**Memory перевод.**

Интересная заметка, что память один из главных компонентов компьютера, часто называемый хранилище. Она хранит вычислительные программы, вычислительные формулы, начальные значения, промежуточные и конечные результаты. Следовательно, функции компьютерной памяти можно классифицировать следующим образом. Во-первых, память компьютера должна хранить информацию, переданную от устройств ввода и других девайсов. Во-вторых, память должна производить информацию необходимую для вычислительных процессов всех других устройств компьютера. Как правило, память состоит из двух основных частей называемых основная, первичная или внутренняя память и вторичная, или внешняя память. Преимущество первичной памяти в экстремально высокой скорости. Вторичная память имеет преимущественно низкую скорость, но позволяет хранить намного большее количество информации чем основная память. Первичное хранилище принимает непосредственное участие в вычислительном процессе. Вторичное хранилище дает информацию необходимую для одного шага в последовательности вычислительных шагов.

Самые важные характеристики производительности устройства хранения — это скорость, емкость и надежность. Его скорость измеряется единицами времени. Его емкость измеряется числом машинных слов или бинарных цифр. Надежность измеряется числом отказов в единицу времени. Существует два типа устройств хранения: устройства только для чтения, устройства для чтения и записи (RWM устройства). Первые содержат данные которые нельзя изменить, данные со вторых могут быть стерты и перезаписаны снова.

Компьютерная память реализована как кремний и изготовлена так же, как и аппаратные процессоры и другие схемы. Единственное различие между компьютерной памятью и, скажем, схемой, взаимодействующей с интернетом или процессором в том, что первая имеет постоянную структуру.

Доступны различные устройства. Масочные ПЗУ сохраняют свои данные или программы на этапе производства и не могут быть изменены. Программируемые “Только для чтения” устройства, обычно называемые PROMs изготавливаются пустыми и могут быть запрограммированными при помощи электронного устройства. Однако, поскольку они доступны только для чтения, это может произойти только один раз - они не могут быть перепрограммированы.

Стираемая программируемая постоянная память, более известная как EPROM, идет на один шаг дальше, чем PROM, поскольку ее можно стереть, а затем перепрограммировать. Существует путаница в отношении EPROMs: поскольку они могут быть перепрограммированы, создается впечатление, что их содержимое может быть изменено путем перезаписи, когда они находятся в компьютере. Перезапись может быть достигнута только с помощью специального оборудования, в которое вставлено устройство EPROM.

Важным шагом вперед, который преодолел мир постоянной памяти и доступной для записи памяти, была разработка электрически стираемой программируемой постоянной памяти, часто называемой EEPROM. Это форма EPROM, которая может быть удалена, когда подключена к компьютеру. Технология, известная как флеш-память, была разработана с использованием идей и технологий, связанных с EEPROM.Flash технология используется в приложениях, где требуется большое количество нестираемой памяти. Например, это на USB-накопителях, которые используются для передачи данных с компьютера на другой или в качестве резервного хранилища в случае сбоя компьютера.

Другие приложения, использующие технологию флэш-памяти, включают в себя цифровые ассистенты, портативные компьютеры, цифровые аудиоплееры, камеры и мобильные телефоны. Маленький портативный компьютер нового поколения, известный как нетбук, использует флеш-память для хранения программ и файлов данных. Используемая память реализуется с использованием ряда технологий. Динамическая память произвольного доступа (DRAM) является наиболее распространенной памятью, используемой на компьютерах. Он реализован таким образом, что он может потерять свои данные за короткий промежуток времени. Из-за этого содержимое ячеек DRAM постоянно считывается и записывается для пополнения своих данных. Статическая память произвольного доступа (SRAM) - это технология, которая не требует обновления, которая должна быть применена к ее ячейкам, если программа не изменяет ячейку или питание удаляется с компьютера, в котором он содержится. Он быстрее, но дороже, чем DRAM.

**Аннотация Memory**

Согласно названию, в статье описывается такой компонент компьютера, как память (хранилище). Подробно рассказывается о функциях, выполняемых памятью. Описываются различные виды хранилищ, первичные (внутренние) и вторичные (внешние), их плюсы и минусы, сходства и различия, случаи использования, стоимость и т.д. Раскрываются основные характеристики, на которые следует обратить внимание при выборе хранилищ. Даются ответы на такие вопросы, как: “Что есть внутренняя память?”, “Что такое внешнее хранилище?” и т.д. Описываются различные устройства хранилищ, такие как: Масочные ПЗУ, только для чтения (PROMs), Стираемая программируемая постоянная память (более известная как EPROM), электрически стираемая программируемая постоянной памяти (EEPROM). Технологии использования памяти - Динамическая память произвольного доступа (DRAM), Статическая память произвольного доступа (SRAM).

В целом, текст достаточно хорошо описывает предмет и формирует общее представление о видах, устройствах и функциях хранилищ компьютера.

As the title implies the article describes such a computer component as memory (storage). Tells in details about the functions performed by the memory. It is spoken in detail different types of storages, primary (internal) and secondary (external) memory, their pros and cons, similarities and differences, use cases, cost, etc. It reveals the main characteristics that you should pay attention to when choosing storage. Answers to questions such as: “What is internal memory?”, “What is external storage?” and so on. The text describes the various storage devices, such as mask-programmed ROMs, programmable read only memories (PROMs), Erasable Programmable read-only memory (also known as EPROM), electrically erasable programmable read-only memory (EEPROM). It discusses memory usage technologies such as Dynamic Random Access Memory (DRAM) and Static Random Access Memory (SRAM).

In general, the author quite well explains the subject and forms a general idea of the types, devices and functions of computer storage.

**Inputs-Outputs перевод**

Как известно, компьютер не может выполнить или завершить полезную работу без возможности взаимодействия с внешней средой. Все данные и инструкции вводят и выходят из центрального процессора через основное хранилище. Устройства ввода-вывода необходимы для связывания первичного хранилища с окружающей средой, которая является внешней по отношению к компьютерной системе. Поэтому устройства ввода используются для ввода данных в основное хранилище. Устройства вывода принимают данные из первичного хранилища, чтобы предоставить пользователям информацию или записать данные на вторичное запоминающее устройство. Некоторые устройства совмещают в себе функции ввода и вывода вместе.

Данные, с которыми работают эти устройства, могут быть или не быть в форме, которую понимают люди. Например, данные, которые оператор ввода данных вводит в память компьютера, набрав на клавиатуре, могут быть прочитаны людьми. Однако данные, которые сообщают компьютеру о производительности автомодуля, не в той форме, которую могут прочесть люди. Это электрические сигналы от аналогового датчика. Точно так же вывод может быть напечатан на странице, которую люди могут легко читать или на каком-либо другом носителе, где данные не видны, например, на магнитной ленте или диске.

Как известно, весь поток данных от ввода до конечного вывода управляется блоком управления в ЦП. Независимо от характера устройств ввода / вывода, для преобразования входных данных во внутренние коды, используемые компьютером, и для преобразования внутренних кодов в формат, который может использоваться устройством вывода, требуются специальные процессоры, называемые интерфейсами ввода / вывода.

**Аннотация I/O**

Эта статья касается устройств ввода-вывода. Подробно описывается, как происходит процесс взаимодействия компьютера с внешней средой. Описывается, какую роль в этом процессе играют устройства ввода и вывода, для чего они предназначены и в чем их польза. В тексте рассказывается для чего устройства вывода принимают данные из первичного хранилища. Какова форма данных, передаваемых этими устройствами, и является ли этот формат понятным для людей? Также здесь вы можете найти ответы на вопросы: какова форма данных, передаваемых этими устройствами, и является ли этот формат понятным для людей? Какие интерфейсы ввода и вывода? Для чего они нужны?

The article deals with input/output devices. It is spoken in detail the process of interaction between the computer and the external environment. The author explains the role of input and output devices, what they are for, and why they are useful. The text tells why output devices receive data from the primary storage. Also here you can find the answers to the questions: What the form are the data transmitted by these devices and is this format understandable to humans? What are input and output interfaces? What are they needed for?

The article is of interest to all those who want to get a brief and clear description of these devices.

**My First Use of Computer**

Мне было 8, когда мы с классом отправились на экскурсию в компьютерный класс старшей школы. Когда мы прошли в класс, то увидели множество компьютеров, довольно немощных по сегодняшним меркам, но вполне сносных для того времени. Мы сели за столы и нам рассказали о компьютерах в целом и провели инструктаж перед работой. Учитель рассказала, для чего используется компьютер, что умеет и какие функции он выполняет. Также рассказали из каких компонентов состоит персональный компьютер и какие функции выполняет каждый из них.

Затем мы сели за компьютеры чтобы попробовать что ни будь сделать на практике. В тот момент было волнительно и страшно, но в то же время интересно, потому что мы используем настоящий компьютер!

На компьютерах была установлена Windows NT, а вместе с ней и пару стандартных программ. Но специально для нас учитель установила несколько игр, в которые мы с радостью могли играть. Это было так весело и интересно, что мы совсем не заметили, как наше время вышло и нам пора было идти…

С тех пор прошло много лет, компьютер появился в каждом доме и этот случай может показаться забавным, но эмоции в тот раз нельзя было описать словами.

После школы я решил, что пойду учиться на программиста. И теперь работа за компьютером стало для меня профессиональной обязанностью, но я навсегда запомню мой первый опыт использования компьютера.

I was 7 when my class took a field trip to the high school computer class. When we went to class, we saw a lot of computers, quite weak by today's standards, but quite powerful for that time. We sat at the tables and told us about computers in general and gave instructions before work. The teacher told us what the computer is used for, what it can do, and what functions it performs. They also told what components a personal computer consists of and what functions each of them performs.

Then we sat down at the computers to try do something in practice. In that moment it was exciting and scary, but at the same time interesting, because we use a real computer!

On computers was installed Windows NT, and with it a couple of standard programs. But special for us, the teacher installed several games that we could happily play. It was so fun and interesting that we didn’t notice at all how our time is over and it was time for us to go ...

Many years have passed since then, a computer appeared in every home, and this case may seem funny, but emotions could not be described at that time with words.

After high school, I decided that I would go to university to be a programmer. And now work at the computer has become my profession, but I will remember forever my first experience of using a computer.

**Перевод терминов с. 16**

Software – программное обеспечение, Hardware – аппаратное обеспечение, Network – сеть, Processing – обработка, Database – база данных, Algorithm - алгоритм, Multisensory – мультисенсорный, Blogging – блоггинг (вести блог), Script –скрипт, Reception – прием, Logging – логгирование, Load – загрузка, Flash - вспышка, Dimension – размерность, Array – массив, Interface – интерфейс, Router – роутер (маршрутизатор), Gateway – шлюз, Aggregation – агрегирование (слияние многих в один), Spam – спам, Scroll - скролл, punched card - перфокарта, loop - цикл, icon – иконка (небольшое изображение), flowchart – блок-схема, simulation - симуляция, paste - вставить, draft – набросок (тут скорее всего имеется ввиду прототип), blink – мерцание

**Перевод терминов с. 23**

1. Wireless sensor network architecture – архитектура сети беспроводных датчиков

2. IBM Web sphere portal primer - ?

3. PCI express system architecture – архитектура PCI express системы.

4. Designing storage area networks - Проектирование сетей хранения данных.

5. Aided design engineering and manufacturing systems - Автоматизированные системы проектирования и производства.

6. Geographic information system implementation - реализация географической информационной системы.

7. Web sphere certification study guide – руководство по сертификации в вэб сфере.

8. C++ pseudo code algorithms foundations - C ++ псевдокод основных алгоритмов - ???

9. File compression utility – Утилиты для сжатия файлов.

10. Word wrap outgoing text – Перенос слов исходящего текста.

**Перевод терминов с. 24**

|  |  |
| --- | --- |
| 1. The class of regulators can be thought of as composed of three parts: a parameter estimator, a linear controller and a block, which determines the controller parameters. | Класс регуляторов можно рассматривать как состоящий из трех частей: оценщика параметров, линейного контроллера и блока, который определяет параметры контроллера. |
| 2. A numeral analysis of two point boundary value problem algorithms was presented in this article. | В этой статье был представлен численный анализ двухточечных алгоритмов граничных задач. |
| 3. Full technical backup and advisory services are guaranteed throughout the world by Messer Oriesheim’s own sales network. | Полное техническое обеспечение и консультационные услуги гарантированы во всем мире собственной сетью продаж Messer Oriesheim. |
| 4. File associations in Windows PCs are based not on some esoteric code within a file but on something almost absurdly simple — the filename extension characters at the end of the name of a file. | Сопоставление файлов в компьютерах на Windows основано не на каком то эзотерическом коде, но на чем то практическа абсурдно простом – на расширении файлов, символов в конце имен файлов. |
| 5. The researchers found that criminals have developed highly sophisticated mechanisms for distributing stolen credit card information “through specialized IRC channels and related Web sites” | Исследователи обнаружили, что преступники разработали очень сложные механизмы для распространения украденной информации о кредитных картах «через специализированные каналы IRC и связанные с ними веб-сайты» |

**Перевод терминов с. 26**

|  |  |
| --- | --- |
| 1. This hardware features extraordinary stability. | Это аппаратные возможности необычайно стабильны. |
| 2. These features are assigned to the stability of systems involved. | Эти фичи придают стабильности вовлеченным системам |
| 3. A heat filter, placed ahead of the first lens, reduces system heat during the summer. | Тепловой фильтр, расположенный перед первой линзой (окном?), снижает тепловую температуру в течение лета. |
| 4. A turn increases the divergence angle of a beam. | Поворот увеличивает угол дивергенции пучка (лучей?). |
| 5. A switch should be turned to the sight before testing operation. | Переключатель должен быть повернут в сторону перед тестирующей операцией |
| 6. The increase of operating speed is achieved by using special kind of gadget. | Увеличение скорости работы достигается за счет использования хорошего гаджета. |
| 7. The speed is increased by using a very cheap technology | Скорость увеличивается благодаря использованию очень дешевой технологии |

**Перевод терминов с. 28**

|  |  |
| --- | --- |
| 1. То the sophisticated eye today, software component of the late 80s appears primitive. | Сегодняшнему утонченному глазу, компоненты ПО поздних 80 кажутся примитивными |
| 2. Hardly had economic depression dripped the world when the stock market crashed machine building industry | Едва экономическая депрессия вылила мир, когда фондовый рынок обвалила индустрия машиностроения - ? |
| 3. Because of the continued need for improved equipment, construction equipment design did not stagnate during the Great Depression | Из-за продолжающейся потребности в улучшенном оборудовании, разработка конструкции оборудования не застаивалась во время Великой депрессии |
| 4. A gradual improvement in productivity brought these innovations | Постепенное повышение продуктивности привело к этим инновациям |
| 5. It was a new kind of technology that permitted higher operating speeds. | Это был новый вид технологии, который позволил ускорить рабочий процесс. |
| 6. Closely relating to this problem is the problem of encoding. | С этой проблемой тесно связана проблема кодирования. |
| 7. Faraday was no mathematician, nor was Hamilton much of a physicist. | Фарадей не был математиком, но и Гамильтон не был физиком |

**Перевод статьи “Using XML and Frameworks to Develop Information Systems” c. 84**

**Использование XML и фреймворков в разработке информационных систем**

**Аннотация:** Для оптимизации времени разработки и затрат на разработку программного обеспечения эта разработка должна проводиться в среде, которая помогает разработчику справляться с большим количеством понятий, полученных на этапе анализа доменной области и семантическим разрывом между этими концепциями и объектно-ориентированной моделью проектирования из-за их разных уровней абстракции. В данной статье описываются основные особенности среды, предназначенной для поддержки разработки IS программного обеспечения, базируемого на фреймворке позволяющем повторное переиспользование и XML спецификациях.

Ограничения в стоимости и времени на продажу, налагаемые на современную разработку программного обеспечения обязывает разработчиков приложений отказаться от подхода разработки с нуля и использовать, и принять подход переиспользования для развития разработки ПО. Как следствие во время разработки системы проверенные решения, такие как Компоненты [6] и фреймворки [7], должны быть совмещены с использованием начальной спецификации приложения для получения окончательной архитектуры / кода. Также важно, чтобы процесс разработки первоначальной спецификации приложения охватывал знания доменной области и направлял разработчика ПО к сопоставлению/трассировки его перевода к какому-либо представлению дизайна, такому как Обьектно-ориентированный дизайн, откуда может быть извлечена окончательная спецификация.

В этой статье мы рассказываем о текущей среде разработки, использующей Domain и Reuse Driven подходы к ПО и проблемам разработки. Эта работа, является развитием подхода, представленного с внедрением стандартов XML / XMI для представления проектов. Другое изменение подхода. Другим изменением подхода является использование фреймворка языка разработки, позволяющего иметь дело с характеристиками общих спецификаций.

Также важно отметить, что такой подход должен быть

основанный на некоторых характеристиках:

* *Совместимость.* Он должен использовать рыночные стандарты для обеспечения совместимости / интеграции с другими системами.
* Читаемость кода - во время разработки компиляция/отладка обычно выполняется с помощью IDE, такого как Borland JBuilder и IBM VisualAge, поэтому пользователь должен понимать финальный код.
* Сфокусированность на OO. Пользователь должен знать только методы ООП.
* UML - благодаря стандартам OMG
* Расширяемость - Повторное использование таких действий, как наследование, шаблоны, рамки и аспекты помогут развиваться.

Учитывая эти особенности, наш подход принимает такие рыночные стандарты, как UML и XMI, которые используются в качестве основы для представления диаграмм. Этот подход начинается с тщательный анализ для определения общих и разных аспектов доменной области с помощью метода FODA и Use-Case-ов для уменьшения так называемого «семантического разрыва». После этого архитектор приложения создает диаграмму классов, основанную на предыдущих моделях, которые будут использоваться в качестве приложения ICEIS 2001 - Анализ и спецификация информационных систем

На этапе повторного использования у нас есть модификация традиционного подхода к разработке программного обеспечения, в котором следует использовать повторное использование. На этом этапе получено представление XMI спецификации домена, чтобы облегчить манипуляцию композицией этой спецификации с использованием спецификации повторного использования (также выраженной в XMI), которая хранится в репозитории повторного использования.

После выбора спецификации повторного использования, которая выполняется эмпирически, пользователь среды будет ориентирован на выполнение действий повторного использования, заявленных как классовые предложения переопределения, использование шаблонов или композиций, которые были определены дизайнером многоразового артефакта. Раздел 2 описывает обзор подхода и его адаптацию к текущим процессам. В разделе 3 мы описываем, как собирается информация о домене. В Разделе 4 изображен подход к повторному использованию. В разделе 5 сообщается о развитии арендной системы, используемом в качестве примера. В последнем разделе мы приводим наши выводы.