

# Tokenização de ativos, tokens fungíveis e não fungíveis

Aplicações Descentralizadas e Blockchain

Prof. Carlos Leonardo dos S. Mendes



**PUC Minas**

# Tokenização de ativos

- A palavra **token** tem muitas traduções, como “símbolo”, “ficha” e “passe”.
- Inicialmente na TI, a palavra foi usada para representar um “passe”, como nas autorizações de internet banking por meio de um dispositivo físico (*token*).
- Esse tipo de *token* é um *token* de **direito de acesso** e hoje é fornecido em diferentes formas, como por SMS ou por e-mail.
- Os *tokens* também podem ser usados para representar um **ativo**. Um **ativo** é algum bem de valor, como uma moeda, um imóvel ou um contrato comercial.
- Quando o *token* é usado nesse contexto (**financeiro**), ele é um **token de ativo** ou simplesmente um **ativo digital**.

# Tokenização de ativos

- ▶ Nas plataformas de blockchain, *tokens* podem ser definidos por meio de **contratos inteligentes**.
- ▶ Um *token* pode representar um **ativo físico** como uma moeda fiduciária, um *ticket* de um evento ou uma medalha.
- ▶ Um *token* pode também representar um **ativo digital** como um vídeo, um *post* do Twitter ou o *download* de uma música.
- ▶ Podem ainda representar **outras classes de ativos**, como uma reputação, identidades digitais ou credenciais de acesso.

# Por que tokenizar?

- ▶ Porque a tecnologia de blockchain permitiu **eliminar o problema do gasto duplo** em transações com ativos digitais.
- ▶ Porque a tecnologia de blockchain permitiu transações de **intermediação direta**.
- ▶ Porque a tecnologia de blockchain permitiu **transações seguras** sem a necessidade de que as partes envolvidas confiem entre si.
- ▶ Porque **transações digitais são mais simples e baratas** do que transações físicas.

# Tokens fungíveis e o padrão ERC-20

# Tokens fungíveis

- ▶ Um ativo possui **fungibilidade** quando suas unidades podem ser trocadas entre si de maneira indistinta.
- ▶ São exemplos de ativos fungíveis: commodities, moedas fiduciárias, metais preciosos, criptomoedas.
- ▶ Exemplos:
  - ▶ o Ether é um ativo fungível porque cada unidade de ETH é equivalente a qualquer outra unidade de ETH.
  - ▶ O ouro é um ativo fungível porque 1g de ouro 24K é igual a qualquer outro grama de ouro 24K, podendo ser trocados indistintamente.
- ▶ Grande parte dos tokens criados em plataformas de blockchain são usados para representar ativos fungíveis.

# O padrão ERC-20

- O **padrão ERC-20** foi criado na plataforma Ethereum, que é hoje a plataforma mais usada para a tokenização de ativos.
- O ERC-20 define um padrão de codificação para um **token fungível**.
- Ao seguir o padrão ERC-20, os tokens criados são automaticamente **interoperáveis** com serviços e aplicativos que suportam este padrão (ex: carteiras digitais, corretoras, etc.).
- Um **contrato compatível com o padrão ERC-20** possui:
  - 6 funções obrigatórias: *totalSupply*, *balanceOf*, *transfer*, *transferFrom*, *approve* e *allowance*
  - Funções opcionais: *name*, *symbol*, *decimal*.
- Uma descrição detalhada do padrão ERC-20 pode ser consultada em <https://ethereum.org/en/developers/tutorials/erc20-annotated-code/>

# Tokens não fungíveis e o padrão ERC-721



# Tokens não fungíveis

- ▶ Um ativo **não fungível** é um ativo que não pode ser substituído por outro da mesma espécie, quantidade e qualidade.
- ▶ São exemplos de ativos não fungíveis: obras de arte, bens produzidos em série que foram personalizados ou objetos raros.
- ▶ A criação da tecnologia de blockchain introduziu o conceito de **escassez digital**, permitindo mapear a escassez existente no mundo real para o mundo digital.
- ▶ **Tokens não fungíveis** (ou criptocolecionáveis) são **tokens únicos** e cada um tem uma **quantidade limitada**.
- ▶ Um token não fungível representa um **ativo exclusivo**.

# O padrão ERC-721

- O **padrão ERC-721** também foi criado na plataforma Ethereum.
- O ERC-721 define um padrão de codificação para um **token não fungível**.
- Ao seguir o padrão ERC-721, os tokens criados são automaticamente **interoperáveis** com serviços e aplicativos que suportam este padrão (ex: corretoras de criptocolecionáveis).
- Um **contrato compatível com o padrão ERC-721** deve possuir as funções:
  - *balanceOf(owner), ownerOf(tokenId), safeTransferFrom(from, to, tokenId), transferFrom(from, to, tokenId), approve(to, tokenId), getApproved(tokenId), setApprovalForAll(operator, \_approved), isApprovedForAll(owner, operator), safeTransferFrom(from, to, tokenId, data)*
- Uma descrição detalhada do padrão ERC-20 pode ser consultada em <https://ethereum.org/en/developers/tutorials/erc-721-vyper-annotated-code/>



**PUC Minas**