Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

Киркоров Сергей Иванович

Спонсор научно-практического семинара "Ada-технологии в современной программной индустрии" в рамках выставки PTS-2009

Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

- Почему Ada-технологии позволяют разрабатывать системы с повышенным уровнем безопасности
- Аппаратно-программная платформа и преимущества Ada-технологий
- SPARK технология позволяющая автоматизировать этапы сертификации ПО на языке Ada
- NSA Tokeneer System - пример системы с повышенным уровнем безопасности Common Criterial EAL5 и выше

Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

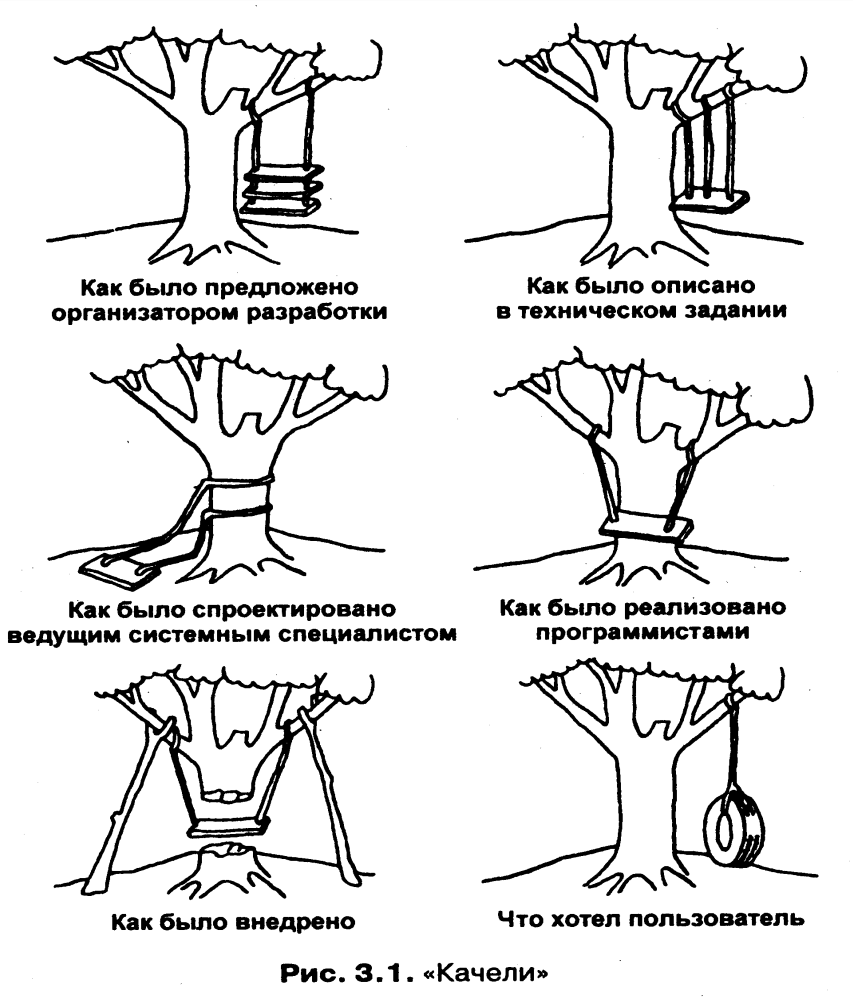


Рис. 3.1. «Качели»

- Известный шарж 70-х годов.
- Язык программирования Ada был разработан 1983 году, чтобы решить в том числе и эти проблемы.

Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

- Существуют прикладные задачи, например, в области технической защиты информации, где требуется как формальное доказательство правильности алгоритма, так и его фактической реализации.
- Моделирование алгоритмов, реализующих параллельные вычисления возможно с применением пакетов типа MatLab и других. Одной из причин ограничений на использование такого инструментария является сложность и высокая стоимость обеспечения соответствующего уровня гарантий конечного продукта в области технической защиты информации по всем трем составляющим – конфиденциальность, целостность и доступность. Это следует из существенной разницы между оптимизированным под прикладную область алгоритмом и его первоначальной математической моделью, реализуемой подобными пакетами.

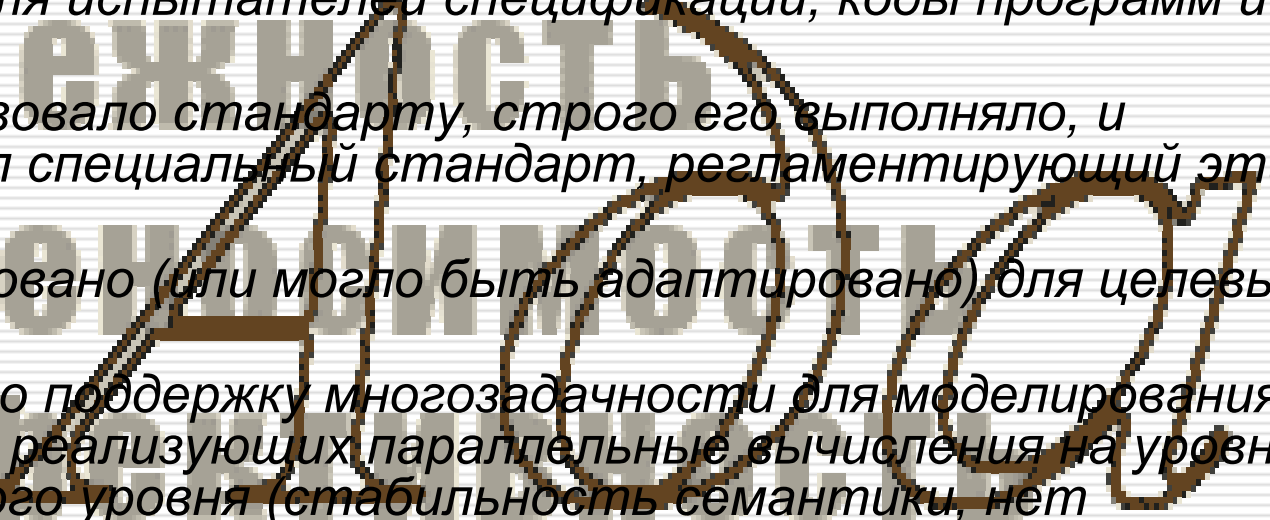
Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

- Под доверенной платформой подразумевается собственный вычислитель, например специализированный процессор с установленной на нем другой операционной системой (или без ОС) или программируемые пользователями вентильные матрицы (Field-Programmable Gate Array – FPGA). .
- Для тестирования доверенной платформы и реализуемых на ней параллельных вычислений, как правило, создается стендовое оборудование. Это стендовое оборудование, рассматриваемое как средство измерения параметров осуществлённой реализации, также требует соответствующего уровня гарантий правильности своего функционирования.

Вопросы сертификации технологий на базе языка Ada по СТБ 34.101.1-3

Правильным выбором в этом случае было бы использовать инструментальное средство, которое:

- *Само могло бы пройти сертификационные испытания в области технической защиты информации, то есть, как минимум имело открытые для испытателей спецификации, коды программ и так далее.*
- *Соответствовало стандарту, строго его выполняло, и существовал специальный стандарт, регламентирующий эти проверки.*
- *Было реализовано (или могло быть адаптировано) для целевых платформ.*
- *Обеспечивало поддержку многозадачности для моделирования алгоритмов, реализующих параллельные вычисления на уровне языка высокого уровня (стабильность семантики, нет необходимости использования разнородных внешних библиотек или собственных решений для обеспечения многозадачности).*



Критичность программного обеспечения

ПО и Критичность

- **Критичность по отношению к бизнес-процессам**
  - Сбой программного обеспечения может привести к значительным финансовым потерям и даже к полной остановке бизнеса
  - Например, система межбанковских платежей

- **Критичность по отношению к решаемой задаче**
  - Сбой программного обеспечения может привести к невыполнимости поставленной задачи
  - Например, спутник для исследования Марса

- **Критичность по отношению к безопасности**
  - Сбой программного обеспечения может привести к человеческим жертвам или большим разрушениям
  - Например, самолет

Стандарты на критичное к безопасности ПО

- **RTCA/EUROCAE DO-178B**
  - Международный стандарт на критичное для безопасности ПО в области авиастроения
- **IEC 880**
  - Стандарт на ПО для атомных электростанций
- **IEC61508 / DEF STAN 00-55/56**
  - Европейский стандарт безопасности
- **Руководство разработчика ПО для транспортных средств**
  - Стандарт безопасности, предложенный Ассоциацией разработчиков безотказного ПО для автомобильной промышленности MISRA (Motor Industry Software Reliability Association)

Уровни критичности ПО согласно DO-178B

Уровень критичности	Последствия от ошибки/сбоя ПО
Уровень A	**Катастрофические** ***(Продукты уровня A сообщают экипажу самолета о его положении в пространстве и предотвращают его от падения, н.п. системы управления полетом, авиационные картографические базы, некоторые дисплеи)***
Уровень B	**Опасные/Значительные** ***(Системы уровня B: слежение за движением и уклонение от столкновений)***
Уровень C	**Большие** ***(Системы уровня C: связь и управление каналами связи)***
Уровень D	**Незначительные** ***(Системы уровня D: системы обеспечения комфорта)***
Уровень E	**Без последствий** ***(Системы уровня E: развлекательные системы)***

IEC61508 Уровни безопасности-сложности-целостности SCIL (Safety-Complexity-Integrity Levels)

Уровень SCIL	Последствия от ошибки/сбоя ПО
SCIL 4	Смерть одного или нескольких людей, существенные финансовые потери *(Область: аэрокосмическая, медицинские системы, системы управления движением, системы управления опасными процессами, системы торможения)*
SCIL 3	Серьезные телесные повреждения или финансовые потери *(Область: управление силовыми установками средств передвижения)*
SCIL 2	Неудобство или недовольство *(Область: кассовые терминалы в супермаркетах, аппараты выдачи сигарет/напитков)*
SCIL 1	Без последствий *(Область: студенческие проекты, исследования)*

Уровни целостности предложенные MISRA

Уровень целост-ности	Возможность контроля со стороны водителя	Оценка допустимости отказа	*Примеры возможных последствий при сбое ПО автомобиля*
4	Не поддается контролю	Абсолютно недопустимо	*Обесточивание усилителя рулевого управления*
3	Сложно контролируется	Чрезвычайно редко	*Отказ тормозной системы*
2	Утомляет	Редко	*Неработоспособность механизма очистки лобового стекла*
1	Отвлекает	Не желательно	*Неработоспособность стеклоподъемника*
0	Только вызывает неудобство	Допускается	*Неработоспособность радио/CD плеера*

Программное обеспечение и безопасность

Стандарты безопасности ПО

- **TCSEC (Оранжевая книга)**
  - Критерии оценки безопасности высоконадежной компьютерной системы
- **Общие критерии оценки безопасности в Информационных технологиях (ИТ) (ISO/IEC 15408-1)**
  - Критерии оценки безопасности ИТ
  - 7 уровней оценки безопасности

Уровни оценки безопасности (EALs)

EAL	Ограничения на разрабатываемое ПО
EAL7	Формально доказанная корректность + тестирование
EAL6	Использование доказательства корректности при проектировании + тестирование
EAL5	Проектирование с использованием формальных методов + тестирование
EAL4	Методологическая проектирование, тестирование и исправление
EAL3	Проведены методологические тесты и проверки
EAL2	Проведен структурный тест
EAL1	Оттестирована функциональность

Содержимое Лицензии Windows 2000

ЗАМЕЧАНИЕ ПО ПОДДЕРЖКЕ JAVA

ДАНОЕ ПРОГРАММНОЕ ИЗДЕЛИЕ МОЖЕТ СОДЕРЖАТЬ ПОДДЕРЖКУ ПРОГРАММ, НАПИСАННЫХ НА JAVA .

ТЕХНОЛОГИЯ JAVA - НЕ УСТОЙЧИВАЯ К СБОЯМ И НЕ РАЗРАБОТАНА, ИЗГОТОВЛЕНА, ИЛИ ПРЕДНАЗНАЧЕНА ДЛЯ ИСПОЛЬЗОВАНИЯ ИЛИ ПЕРЕПРОДАЖИ КАК ИНТЕРАКТИВНОЕ ОБОРУДОВАНИЕ УПРАВЛЕНИЯ В ОПАСНЫХ СРЕДАХ, ТРЕБУЮЩИХ ОТКАЗОУСТОЙЧИВОЙ РАБОТЫ, ТАКИХ КАК СИСТЕМЫ УПРАВЛЕНИЯ ЯДЕРНЫМ ОБОРУДОВАНИЕМ, СИСТЕМЫ НАВИГАЦИИ САМОЛЕТА ИЛИ СИСТЕМЫ СВЯЗИ, СИСТЕМЫ УПРАВЛЕНИЯ ВОЗДУШНЫМ ДВИЖЕНИЕМ, МАШИНЫ ПОДДЕРЖАНИЯ ЖИЗНЕОБЕСПЕЧЕНИЯ ИЛИ ОРУЖЕЙНЫЕ СИСТЕМЫ, В КОТОРЫХ СБОЙ В ТЕХНОЛОГИИ JAVA МОЖЕТ ПРИВЕСТИ НЕПОСРЕДСТВЕННО К СМЕРТИ, ТЕЛЕСНОМУ ПОВРЕЖДЕНИЮ, ИЛИ СЕРЬЕЗНОМУ ФИЗИЧЕСКОМУ ИЛИ ЭКОЛОГИЧЕСКОМУ УЩЕРБУ.

Sun Microsystems, Inc письменно обязал Microsoft делать эту оговорку.

Ада: использовать для систем, связанных с безопасностью

- **Требования безопасности рекомендуют использование Ады для самых высоких уровней целостности**

- **Даже документ MISRA-C рекомендует использование Ады:**

  Рекомендации по использованию языка С для создания ПО для транспортных средств:

  •*“...очевидно, что есть и другие языки, которые в многом лучше подходят для создания систем, связанных с безопасностью, обладающие (к примеру) большей надежностью и лучшим контролем соответствия типов . Примером подобных языков, в целом значительно превосходящих С, есть Ада и Модула 2. Если эти языки доступны для предлагаемых систем, то их применение, в сравнении с С, считается более предпочтительным.” стр.3.*

Вдохновленные Адой свойства других языков программирования

- **C++**
  - Шаблоны (настраиваемые модули)
  - Исключения
- **Java**
  - Проверка индекса массива
  - Проверка деления на 0

Некоторые языки, производные из Ады

- **SPARK**
  - Подмножество Ады используемое для проектирования особо критичных по отношению к безопасности систем
- **VHDL**
  - Используется для проектирования чипов
- **PL SQL**
  - Язык программирования, предназначенный для расширения SQL и превращения его в полноценный императивный язык программирования

Некоторые индустриальные приложения написанные на Аде

- **Критичные по отношению к бизнес-процессам**
  - Canal+ Technologies: Плата-за-просмотр. Управление доступом
  - BNP: Язык принятия решений в торговле
  - Philips: Линия по производству полупроводников
  - Хельсинский радиотелескоп

- **Критичные по отношению к решаемой задаче**
  - Astree: Трансевропейская система передачи сигналов на железной дороге
  - Weirton Steel – управление сталелитейным производством
  - Электронные деньги Mondex
  - Сканирующий электронный микроскоп

- **Критичные по отношению к безопасности**
  - Аэробус Airbus A340
  - Аэробус Boeing 777
  - Российский самолет-амфибия Бе-200

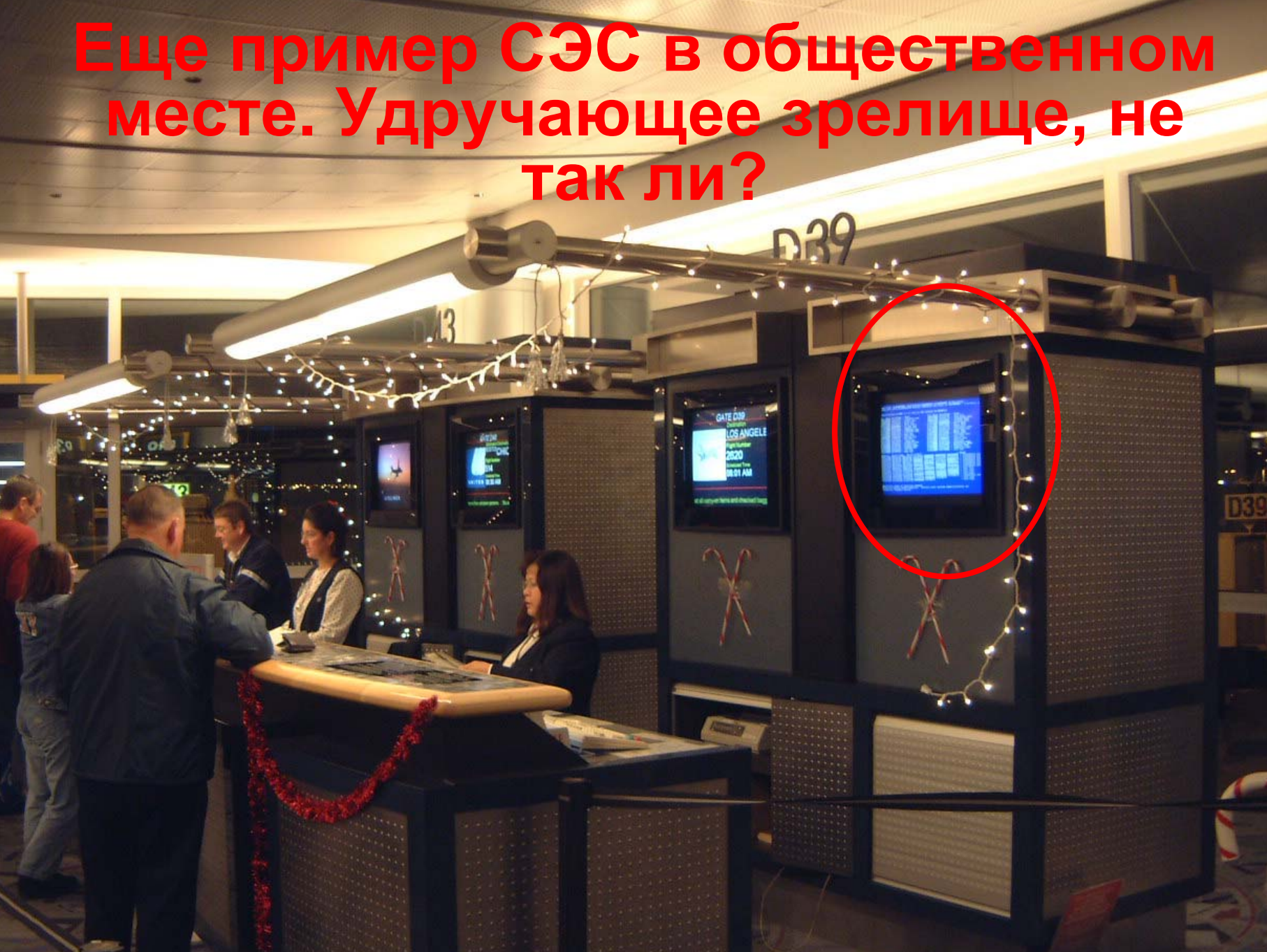
Надежность программного обеспечения

Надежность программного обеспечения

Степень уверенности пользователя в том, что ПО будет работать ожидаемым образом, и без сбоев при использовании его в нормальных режимах работы

```
* STOP: 0x0000001E (0x80000000,0xC06E8FF0,0xFFFFFFD4,0xC0000000)
D_POOL_HEADER

UID:GenuineIntel 5.2.c irql:1f  SYSVER 0xf0000565

l Base DateStmp - Name                  Dll Base DateStmp - Name
100000 3202c07e - ntoskrnl.exe          80010000 31ee6c52 - hal.dll
001000 31ed06b4 - atapi.sys             80006000 31ec6c74 - SCSIPORT.SYS
2c6000 31ed06bf - aic78xx.sys           802cd000 31ed237c - Disk.sys
2d1000 31ec6c7a - CLASS2.SYS            8037c000 31eed0a7 - Ntfs.sys
598000 31ec6c7d - Floppy.SYS            fc6a8000 31ec6ca1 - Cdrom.SYS
90a000 31ec6df7 - Fs_Rec.SYS            fc9c9000 31ec6c99 - Null.SYS
364000 31ed868b - KSecDD.SYS            fc9ca000 31ec6c78 - Beep.SYS
5d8000 31ec6c90 - i8042prt.sys          fc86c000 31ec6c97 - mouclass.sys
374000 31ec6c94 - kbdclass.sys          fc6f0000 31f50722 - VIDEOPORT.SYS
ffa000 31ec6c62 - mga_mil.sys           fc890000 31ec6c6d - vga.sys
708000 31ec6ccb - Msfs.SYS              fc4b0000 31ec6cc7 - Npfs.SYS
fbc000 31eed262 - NDIS.SYS              a0000000 31f954f7 - win32k.sys
fa4000 31f91a51 - mga.dll               fec31000 31eedd07 - Fastfat.SYS
b8c000 31ec6e6c - TDI.SYS               feaf0000 31ed0754 - nbf.sys
acf000 31f130a7 - tcpip.sys             feab3000 31f50a65 - netbt.sys
550000 31601a30 - el59x.sys             fc560000 31f8f864 - afd.sys
718000 31ec6e7a - netbios.sys           fc858000 31ec6c9b - Parport.sys
370000 31ec6c9b - Parallel.SYS          fc954000 31ec6c9d - ParVdm.SYS
5b0000 31ec6cb1 - Serial.SYS            fea4c000 31f5003b - rdr.sys
a3b000 31f7a1ba - mup.sys               fe9da000 32031abe - srv.sys

dress  dword dump   Build [1381]                            - Name
c32d84 80143e00 80143e00 80144000 ffdff000 00070b02          - KSecDD.SYS
1471c8 80144000 80144000 ffdff000 c03000b0 00000001          - ntoskrnl.exe
1471dc 80122000 f0003fe0 f030eee0 e133c4b4 e133cd40          - ntoskrnl.exe
147304 803023f0 0000023c 00000034 00000000 00000000          - ntoskrnl.exe

start and set the recovery options in the system control panel
 the /CRASHDEBUG system start option.
```

Еще пример СЭС в общественном месте. Удручающее зрелище, не так ли?



Значима ли надёжность программного обеспечения?

- **Несомненно значима! На маркетинговом уровне ☺**
  - Ни один поставщик не скажет, что его программное обеспечение ненадёжно
  - Ни одна команда разработчиков не сообщит, что разрабатывает ненадёжное ПО
- **В действительности, есть огромное количество ПО, ошибки в котором нас никак не задевают**
- **Не все программы нуждаются в том, что бы требование надёжност ставилось на первый план**
- **Сбой полезных, но некритичных программ все еще приемлемо 😐**
  - Если произойдет сбой во время этой презентации – достаточно просто перегрузить компьютер
  - Если Ваш текстовый редактор зависнет во время набора важного документа, это не принесет Вам ощутимого вреда, если Вы часто сохранял результаты своей работы

Надёжность программного обеспечения

- **Надёжность ≠ Пригодность**
  - Пример: текстовый редактор

Предостережение относительно подсчета количества отказов

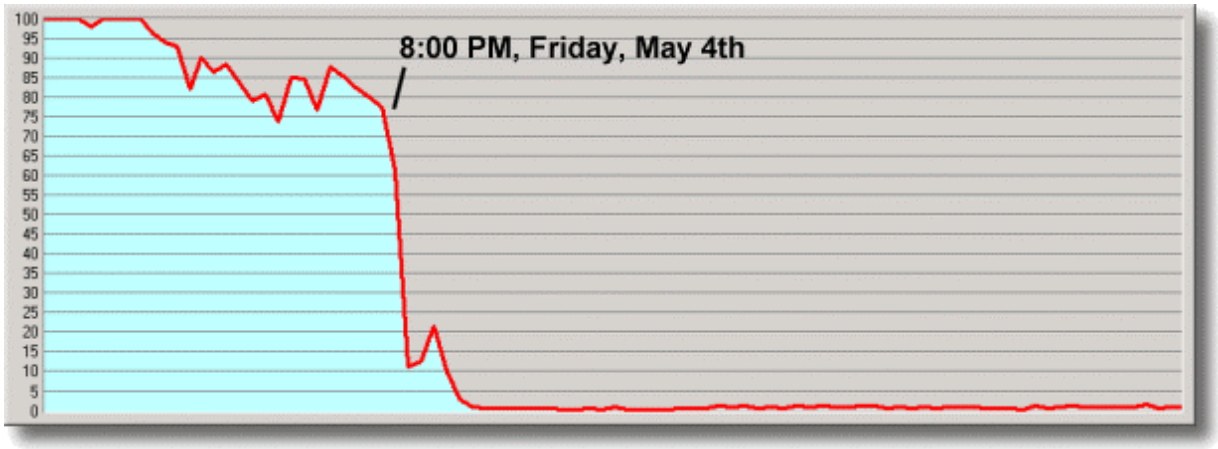
Приемлемо ли ПО с 99.9% защитой от отказа? Все зависит от ситуации... Ведь может случиться, что оставшийся 0.1% это:

- **1 документ в год потерянный в момент редактирования**
  - Хорошо ☺
- **2 происшествия в месяц в Международном Аэропорту в Лондоне**
  - ☹☹
- **22 000 чеков в час выписанные с ошибочных счетов по всей Америке**
  - ☹☹

Следствие: количество отказов ПО необходимо анализировать в контексте приложения

Сбои ПО: Доступность

- **Атаки отказа от обслуживания**
  - Пример: атака GRC.com
    - Атаковано 195 серверов Windows 2000 исполнявших недостаточно защищенную версию web-сервера Microsoft IIS. Для проникновения в систему хакерами была использована брешь в защите IIS. Это привело к остановке серверов и временной их недоступности



Сбои ПО: Безошибочность

- **Январь 15, 1990: на 9 часов остановлена общенациональная телефонная сеть США**
  - месяц ранее AT&T обновила ПО на 114 коммутируемых телефонных станциях
  - Причина: 1 неуместный оператор “break” в программе на языке C
- **Январь 2001: отзывается 230,000 единиц новых мобильных телефонов с доступом в Интернет**
  - Пользователи сообщают, что их телефоны зависают после посещения некоторых web-узлов, а после перезапуска телефона все сохраненная на нем информация (адреса, ссылки, записи) теряется
- **Matracom 6500 PABX (телефонный коммутатор)**
  - Искажение случайных телефонных разговоров
  - Внезапное прерывание длинных телефонных звонков
- **Windows NT**
  - Сентябрь 1997: повреждение силовой установки судна USS Yorktown
  - Причина: крах Windows NT 4.0

Сбои ПО: Безопасность

- **1986: Медицинская облучающая установка Therac 25 убила несколько пациентов**
  - Причина: недостаточно тестированное ПО установки
- **Июнь 4, 1996: 1-й полет ракеты Ariane 5 завершился неудачей: сработал механизм самоуничтожения**
  - Причина: проверенный временем код системы управления ракетой Ariane 4 был перенесен на Ariane 5, но не был протестирован.
- **2000: Большая автомобильная катастрофа на скоростном шоссе во Франции**
  - Причина: Неисправность ПО тормозной системы автомобиля. Производитель автомобиля признал свою ответственность за случившееся.

Сбои ПО: Защищенность

- **Ноябрь 2, 1988 Интернет-червь**
  - Самораспространяемая программа начала свое шествие через Интернет
  - Эта программа (червь) заражала компьютеры VAX и Sun работающие под Berkeley UNIX, и использовала их для атаки на другие компьютеры
  - За нескольких часов она распространилась на все Соединенные Штаты, инфицируя тысячи компьютеров и делая многие из них неработоспособными, чрезмерно нагружая их своим кодом
  - Причина: необнаружаемое переполнение буфера в функции gets() библиотеки времени выполнения языка C
- **Множество занимательных историй о вирусах, в особенности для ОС Windows**

… и 30% проектов ПО, которые не дожили даже до этих стадий

- **Модернизация налогового управления США**
  - $4 миллиарда, прекращена в начале 1997
- **Система анализа отпечатков пальцев для ФБР**
  - $500 миллионов, прекращена
- **Bell Atlantic 411**
  - Ноябрь 1996, устарела, принято решение систему не модернизировать

Введение в конструирование программного обеспечения

Франко Гасперони
gasperon@act-europe.fr
http://libre.act-europe.fr/Software_Matters
перевод: Владислав Козловский
dbdeveloper@rambler.ru

Домашняя страница курса

- **http://www.ada-ru.org/slides**
  - Здесь находятся все слайды курса (PDF и PowerPoint)

Местонахождение оригинального курса

- **http://libre.act-europe.fr/Software_Matters**
  - Здесь находятся все слайды оригинального курса (PDF и PowerPoint)

Уведомление об авторском праве

- **© ACT Europe согласно GNU Free Documentation License**
- **© Владислав Козловский (перевод) согласно GNU Free Documentation License**
- Позволяется копировать, распространять и/или модифицировать этот документ согласно условиям GNU Free Documentation License, Версии 1.1 или более поздней, опубликованной Free Software Foundation, при условии упоминания автора оригинала и переводчика, а также сохранении ссылки на первоисточник (http://libre.act-europe.fr/). Полный текст лицензии доступен по адресу:
- **http://www.fsf.org/licenses/fdl.html**

Интересные ссылки

- **http://www.ada-ru.org**
  - Ада по-русски. Сайт русскоязычного сообщества языка Ада.
- **http://www.fsf.org**
  - Сайт Фонда Свободного ПО (the Free Software Foundation) и проекта GNU
- **http://libre.act-europe.fr**
  - Interesting Free Software projects written in Ada 95
- **http://adapower.com**
  - Очень интересный сайт посвященный языку Ада, с огромным колличеством информации и учебных пособий
- **http://www.adaic.com/whyada/ada-vs-c/cada_art.html**
  - Сравнение цены разработки с использованием языков C и Ada

Спасибо за внимание

Спонсор научно-практического семинара "Ada-технологии в современной программной индустрии" в рамках выставки PTS-2009