

進階概念與合約部署

Jarek from Bucket Protocol



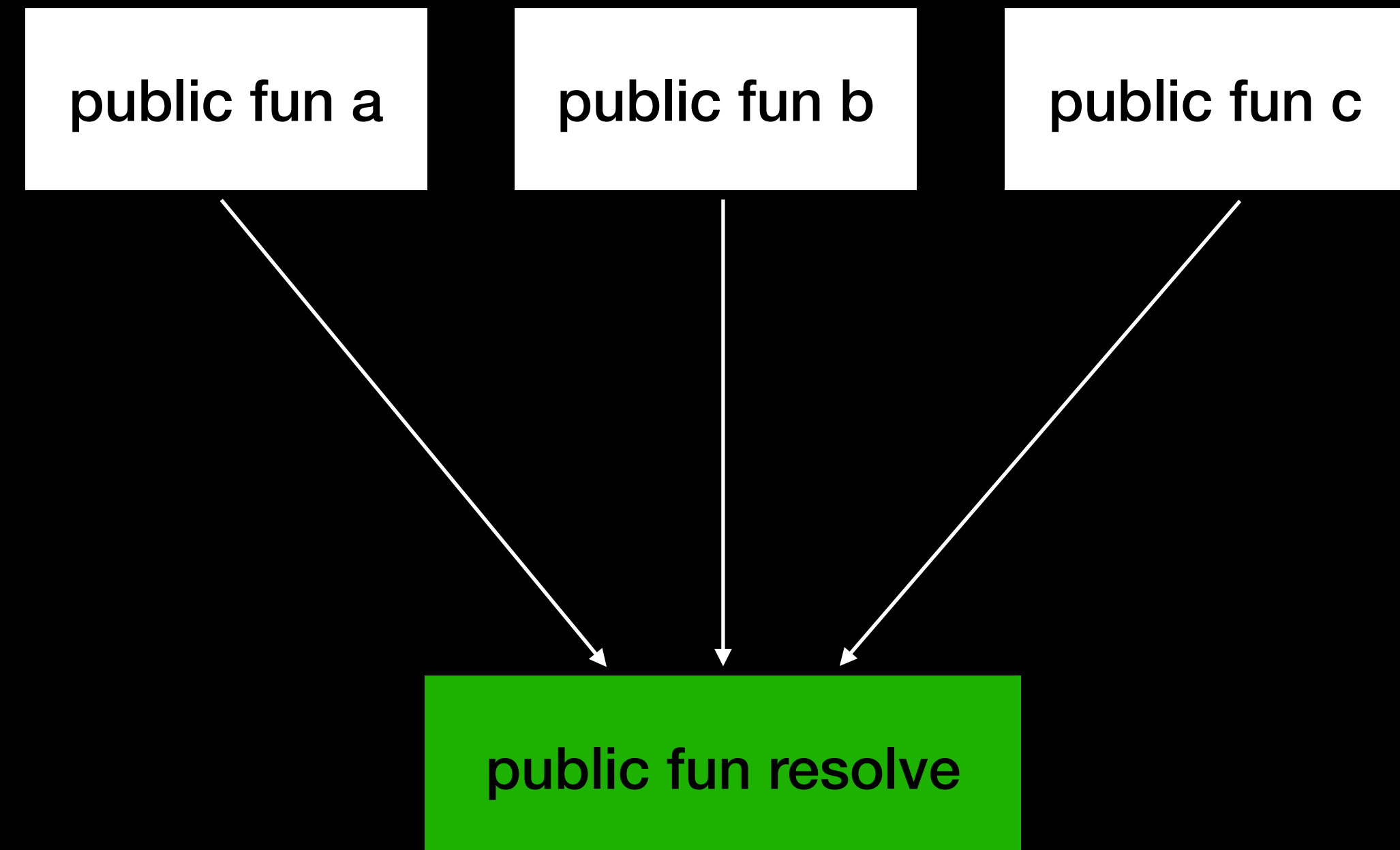
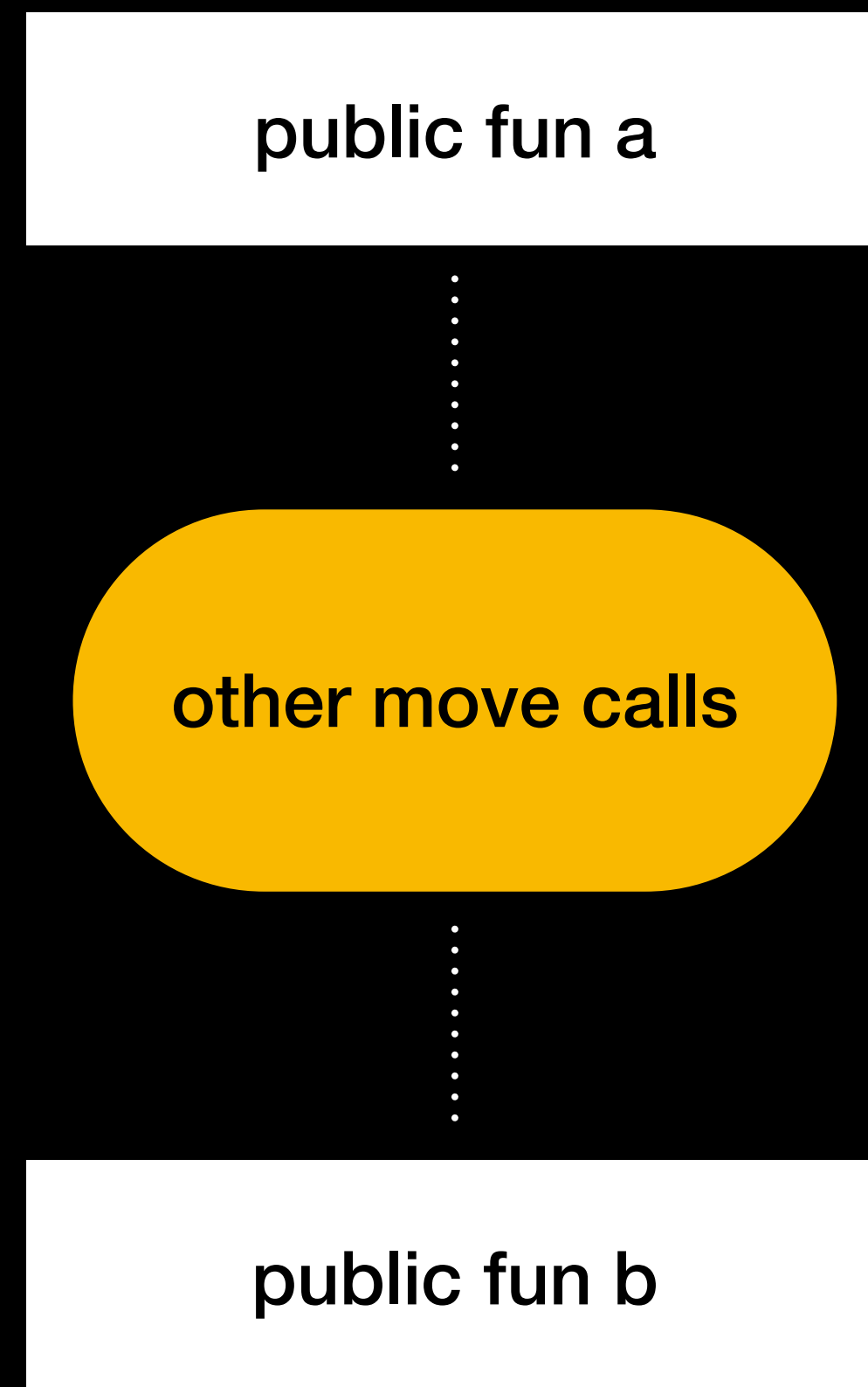
Bucket

大綱

- Hot Potato
- Dynamic Fields
- How to Publish

Hot Potato

何時會用到



Hot Potato

定義：不具有任何能力

```
public struct LoanPool has key {  
    id: UID,  
    amount: Balance<SUI>,  
}  
  
/// Hot Potato  
public struct Loan {  
    amount: u64  
}
```

Hot Potato

生成

```
public fun borrow(  
    pool: &mut LoanPool,  
    amount: u64,  
    ctx: &mut TxContext  
): (Coin<SUI>, Loan) {  
    assert!(amount <= balance::value(&pool.amount), ELoanAmountExceedPool);  
  
    let coin = coin::from_balance(pool.amount.split(amount), ctx);  
    let loan = Loan {  
        amount  
    };  
  
    (coin, loan)  
}
```

Hot Potato

消耗

```
public fun repay(pool: &mut LoanPool, loan: Loan, payment: Coin<SUI>) {  
    let Loan { amount } = loan;  
    assert!(coin::value(&payment) == amount, ERepayAmountInvalid);  
  
    pool.amount.join(payment.into_balance());  
}
```

Dynamic Fields

讓資料可以安插在 object 上

```
public fun add_info(  
    kapy: &mut Kapy,  
    age: u8,  
    gender: bool,  
    height: u64  
) {  
    let info = KapyWrappedInfo {  
        age,  
        gender,  
        height  
    };  
  
    df::add(&mut kapy.id, std::type_name::get<KapyWrappedInfo>(), info);  
}
```

Dynamic Fields

讓資料可以安插在 object 上

```
public fun equip_hat(  
    kapy: &mut Kapy,  
    color: String,  
    size: u8,  
    ctx: &mut TxContext  
) {  
    let hat = Hat {  
        id: object::new(ctx),  
        color,  
        size  
    };  
  
    dof::add(&mut kapy.id, std::type_name::get<Hat>(), hat);  
}
```


How to Publish

- sui client publish
- 紀錄Package ID, UpgradeCap ID
- 紀錄初始Created Objects
- 更改 Move.toml
 - [package] (published-at=
 - [addresses]