

MC 106
Organização e Arquitetura de Computadores

Unidade I - Introdução

Prof. Dr. Daniel Mesquita

Hoje

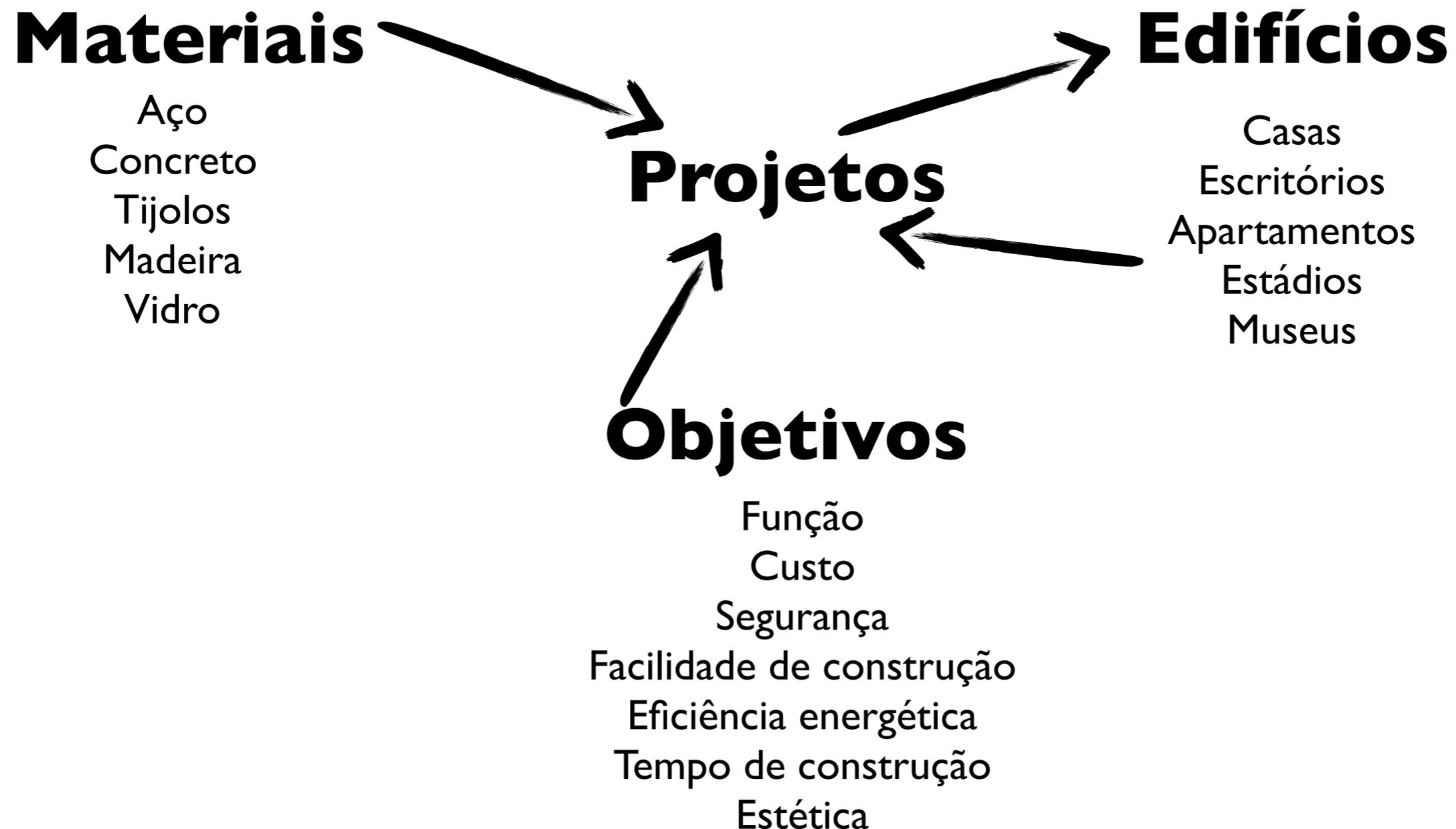
- O que é arquitetura de computadores
- Tecnologia
- Microprocessador, lei de Moore
- Paralelismo implícito
- Paralelismo explícito
- Por quê estudar arquitetura de computadores
- Planejamento do curso

O que é arquitetura de computadores

- “ É a ciência e a arte de selecionar e interconectar componentes de hardware para criar computadores que atendam objetivos funcionais, de desempenho e de custo” - (Martin/Roth, Univ. of Pennsylvania)
- Não muito diferente da arquitetura de edifícios...

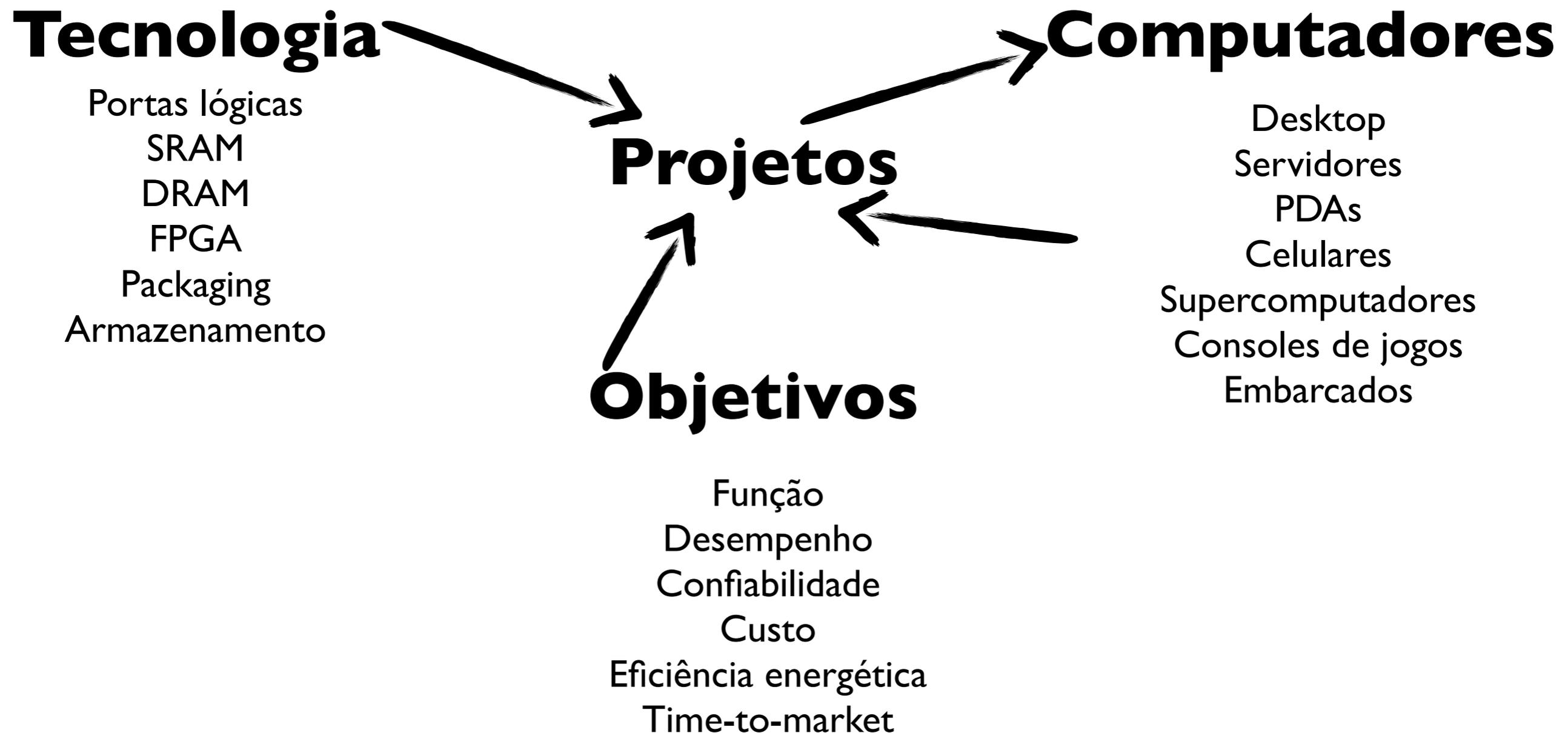
O que é arquitetura ~~de computadores~~

Papel do arquiteto de edifícios



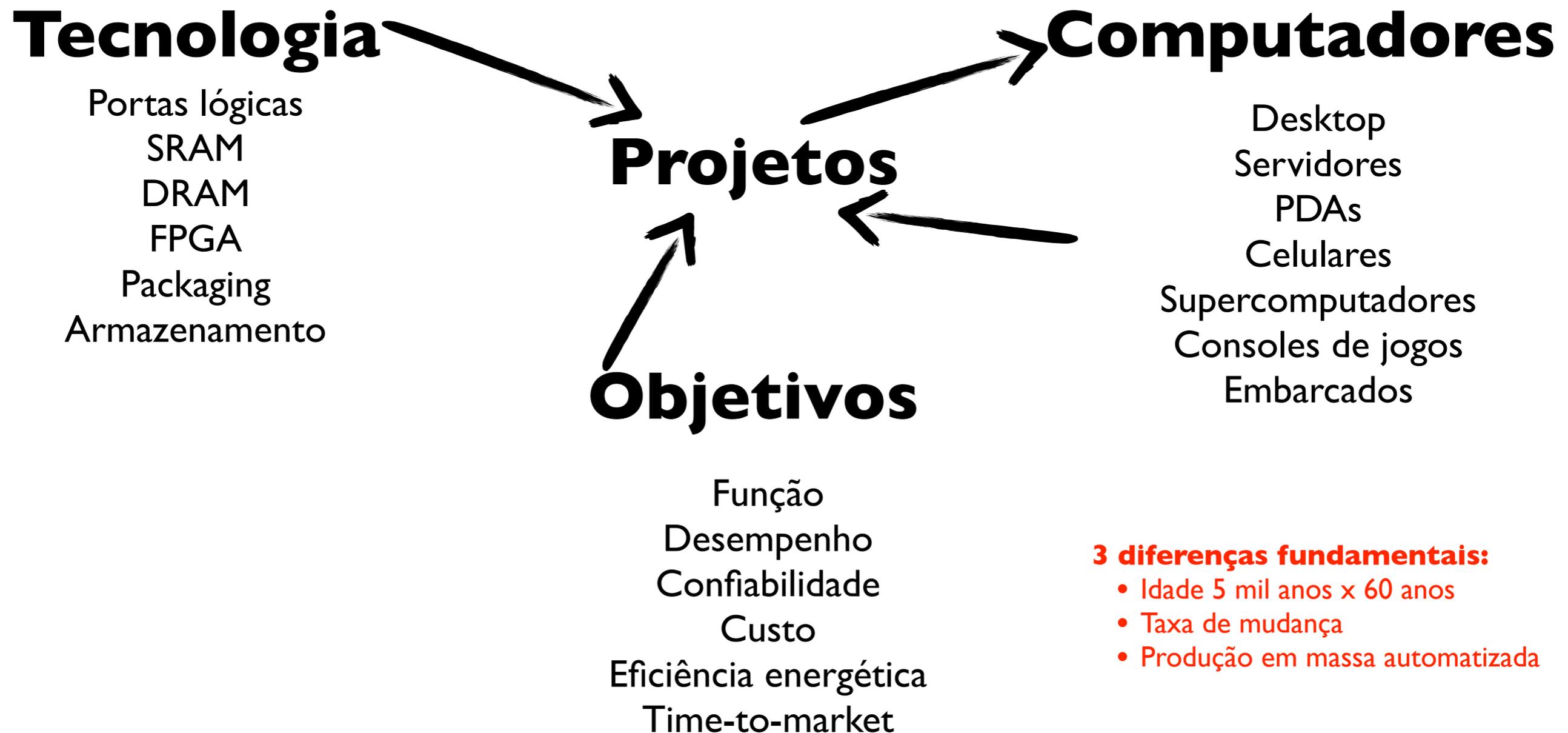
O que é arquitetura de computadores

Papel do arquiteto de computadores



O que é arquitetura de computadores

Papel do arquiteto de computadores



O que é arquitetura de computadores

Objetivos de projeto

- Funcionalidade
- Confiabilidade
- Alto desempenho
- Baixo custo
- Eficiência energética

O que é arquitetura de computadores

Objetivos de projeto

- Funcionalidade
- Confiabilidade
- Alto desempenho
- Baixo custo
- Eficiência energética

Desafio:

- Como equilibrar a importância desses objetivos?
- Foco da disciplina: *desempenho*

O que é arquitetura de computadores

Outro fator de definição: aplicações

- Aplicações: uso e contexto
 - Aplicações e domínios de aplicações tem diferentes requisitos
 - Domínio: grupo com características similares
 - Leva a diferentes projetos

O que é arquitetura de computadores

Outro fator de definição: aplicações

- Aplicações: uso e contexto
 - Aplicações e domínios de aplicações tem diferentes requisitos
 - Domínio: grupo com características similares
 - Leva a diferentes projetos

Constatação:

- Uma arquitetura sozinha não pode ser a solução para todos problemas

O que é arquitetura de computadores

Outro fator de definição: aplicações

- Aplicações científicas

- **Necessidades:** grande memória, trabalho intensivo para unidades de vírgula flutuante
- **Exemplos:** Cray T3E, IBM BlueGene

- Aplicações Comerciais

- **Necessidades:** movimento de dados, grande memória, E/S intensiva, grande largura de banda
- **Exemplos:** Sun Enterprise Server, AMD Opteron, Intel Xeon

O que é arquitetura de computadores

Outro fator de definição: aplicações

- Desktops

- **Necessidades:** grande memória, gráficos, rede, largura de banda

- **Exemplos:** Intel Core i7, AMD Athlon, PowerPC G5

- Computação móvel (notebooks, celulares,...)

- **Necessidades:** baixo consumo, desempenho com inteiros, wireless

- **Exemplos:** Intel Core 2 mobile, AMD Turion, Atom, ARM, MIPS

O que é arquitetura de computadores

Outro fator de definição: aplicações

- Embarcados (microcontroladores)
 - **Necessidades:** baixo consumo, baixo custo
 - **Exemplos:** ARM, DSPs, MIPS
 - **Curiosidade:** mais de 1 bi de ARMs vendidos em 2006
- Profundamente embarcado (sensores *smart dust*)
 - **Necessidades:** extremamo baixo custo, extremo baixo consumo
 - **Exemplos:** Por enquanto, apenas em Sci-Fi...

O que é arquitetura de computadores

General purpose x Application Specific

- *General Purpose*

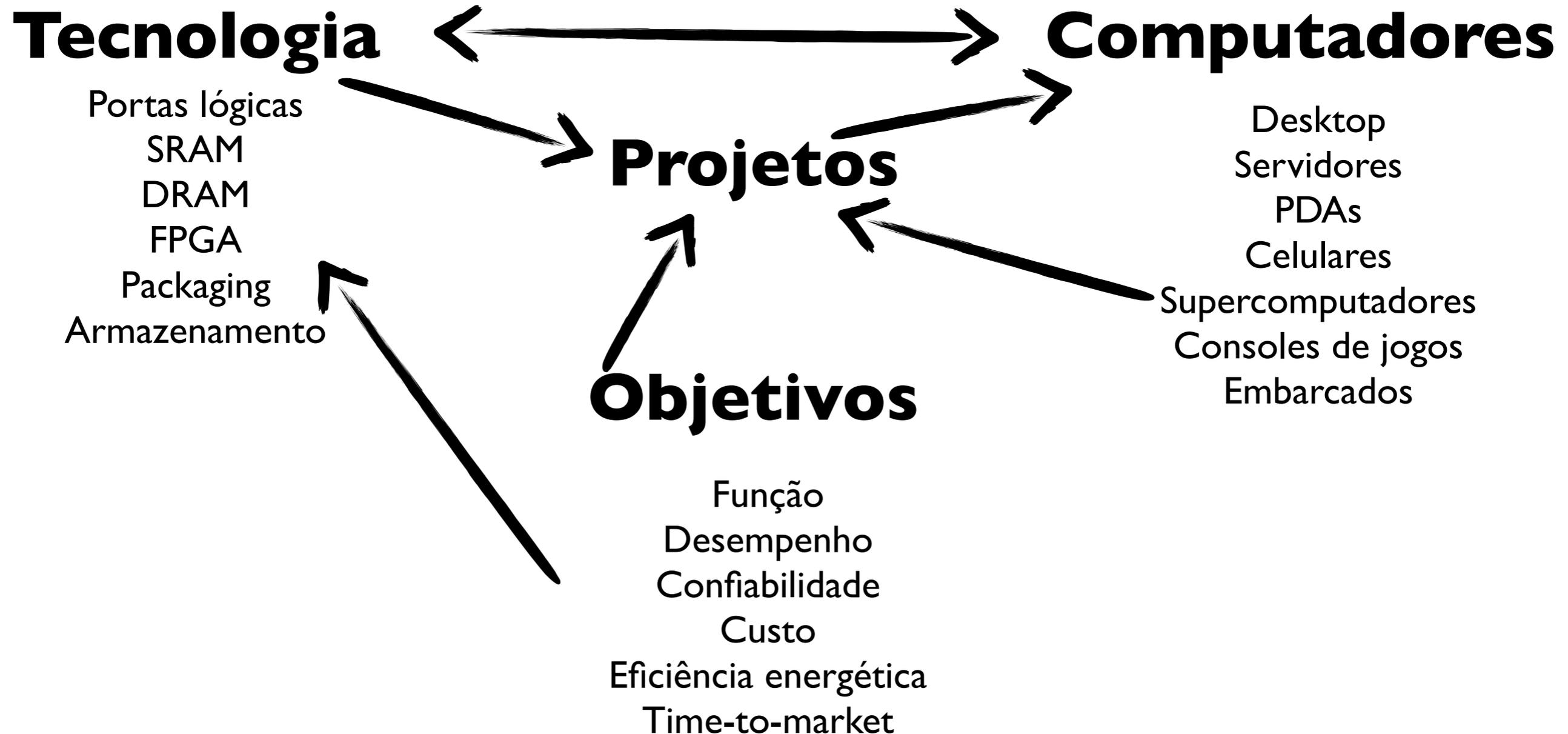
- Este curso versa sobre UCPs de propósito geral
- Intel Core 2, MIPS, ARM, etc.

- *Application Specific*

- ASICs, ASIPs
- Implementam funcionalidades de domínio específico em hardware
- Exemplos: codificação de vídeo, criptografia

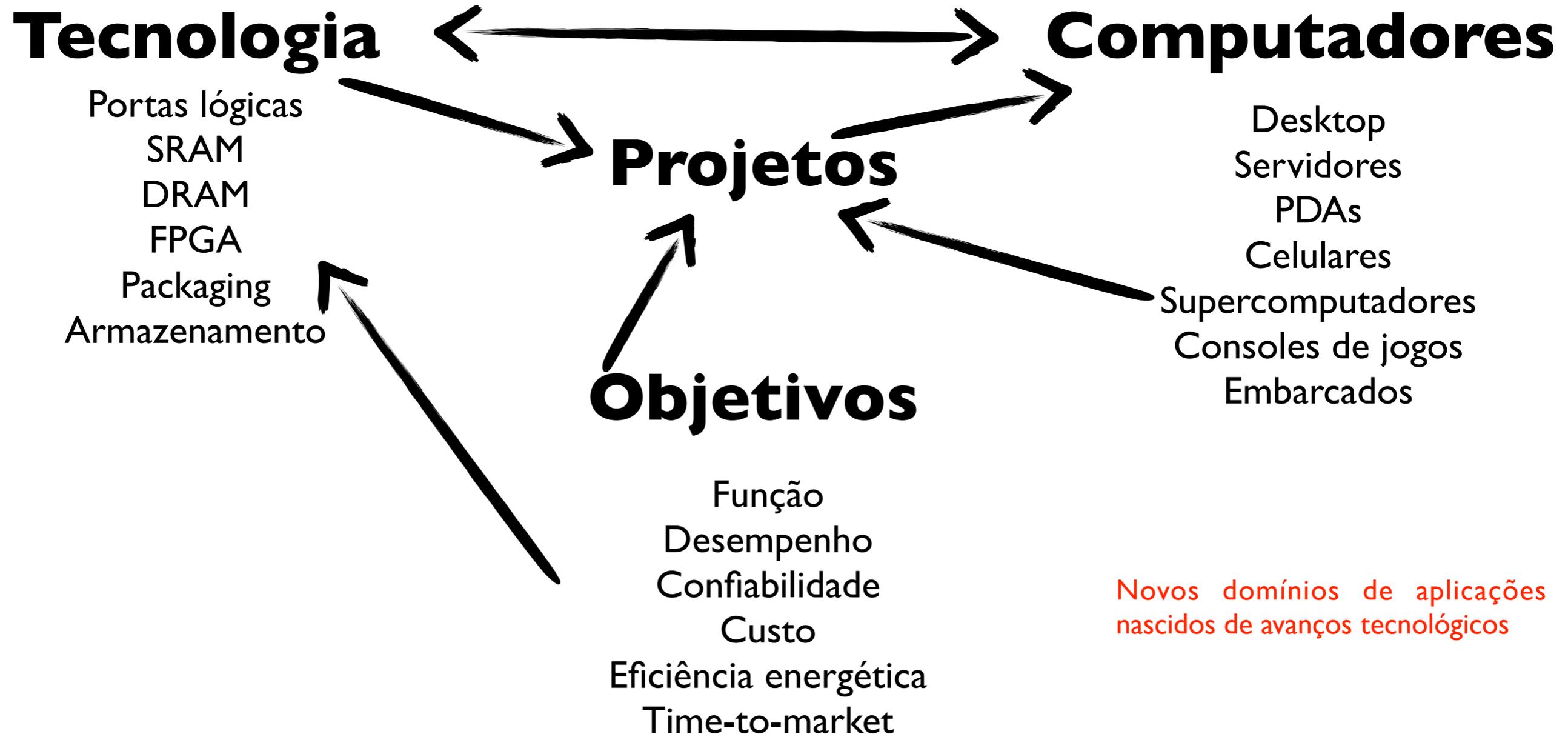
Tecnologia

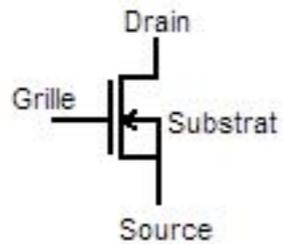
Mudanças constantes



Tecnologia

Mudanças constantes





Tecnologia



- Elemento Básico: transistor
- Qual o interesse? Todo!
- Famílias de ICs
 - **SRAM:** *otimizado para velocidade, usada em processadores*
 - **DRAM:** *otimizado para densidade, custo, energia, usado para memória*
 - **Flash:** *memória não-volátil*



Tecnologia

Tendências

- Lei de Moore: miniaturização do transístor
- Consequências
 - Incrementos extremos em densidade, velocidade, energia e custo
 - SRAM: Densidade ~30%, velocidade ~20% (anual)
 - DRAM: Densidade ~60%, velocidade ~4% (anual)
 - Disco (magnético): Densidade ~60%, velocidade ~10% (anual)
 - Necessidade de reavaliar/reprojetar para cada geração de tecnologia

Tecnologia

Mudanças na tecnologia direcionam tudo

- **Computadores 10x mais rápidos e baratos a cada 5 anos**
 - 10x é uma mudança quantitativa E qualitativa (avião x carro)
 - Necessidade de reavaliar/reprojetar para cada geração de tecnologia
- **Novas aplicações criam mercados**
 - “Invenção é a mãe da necessidade”
 - Smartphones, câmeras digitais, MP3 players

Microprocessador

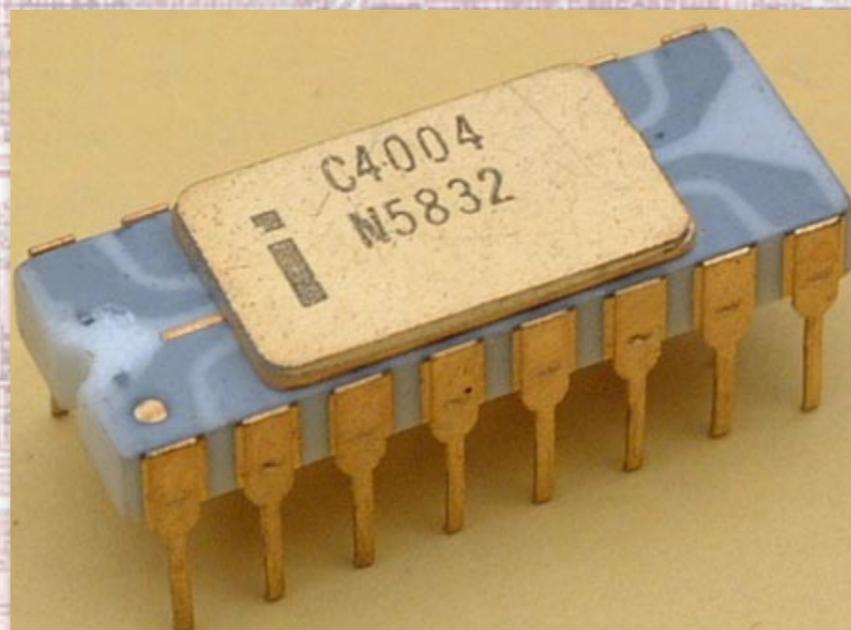
1ª revolução na computação

- Transistores suficientes para fazer um processador inteiro em um chip (anos 80)
- Desempenho enorme, ainda com menos chips conectados, menor custo
- Criou novos segmentos de mercado
 - Desktops, CD players, laptops, consoles de jogos, set-top boxes, etc.
- Superou e substituiu mercados existentes
 - Supercomputadores, mainframes, minicomputadores, etc.

Microprocessador

1º microprocessador

- Intel 4004 (1971)
 - Aplicação: calculadoras
 - Tecnologia: 10µ
 - 2300 transístors
 - 13mm²
 - 108KHz
 - 12 Volts
 - 4 bits



Microprocessador

Um microprocessador mais recente

- Intel Pentium 4 (2003)
 - Aplicação: desktop/server
 - Tecnologia: 0,09 μ (1/100)
 - 55M transístors (20K x)
 - 101mm² (10x)
 - 3,4GHz (10k x)
 - 1,2 Volts (1/10)
 - 64 bit (16x)
 - Pipeline de 22 estágios
 - 4 instruções por ciclo (superescalar)
 - Cache de dois níveis
 - Instruções com paralelismo de dados (SIMD), *hyperthreading*



Microprocessador

Onde a revolução está sendo usada?

- Largura do datapath (4 para 64 bit)
- Instruções mais poderosas
 - Amortiza o custo de busca e decodificação
 - Simplifica a programação
 - Outros?

Microprocessador

Onde a revolução está sendo usada?

- Largura do datapath (4 para 64 bit)
- Instruções mais poderosas
 - Amortiza o custo de busca e decodificação
 - Simplifica a programação
 - Outros?

Leitura:

- Artigo do Gordon Moore

Paralelismo implícito

A 2ª revolução

- Paralelismo em nível de instrução
 - Hardware fornece recursos de paralelismo, programadores tem que saber como usá-los
- Pipelining
 - Ajuda a aumentar a frequência do relógio
- Cache
 - Tornou-se necessário por causa do aumento na frequência

Paralelismo implícito

A 2ª revolução

- Vírgula flutuante integrada
- *Pipelines* mais profundos e saltos especulativos
- *Superescalar*
- Escalonamento dinâmico (execução fora de ordem)

Paralelismo implícito

A 2ª revolução

- Vírgula flutuante integrada
- *Pipelines* mais profundos e saltos especulativos
- *Superescalar*
- Escalonamento dinâmico (execução fora de ordem)

Falaremos mais ao longo do curso.

Paralelismo explícito

A 3ª revolução

- Suporte ao paralelismo ao nível de dados e de *threads*
- Primeiro uso: instruções vetoriais (Intel SSE)
 - Uma instrução faz 4 multiplicações em paralelo
- Suporte para programas multi-thread
 - Coerência de cache, primitivas de sincronização em hardware

Paralelismo explícito

A 3ª revolução

- Suporte para múltiplas threads concorrentes em um chip
 - Primeiro com single core, agora com multi core
- GPUs são altamente paralelas
 - Convergência com CPUs:
 - AMD comprou ATI,
 - Intel está fazendo GPUs
 - NVidia estaria pensando em CPUs?

Paralelismo explícito

A 3ª revolução

- Suporte para múltiplas threads concorrentes em um chip
 - Primeiro com single core, agora com multi core
- GPUs são altamente paralelas
 - Convergência com CPUs:
 - AMD comprou ATI,
 - Intel está fazendo GPUs
 - NVidia estaria pensando em CPUs?

Falaremos mais ao longo do curso.

Por quê estudar arquitetura de computadores

Para ponderar

A **“revolução multicore”** desta década é comparável à **“revolução do microprocessador”** original?

Por quê estudar arquitetura de computadores

Arquitetura para gerenciar a confusão

- Deve-se considerar todos fatores
 - Objetivos, Aplicações, Tecnologia
- Questões:
 - Como lidar com todas essas entradas?
 - Como gerenciar as mudanças?
- Respostas
 - Conhecimento institucional
 - Experiência, disciplina
 - Abstrações e camadas

Por quê estudar arquitetura de computadores

Arquitetura para gerenciar a confusão

- Deve-se considerar todos fatores
 - Objetivos, Aplicações, Tecnologia
- Questões:
 - Como lidar com todas essas entradas?
 - Como gerenciar as mudanças?
- Respostas
 - Conhecimento institucional
 - Experiência, disciplina
 - Abstrações e camadas

Mais na próxima aula

Por quê estudar arquitetura de computadores

Objetivos

- Entender para onde os computadores estão indo
 - Capacidades futuras
 - Impactos no mundo real: **sem arquitetura de computadores, sem computadores!**
- Entender conceitos de projeto em alto nível
 - Melhores projetistas compreendem todos os níveis
 - Dispositivos, circuitos, arquiteturas, compiladores, aplicações
- Conseguir melhores empregos
 - Melhores projetistas de sw entendem de hw
 - Muitos empregos para projetistas de hw

Por quê estudar arquitetura de computadores

Tópicos do curso

- Métricas de avaliação e tendências
- Revisão e extensão de conhecimentos da graduação
- Paralelismo de instrução, de dados e de threads
- Mais coisas divertidas, se tivermos tempo (computação reconfigurável!)

Planejamento do curso

Ementa, objetivos, programa, calendário, especificação de trabalhos, material de apoio

- www.facom.ufu.br/~mesquita
- mesquita@facom.ufu.br

Planejamento do curso

Bibliografia

- Computer Architecture, A Quantitative Approach. Hennessy & Patterson. Third Edition. Morgan Kaufmann.
- Organização Estruturada de Computadores. Andrew S. Tanenbaum. Quinta Edição. Pearson.
- Artigos selecionados.

Planejamento do curso

Avaliações

- 10% participação em aula
- 10% tarefas de casa
- 40% prova
- 40% artigo / trabalho

Planejamento do curso

Avaliações

- 10% participação em aula
- 10% tarefas de casa
- 40% prova
- 40% artigo / trabalho

Percentuais da prova e do artigo podem mudar em função do envolvimento da turma.

Planejamento do curso

Atendimento

- Preferencialmente nas quartas-feiras à tarde
- Sob agendamento

Recuperação

- Prova substitutiva para quem perder a prova
- Estudos dirigidos/trabalhos para recuperação de notas/aprendizagem

- [http://www.scribd.com/doc/7279033/
Patterson-Hennessy-Computer-
Architecture](http://www.scribd.com/doc/7279033/Patterson-Hennessy-Computer-Architecture)