

## Mapping Unicrow Entities to w3rk:

- **Buyer (in Unicrow)**: This corresponds to the **Client** on w3rk who is hiring a freelancer for a project.
- **Seller (in Unicrow)**: This corresponds to the **Freelancer** on w3rk who is providing services in return for payment.
- **Marketplace (in Unicrow)**: This is **w3rk** itself, acting as the platform that facilitates the transaction between the Client and Freelancer.

## How the Transaction Works:

### 1. Contract Creation:

- When a Client on w3rk initiates a contract, they act as the **Buyer**. They propose a contract with a specific payment amount (e.g., 100 USDC).
- The Client then proceeds to pay, where an additional 5% fee is calculated. The total amount (e.g., 105 USDC) is deducted from the Client's wallet.
- Using the Unicrow SDK, specifically the `pay` function from the `ui` module, the total payment is split:
  - **5% Fee**: Automatically sent to the w3rk Organization Wallet.
  - **95% (100 USDC)**: Held in Unicrow's escrow, linked to the specific contract, until the work is completed.

### 2. Escrow Management:

- The payment held in escrow will remain locked until the work is completed. This corresponds to the **Work Approved** phase in our transaction flow.
- The Freelancer (Seller) submits their work through our platform (likely using XMTP for messaging), and the Client can then review it.
- Depending on the Client's actions, different functions from the Unicrow SDK will be invoked:
  - **Approval**: The Client approves the work, which triggers the `release` function. This releases the escrowed funds minus a 5% fee, which is sent to the w3rk Rewards Wallet. The remaining 95% is transferred to the Freelancer's wallet.
  - **Revision Requests or Disputes**: If revisions are requested or a dispute arises, the contract remains in escrow until resolved.

### 3. Challenges, Dispute Management, and Settlements:

- If a dispute occurs, we want to include the ability for the Client to raise a **challenge**. This can be done using the `challenge` function provided by the Unicrow SDK.
- The escrowed funds will then be temporarily locked while the challenge is resolved. We aim to handle disputes and settlements **without involving a**

- third party, allowing the Client and Freelancer to negotiate and resolve the issue directly through the platform.
- Depending on the outcome of the dispute, the funds can be split accordingly:
  - In favor of the Freelancer:** If the challenge is resolved in favor of the Freelancer, the remaining 95% of the funds are released to them.
  - In favor of the Client:** If the Client's challenge is successful, they could be refunded a portion or all of the escrowed funds, depending on the resolution terms.

## Final Outcome:

- Client:** Paid 105 USDC.
- w3rk Org Wallet:** Received 5 USDC.
- w3rk Rewards Wallet:** Received 5 USDC (if work is approved).
- Freelancer:** Received 95 USDC (if work is approved).

## Settlement Process:

In the event of a dispute between the Client (Buyer) and the Freelancer (Seller), we want to enable both parties to reach a mutual settlement agreement without involving a third party. Here's how this should work:

### 1. Mutual On-Chain Consensus:

- Both the Client and Freelancer must agree on the terms of the settlement. For example, if they agree that the Freelancer will offer a discount to the Client, they need to define the percentage of the payment that will be returned to the Client and what percentage the Freelancer will retain.

### 2. Settlement Offers:

- Once they agree, both parties will submit matching settlement offers to the Unicrow protocol's smart contract. The offer defines how the escrowed funds should be split.
- For example:
  - If they agree that 20% of the payment should go back to the Client and 80% should be released to the Freelancer, the Client would submit an offer stating this 20/80 split.
  - The Freelancer must then submit an identical offer with the same parameters.

### 3. Settlement Execution:

- The Unicrow smart contract will verify that both parties have submitted matching settlement offers.
- If the offers match, the smart contract will execute the settlement automatically, splitting the escrowed funds as agreed:
  - The agreed percentage (e.g., 20%) will be refunded to the Client's wallet.
  - The remaining amount (e.g., 80%) will be released to the Freelancer's wallet.

## **Example in w3rk Context:**

- **Scenario:** A Client hires a Freelancer for 100 USDC. After work submission, the Client is partially satisfied and agrees with the Freelancer that 20% of the payment should be refunded.
- **Action:**
  1. The Client sends a settlement offer stating that 20 USDC should be returned to them, and 80 USDC should be released to the Freelancer.
  2. The Freelancer submits an identical settlement offer.
  3. The Unicrow smart contract checks both offers. If they match, the contract processes the settlement:
    - **20 USDC** is refunded to the Client.
    - **80 USDC** is released to the Freelancer.