



# Moonbeam开发进阶课程

## XCM介绍和原理

---

PureStake开发者关系团队



# 课程导航

## XCM技术概述

- 频道
- 交易费用
- 主权账户

## XC-20标准

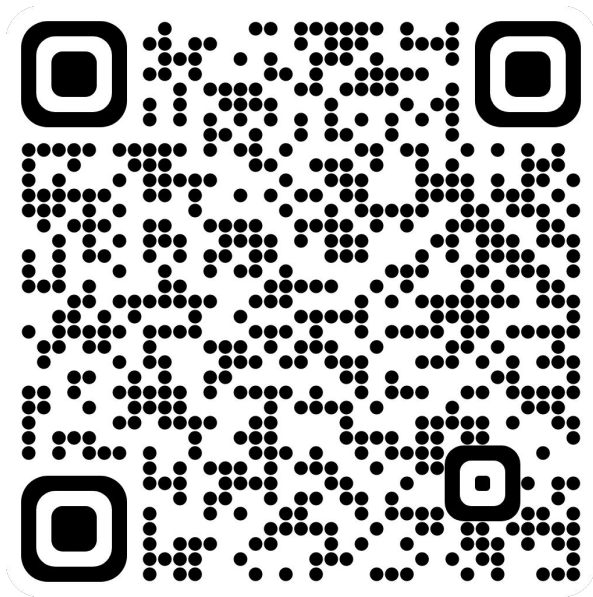
## 资产代表方式

- Multi-location
- 资产ID

## 转帐流程

## 课程资源GitHub Repo

<https://github.com/PureStake/moonbuilders-academy/tree/main/chinese/advanced-course>





# XCM: 技术概述

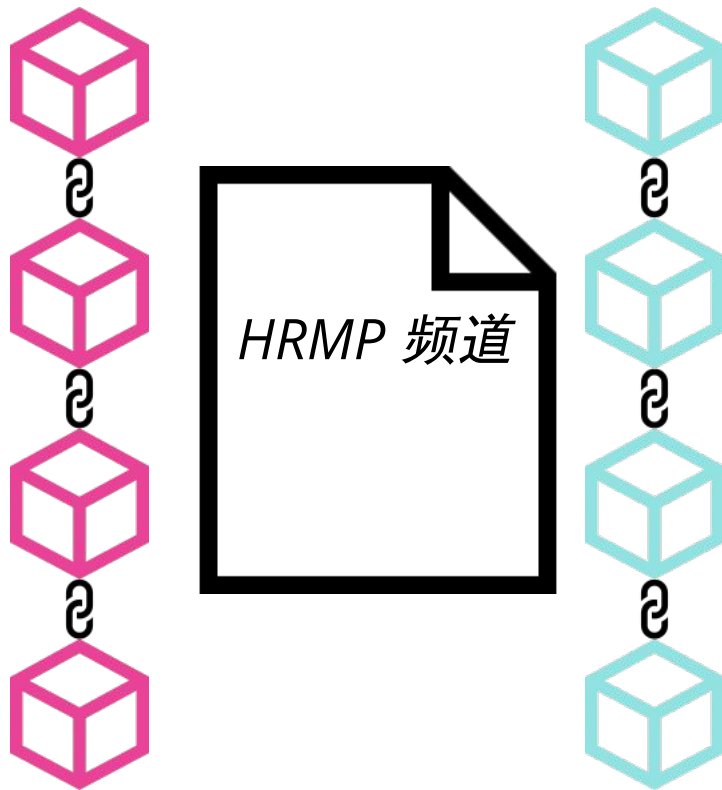
# 什么是XCM？

○○○ **跨共识信息格式**  
○○○ 通用语言定义

○○○ **可延伸性**  
○○○ 无主观性(non opinionated)

○○○ **向前和向后兼容性**  
○○○ 可以在信息中标注XCM版本号

○○○ **轻量而高效**  
○○○ 支持多链异构, 底层基于XCVM语言



# 传输协议

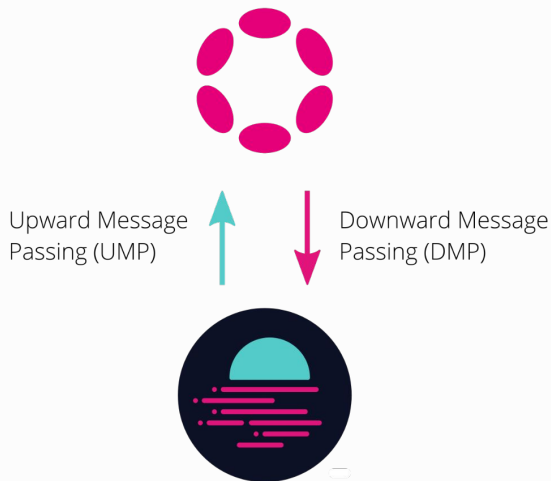
## XCM信息如何传递

○○○ **VMP**  
○○○ 中继链 - 平行链

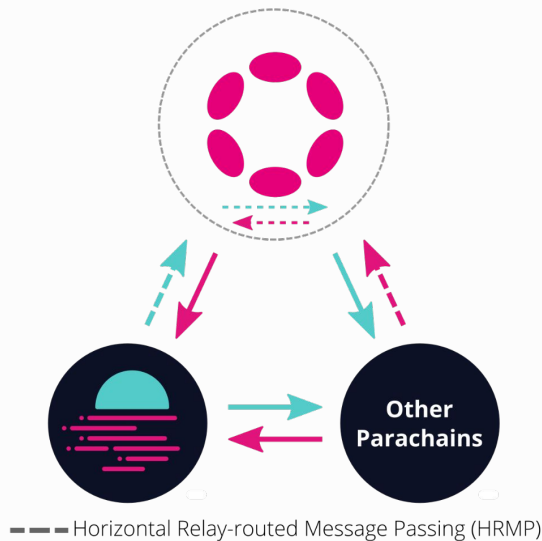
○○○ **XCMP**  
○○○ 平行链 - 平行链

○○○ **HRMP**  
○○○ 平行链 - 平行链信息传递的临时解决方法

Vertical Message Passing (VMP)



Cross-Chain Message Passing (XCMP)

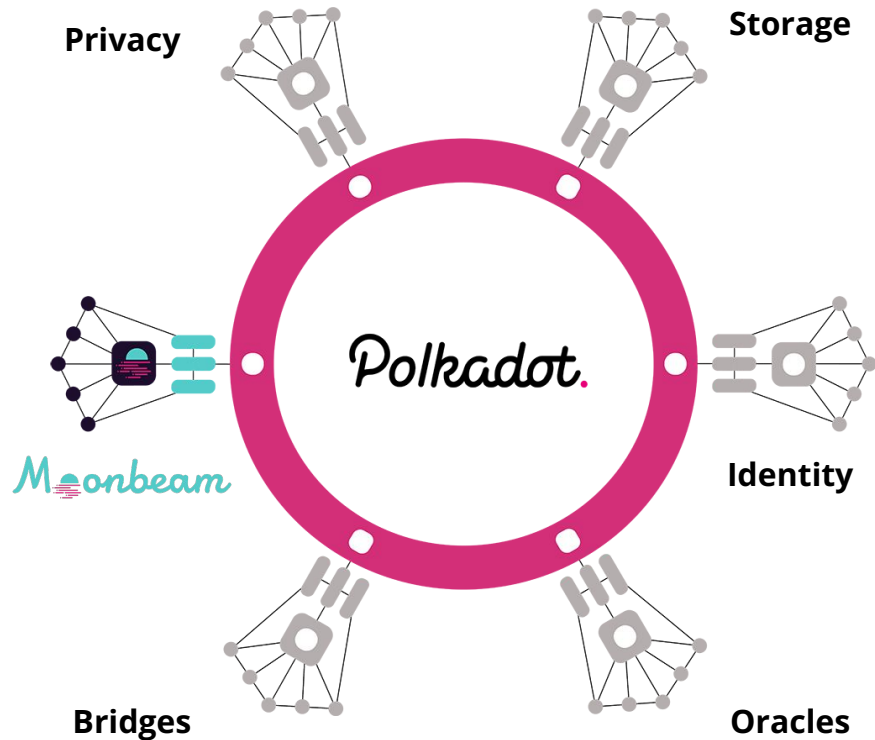




XCM频道

## XCM频道概述

- 频道是单方向的  
所以如果需要双向通信，需要开两个频道
- 与中继链之间的频道 ✓  
在平行链连接时默认打开
- 发起和接受请求需要从root用户发出  
可以通过治理或Sudo
- 两条链之间最多有两个频道  
一个发送信息，一个接收信息





## XCM频道注册

### 注册频道信息格式

parachain ID,  
`max-capacity`,  
`max-message-size`

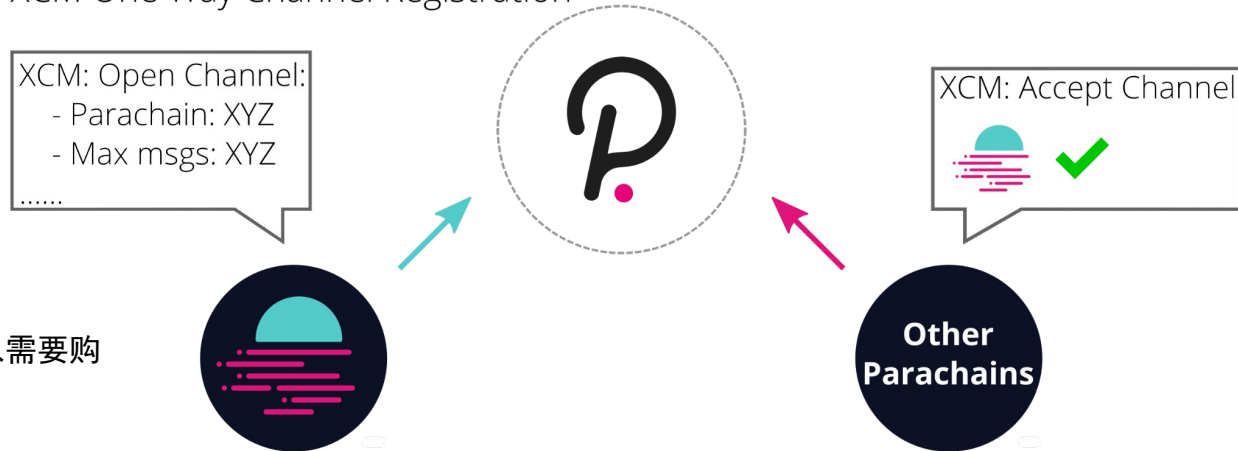
### 购买中继链执行时间

注册频道需要在中继链执行, 所以需要购买中继链的执行时间

### 频道注册需要绑定一定DOT或KSM

波卡: 20 DOT; Kusama: 10 KSM  
可以用 `chainstate.configuration.activeConfig` 来查询

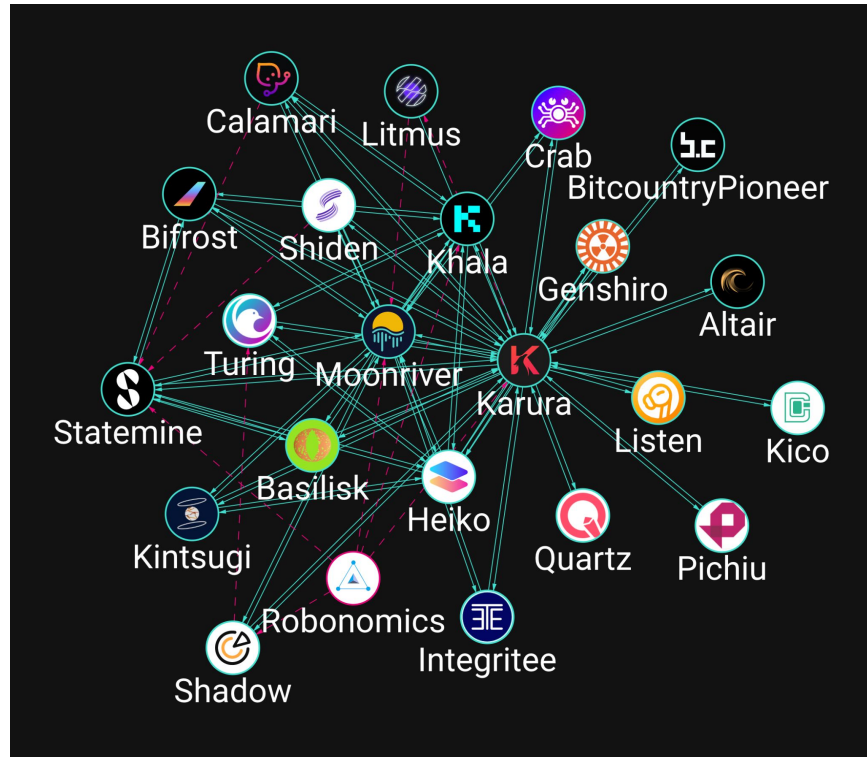
### XCM One Way Channel Registration



## HRMP频道可视化工具



<https://dotsama-channels.vercel.app/#/>





# XCM频道信息查询



# 交易费用计算

# 灵活收费机制

## 可以用任何资产支付交易费用

### ○○○ 购买执行时间

○○○ 符合以下条件的资产可以用来支付交易费用

1. Holding Register持有此资产
2. 目标链接接收此资产

### ○○○ 多种付费策略


1. 只接收原生代币
2. 接收指定的非原生代币
3. 接收非原生币, 但会转换为原生币
4. 不受交易费




# XCM操作指令

## 每个操作指令都有相应的重量(执行成本)

### WithdrawAsset

 将资产转入Holding Register

### TransferAsset

 将资产转移到某一位受益人


### BuyExecution

 使用Holding Register内的资金购买执行时间


### DepositAsset

 从Holding Register内移出资产

### Transact

 尝试在链上调用某一个方法

### SubscribeVersion

 要求更改目标链支持的XCM版本

	XCM执行重量
Moonbeam	200,000,000
Moonriver	200,000,000
Moonbase Alpha	100,000,000



# 主权账户

# 什么是 主权账户？

○○○ 平行链拥有的账户  
○○ 在另外一条平行链上

○○○ 持有正在发送的资金  
○○ 类似以太坊桥的锁定合约

○○○ 地址是由确定性算法生成  
○○ 衍生于Parachain ID

```
const sovAddressRelay = u8aToHex(  
  new Uint8Array([...new TextEncoder().encode('para'),  
    ...targetParaId.toU8a()])  
).padEnd(66, '0');  
  
const sovAddressPara = u8aToHex(  
  new Uint8Array([...new TextEncoder().encode('sibl'),  
    ...targetParaId.toU8a()])  
).padEnd(66, '0');  
  
console.log(`Sovereign Account Address on Relay:  
${sovAddressRelay}`);  
console.log(`Sovereign Account Address on other  
Parachains (Generic): ${sovAddressPara}`);  
console.log(`Sovereign Account Address on Moonbase Alpha:  
${sovAddressPara.slice(0, 42)}`);
```



# 如何计算 主权账户?



## xcmTools Repo

<https://github.com/PureStake/moonbuilders-academy/blob/main/chinese/advanced-course/week2-XCM-introduction/xcmTools/calculateSovereignAddress.ts>

```
purestake@moonbeam:~/data/xcmTools$ ts-node calculateSovereignAddress.ts --paraid 1000
Sovereign Account Address on Relay: 0x70617261e8030000000000000000000000000000000000000000000000000000
Sovereign Account Address on other Parachains (Generic): 0x7369626ce8030000000000000000000000000000000000000000000000000000
Sovereign Account Address on Moonbase Alpha: 0x7369626ce8030000000000000000000000000000000000000000000000000000
```



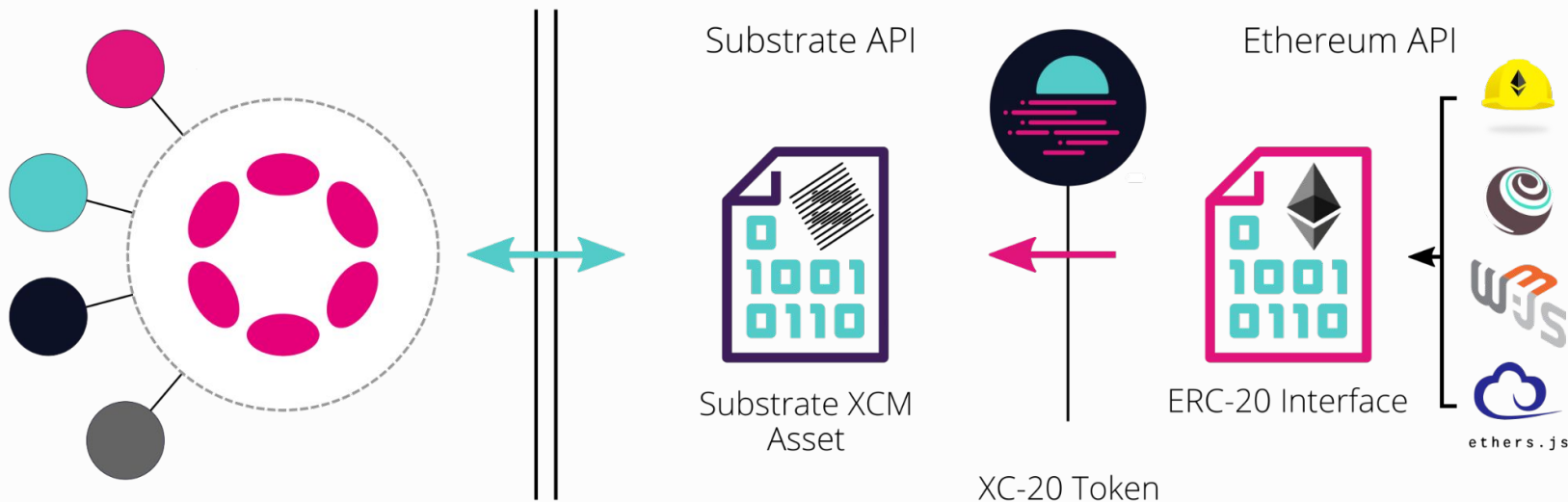
# Moonbeam XC-20标准

# XC-20s

## Moonbeam的以太坊兼容解决方案

Substrate资产  
原生兼容XCM转账

ERC-20接口  
可以轻松集成到Moonbeam EVM生态系统






# Moonbeam上现有的XC-20资产

Moonriver:

<https://polkadot.js.org/apps/?rpc=wss%3A%2F%2Fwss.api.moonriver.moonbeam.network#/assets>

Moonbeam:

<https://polkadot.js.org/apps/?rpc=wss%3A%2F%2Fwss.api.moonbeam.network#/assets>

assets		owner	supply
32,615,670,524,745,285,411,807,346,420,584,982,855	PARA	 0x6D6f...000000	3,709.1758 XCPARA
42,259,045,809,535,163,221,576,417,993,425,387,648	xcDOT	 0x6D6f...000000	931,063.5816 xcdOT
110,021,739,665,376,159,354,538,090,254,163,045,594	Acala Dollar	 0x6D6f...000000	250,014.6640 XCAUSD
224,821,240,862,170,613,278,369,189,818,311,486,111	Acala	 0x6D6f...000000	62,514.8536 XCACA

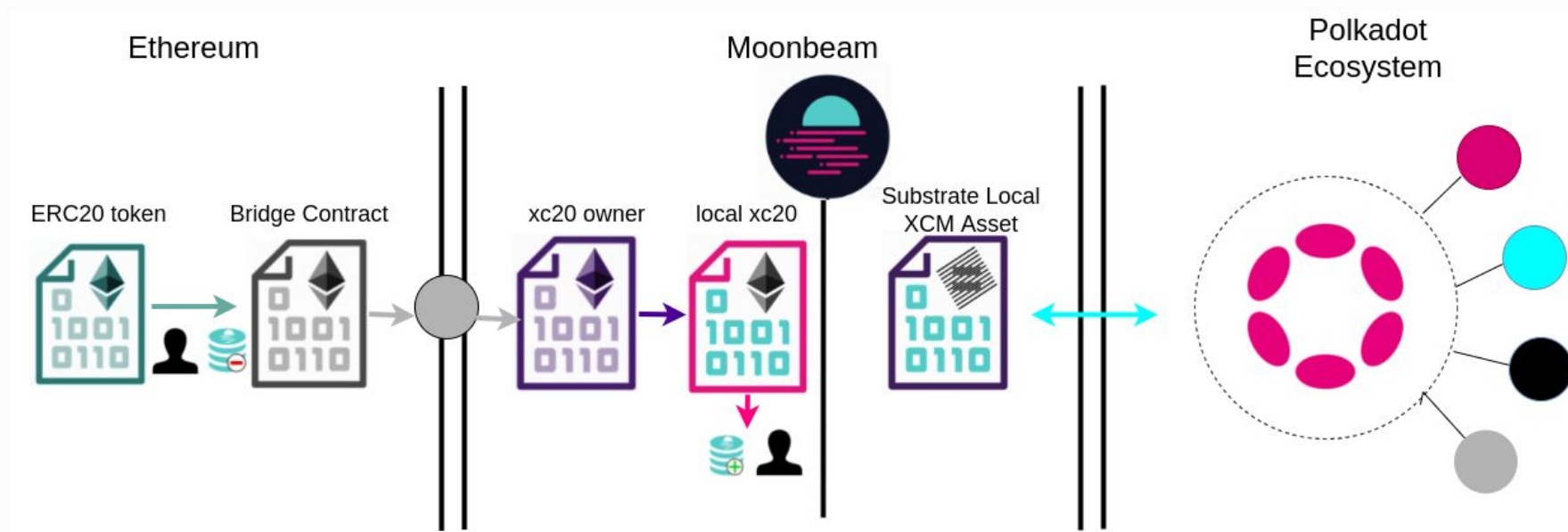
# 可铸造XC-20

使用Substrate Assets pallet  
铸造

同时在Substrate注册这个资产

通过EVM管理  
可以与智能合约直接交互

XCM  
可以轻松跨链交易





# XC-20资产代表方式

# 代表方式#1: Multilocations

## 一种在波卡生态系统中代表“位置”的方法



### 类似文件存储的路径系统

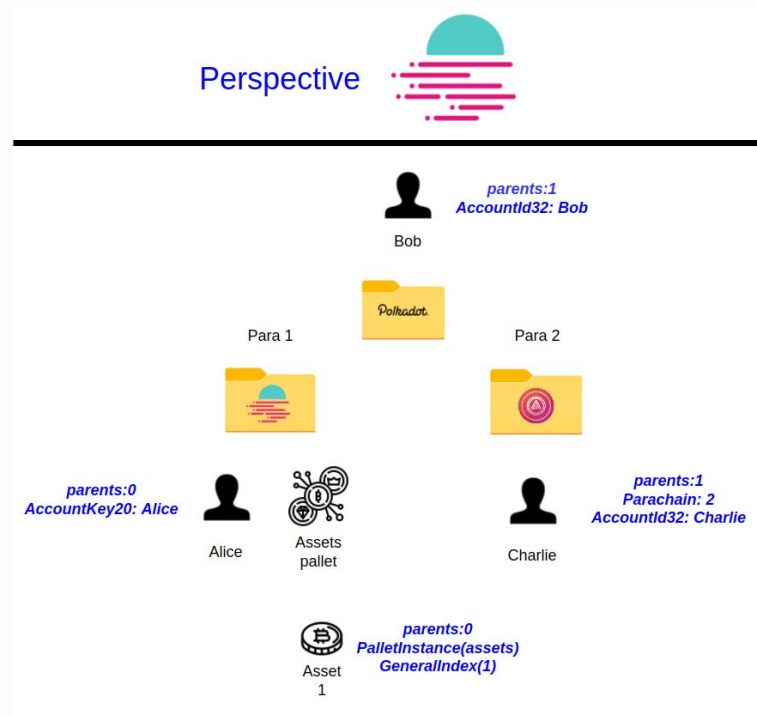
1. 采用相对路径命名方式
2. 本地的资产和账户不用穿越“parents”或中继链
3. 其它链上的账户和资产则需要



### 资产，账户和并行链可以由 MultiLocation 来代表



### 因为是相对路径，所以需要考虑调用的起源点在哪里



# 代表方式#1: Multilocations

## 一种在波卡生态系统中代表“位置”的方法



### 类似文件存储的路径系统

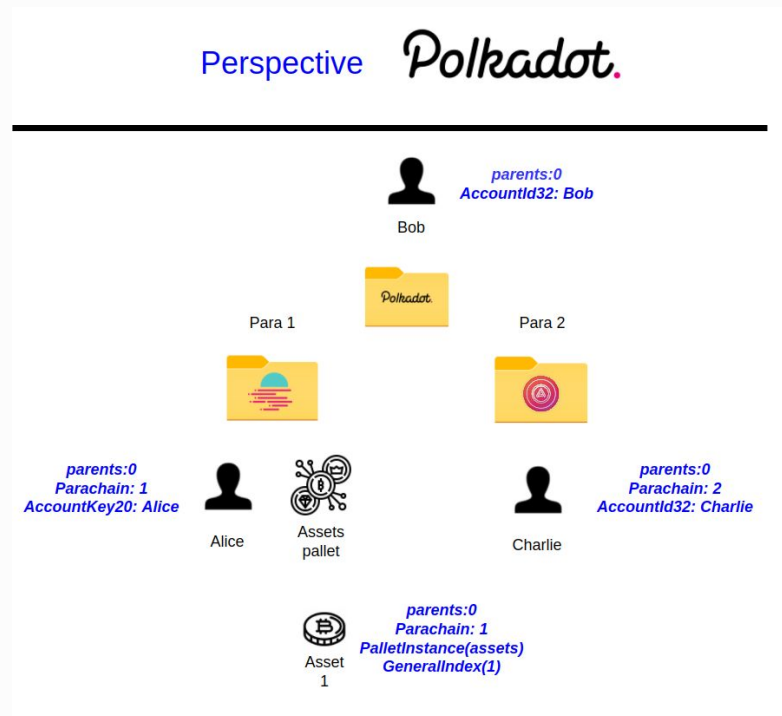
1. 采用相对路径命名方式
2. 本地的资产和账户不用穿越“parents”或中继链
3. 其它链上的账户和资产则需要



### 资产，账户和并行链可以由 MultiLocation 来代表



### 因为是相对路径，所以需要考虑调用的起源点在哪里





# Multilocation: 数据结构

- 组成部分: parents 和 interior
- parents = 0 或 1, 代表是否需要跨越中继链
- interior 代表定义目标位置; X\_ 代表需要几个字段, X1 代表一个字段, X2 代表两个...
- 
- 举例:
- 从中继链到一条 Parachain ID 是“1000”的平行链:  
`{ "parents": 0, "interior": { "X1": { "Parachain": 1000 } }}`
- 从某一条平行链到一条 Parachain ID 是“1000”的平行链:  
`{ "parents": 1, "interior": { "X1": { "Parachain": 1000 } }}`
- 从某一条平行链到一条 Parachain ID 是“1000”的平行链上的 PalletInstance 3:  
`{ "parents": 1, "interior": { "X2": [ { "Parachain": 1000 }, { "PalletInstance": 3 } ] }}`
- 从某一条平行链到中继链的原生代币 (Interior = “Here” = [] = null):  
`{ "parents": 1 }`
- 从某一条平行链到一条 Parachain ID 是“1000”的平行链的 AccountKey20 账户:  
`{ "parents": 1, "interior": { "X2": [ { "Parachain": 1000 }, { "AccountKey20": { "network": "Any", "key": "0xf24FF3a9CF04c71Dbc94D0b566f7A27B94566cac" } } ] }}`

## 代表方式#2: 资产ID (AssetID)

- 也称为Currency ID
- 10进制数字
- SelfReserve指本地的原生代币
- ForeignAsset指XC-20的资产ID
- 可以从Multilocation计算
- 在Polkadot.js apps钱包的  
Network/Asset下可以查看

### assets

10,810,581,592,933,651,521,121,702,237,638,664,357	Karura
42,259,045,809,535,163,221,576,417,993,425,387,648	xcKSM
76,100,021,443,485,661,246,318,545,281,171,740,067	HKO
105,075,627,293,246,237,499,203,909,093,923,548,958	TEER
108,457,044,225,666,871,745,333,730,479,173,774,551	Crust Shadow Native Token
173,481,220,575,862,801,646,329,923,366,065,693,029	Crab Parachain Token
175,400,718,394,635,817,552,109,270,754,364,440,562	Kintsugi Native Token
182,365,888,117,048,807,484,804,376,330,534,607,370	xcRMRK
189,307,976,387,032,586,987,344,677,431,204,943,363	Phala Token
213,357,169,630,950,964,874,127,107,356,898,319,277	Calamari
214,920,334,981,412,447,805,621,250,067,209,749,032	Acala Dollar
311,091,173,110,107,856,861,649,819,128,533,077,277	Tether USD
319,623,561,105,283,008,236,062,145,480,775,032,445	xcBNC
328,179,947,973,504,579,459,046,439,826,496,046,832	Kintsugi Wrapped BTC

## 代表方式#3: XC-20预编译合约地址

- 从Multilocation和资产ID计算

`address = "0xFFFFFFFF..." + DecimalToHex(AssetId)`

- 以太坊合约地址格式
- 代表某一个X-20资产的ERC-20预编译地址
- 例子:

Moonbeam

Moonriver

Moonbase Alpha

Origin	Symbol	XC-20 Address
Polkadot	xcDOT	0xFfFfFff1FcaCBd218EDc0EbA20Fc2308C778080
Acala	xcaUSD	0xFfFFFFF52C56A9257bB97f4B2b6F7B2D624ecda
Acala	xcACA	0xfFFFFffa922Fef94566104a6e5A35a4fCDDAA9f



# 计算资产ID和XC-20预编译地址



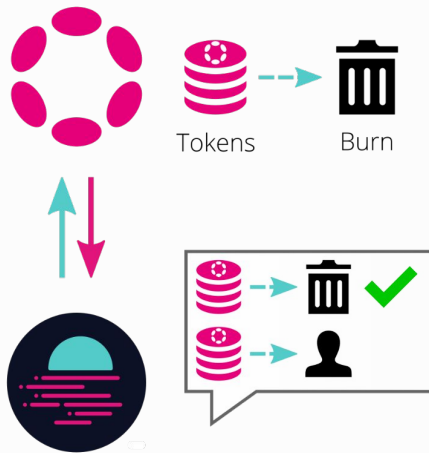
# XCM转帐流程

# XCM转帐机制

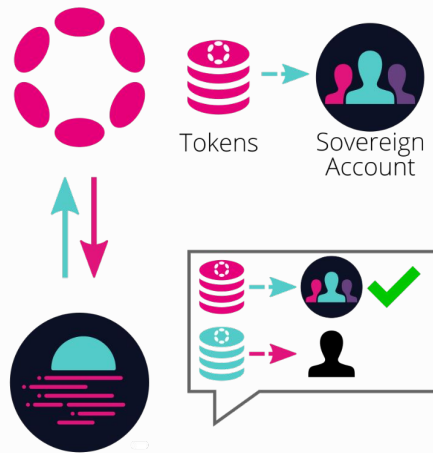
○○○ 传送  
○○○ 焚毁 - 铸造

○○○ 远程转帐  
○○○ 锁定-铸造/焚毁-解锁  
主权账户

Asset Teleporting

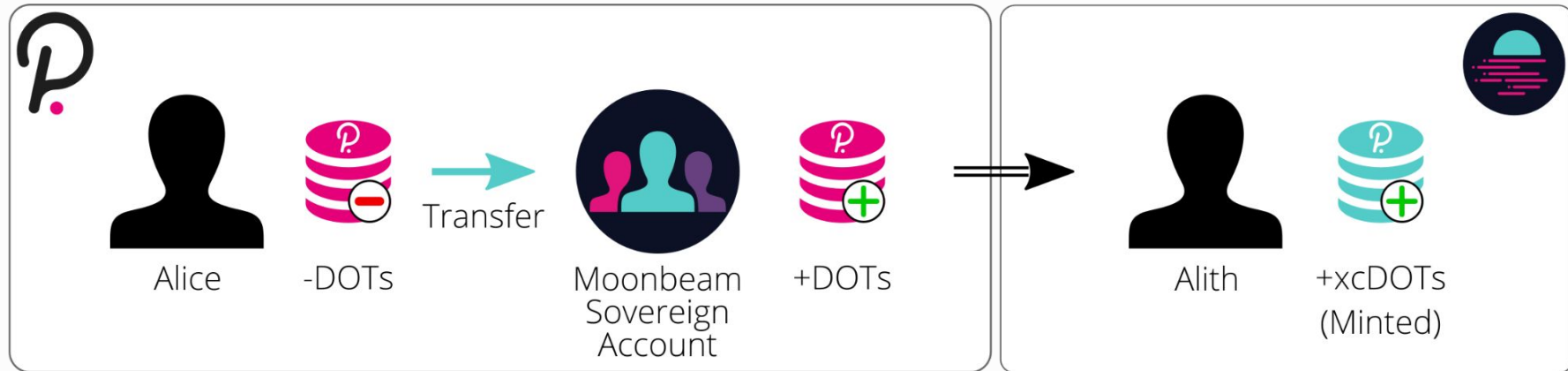


Remote Transfers



# 从中继链向平行链转帐

Asset Transfer from Polkadot (Alice) to Moonbeam (Alith)



# 平行链向平行链转帐

GLMR Transfer from Moonbeam (Alith) to Another Parachain (Alice)

