ранних этапах разработки ПО, например:

- переполнение буфера
- опечатки
- выход за границы
- разыменовывание нулевого указателя
- ошибки доступа к памяти
- использование неинициализированных переменных
- недостижимый код
- утечка информации через сообщения об ошибках
- и пр.

СВД
Встраиваемые Системы

# Статический анализ

- Статический анализатор для кода C, C++, C# и Java
- 15 лет на рынке
- Входит в реестр отечественного ПО (запись № 9837)
- Стандарты надёжности: MISRA и AUTOSAR
- Стандарты: CWE, OWASP ASVS, SEI CERT
- Поддержка встраиваемых ОСРВ: Нейтрино и др.
- Хост-системы: Linux, Astra Linux, macOS, Windows
- Соответствует функциональным требованиям ФСТЭК к инструментам статического анализа


СВД
Встраиваемые Системы

# Рассмотрим два типа инструментов

Таблица B.8 - Статический анализ (см. таблицу A.9 приложения A)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Анализ граничных значений | C.5.4 | R | R | HR | HR |
| 2 Таблица контрольных проверок | B.2.5 | R | R | R | R |
| 3 Анализ потоков управления | C.5.9 | R | HR | HR | HR |
| 4 Анализ потоков данных | C.5.10 | R | HR | HR | HR |
| 5 Предположение ошибок | C.5.5 | R | R | R | R |
| 6a Формальные проверки, включая конкретные критерии | C.5.14 | R | R | HR | HR |
| 6b Сквозной контроль (программного обеспечения) | C.5.15 | R | R | R | R |
| 7 Тестирование на символьном уровне | C.5.11 | - | - | R | R |
| 8 Анализ проекта | C.5.16 | HR | HR | HR | HR |
| 9 Статический анализ выполнения программы с ошибкой | B.2.2, C.2.4 | R | R | R | HR |
| 10 Временной анализ выполнения при наихудших условиях | C.5.20 | R | R | R | R |

Статический анализ –
без выполнения программы

Динамический анализ –
требуется выполнение кода

Таблица B.2 - Динамический анализ и тестирование (см. таблицы A.5 и A.9 приложения A)

| Метод/средство 1) | Ссылка | УПБ1 | УПБ2 | УПБ3 | УПБ4 |
|---|---|---|---|---|---|
| 1 Выполнение тестового примера, связанного с анализом граничных значений | C.5.4 | R | HR | HR | HR |
| 2 Выполнение тестового примера, связанного с предполагаемой ошибкой | C.5.5 | R | R | R | R |
| 3 Выполнение тестового примера, связанного с введением ошибки | C.5.6 | - | R | R | R |
| 4 Выполнение тестового примера, сгенерированного на основе модели | C.5.27 | R | R | HR | HR |
| 5 Моделирование реализации | C.5.20 | R | R | R | HR |
| 6 Разделение входных данных на классы эквивалентности | C.5.7 | R | R | R | HR |
| 7a Структурный тест со 100%-ным охватом (точки входа) 2) | C.5.8 | HR | HR | HR | HR |
| 7b Структурный тест со 100%-ным охватом (операторы) 2) | C.5.8 | R | HR | HR | HR |
| 7c Структурный тест со 100%-ным охватом (условные переходы) 2) | C.5.8 | R | R | HR | HR |
| 7d Структурный тест со 100%-ным охватом (составные условия, MC/DC) 2) | C.5.8 | R | R | R | HR |

СВД
Встраиваемые
системы

# Динамический анализ

Динамический анализ требует применения нескольких инструментов:

- Фаззинг (мутационный, гибридный)
- Трассировка потоков данных
- Модульные тесты
- Инструментирование кода для анализа покрытия
- Нагрузочные тесты
- Тесты СЗИ
- Пентесты

**Для систем реального времени готовых универсальных российских инструментов нет**

СВД
Встраиваемые Системы

# Инструменты, применяемые ООО «СВД ВС»

- Open Project PMS (доработанный)
- Git+Gitea
- Jenkins CI/CD
- Svace (ИСП РАН), cppcheck
- Управление модульными тестами (собственная разработка)
- Фаззинг (AFL и др. – существенно доработанные)
- Трассировка Valgind+Taintgrind (существенно доработанные)
- Трассировка событий микроядра (собственная разработка)
- Система генерации документации (собственная разработка)
- LibLan, Moodle и пр.

СВД
Встраиваемые Системы

# Результаты внедрения безопасной разработки

Разработка ДО…

МЭК 61508

Разработка ПОСЛЕ

СВД
Встраиваемые
Системы

# Спасибо за внимание!

Сергей Зыль

Заместитель генерального директора ООО «СВД ВС»
по научной работе

s.zyl@kpda.ru
+7 904 611 5026

СВД
Встраиваемые
Системы