

OmniBOLT Guide I

Smart asset lightning network, off-chain contract system
for DeFi

Drafted by: Neo Carmack

Github: <https://github.com/omnilaboratory>

Powered By  Omni

What is OmniBOLT?

1. Lightning network for smart assets, issued on OmniBOLT via OmniLayer;
2. Build upon BTC/OmniLayer network, implements HTLC operations on the graph/network of Poon-Dryja channels;
3. Uses atomic swap to exchange assets among networks, Ind and obd;
4. Off chain smart contract written in Turing complete script language for various finance scenarios. Easily embed into existing applications;
5. HD light client mechanism for secure wallet operations;
6. Lastest update: <https://github.com/omnilaboratory/OmniBOLT-spec>

Rationale of OmniBOLT

1. Build upon OmniLayer, proven secure, stable and anti-censor technology;
2. Constructs lightning channel by raw Omni/BTC transactions:
 - a) Seemlessly work with omnicore, smart assets payloads and validation managed by omnicore;
 - b) Lightning channel backed by 40+ transactions naturally operates assets without any extra cost of validation;
 - c) Extends to popular public chains, including Ethereum(soon future);
3. Scalability and speed proved by lightning theory;
4. HTLC chains asset channels to be a network, so that interoperation among assets networks is necessary. For example usdt network and btc network;
5. Much more flexible and extensible finance applications require quick and cheap smart asset/contract solutions like OmniBOLT;

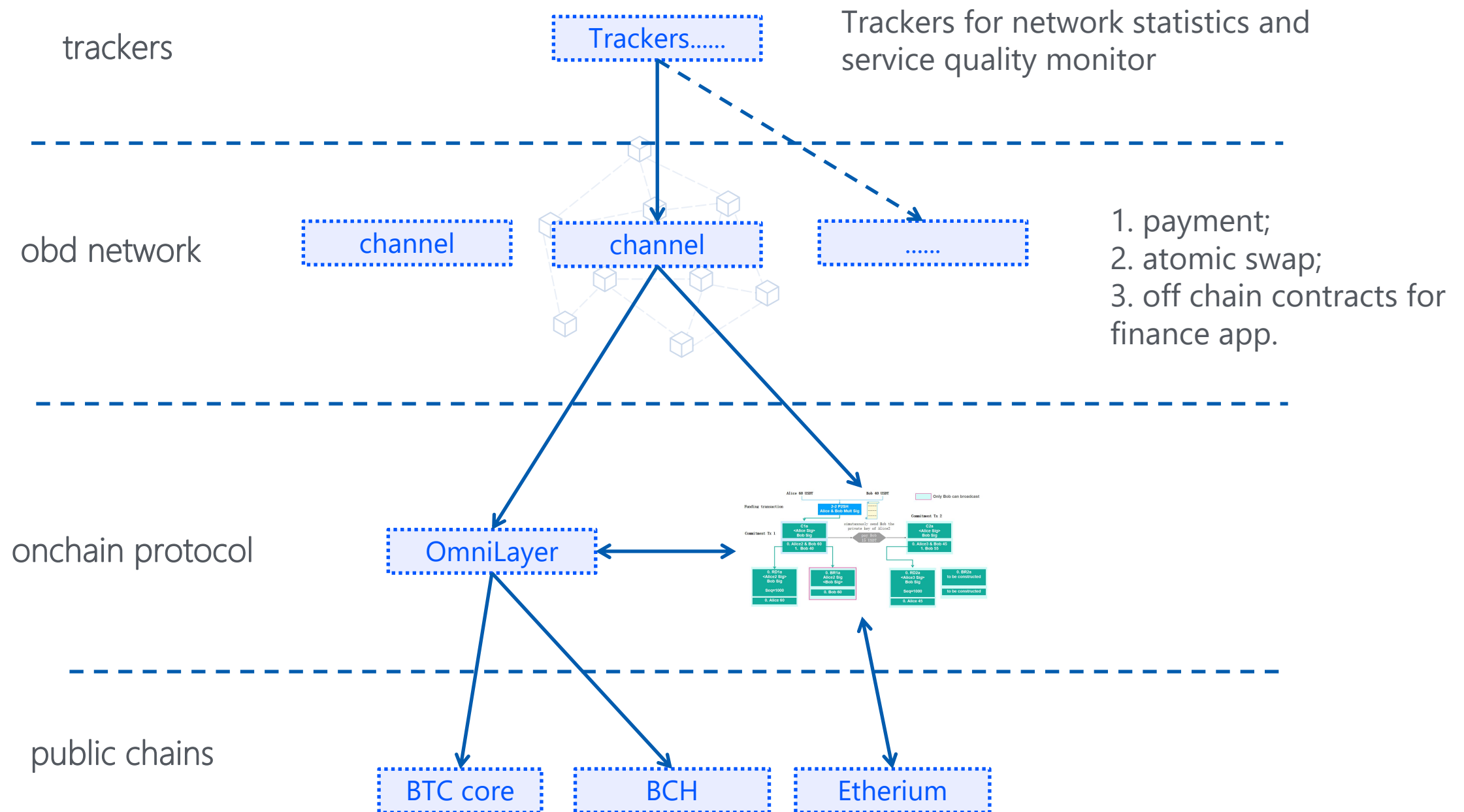
Advantages of OmniBOLT

1. Technical ready. OBD testnet is online in May, 2020, and client API, which are highly similar to Ind client, is releasing, so that integration costs will be minimal;
2. Instant payment of smart assets issued on OmniLayer and Ethereum(soon future) with the natural advantage of lightning theory.
3. Cross channel atomic swap of different assets.
4. Decentralized exchange on top of lightning channels with quick exchange speed and almost zero fee.
5. Collateral Lending Contract and more flexible contracts for various DeFi scenarios based on atomic swap, without any extra cost of transaction fee or any intermediary;

Architecture

Architecture of OmniBOLT

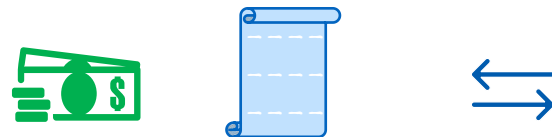
—————→ Data request



Protocol Suite

→ Data request

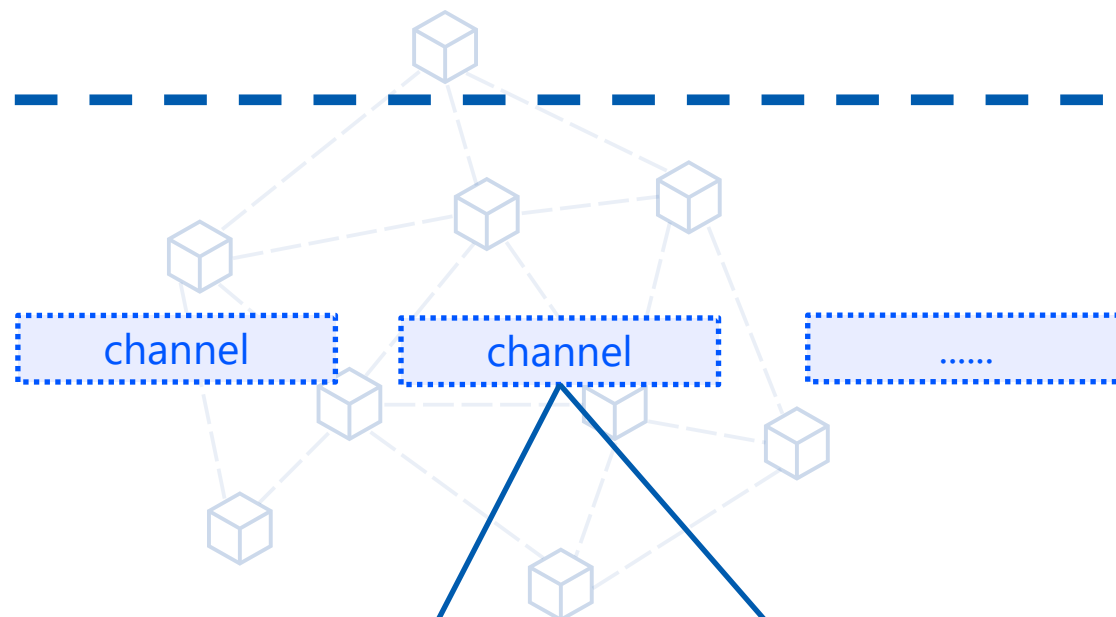
Contracts and Applications



OmniBOLT #05: Atomic Swap Protocol among Channels

OmniBOLT #06: DEX, Collateral Lending Contract, online store and more applications

obd network

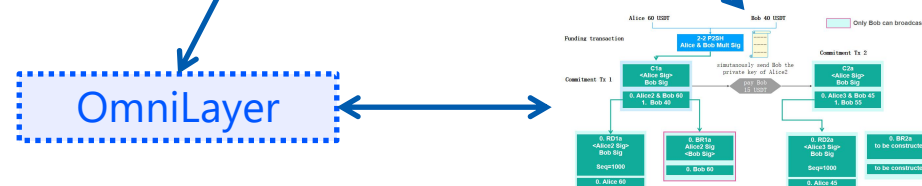


OmniBOLT #02: peer-protocol, Poon-Dryja channel open

OmniBOLT #03: RSMC and OmniLayer Transactions

OmniBOLT #04: HTLC and payment Routing

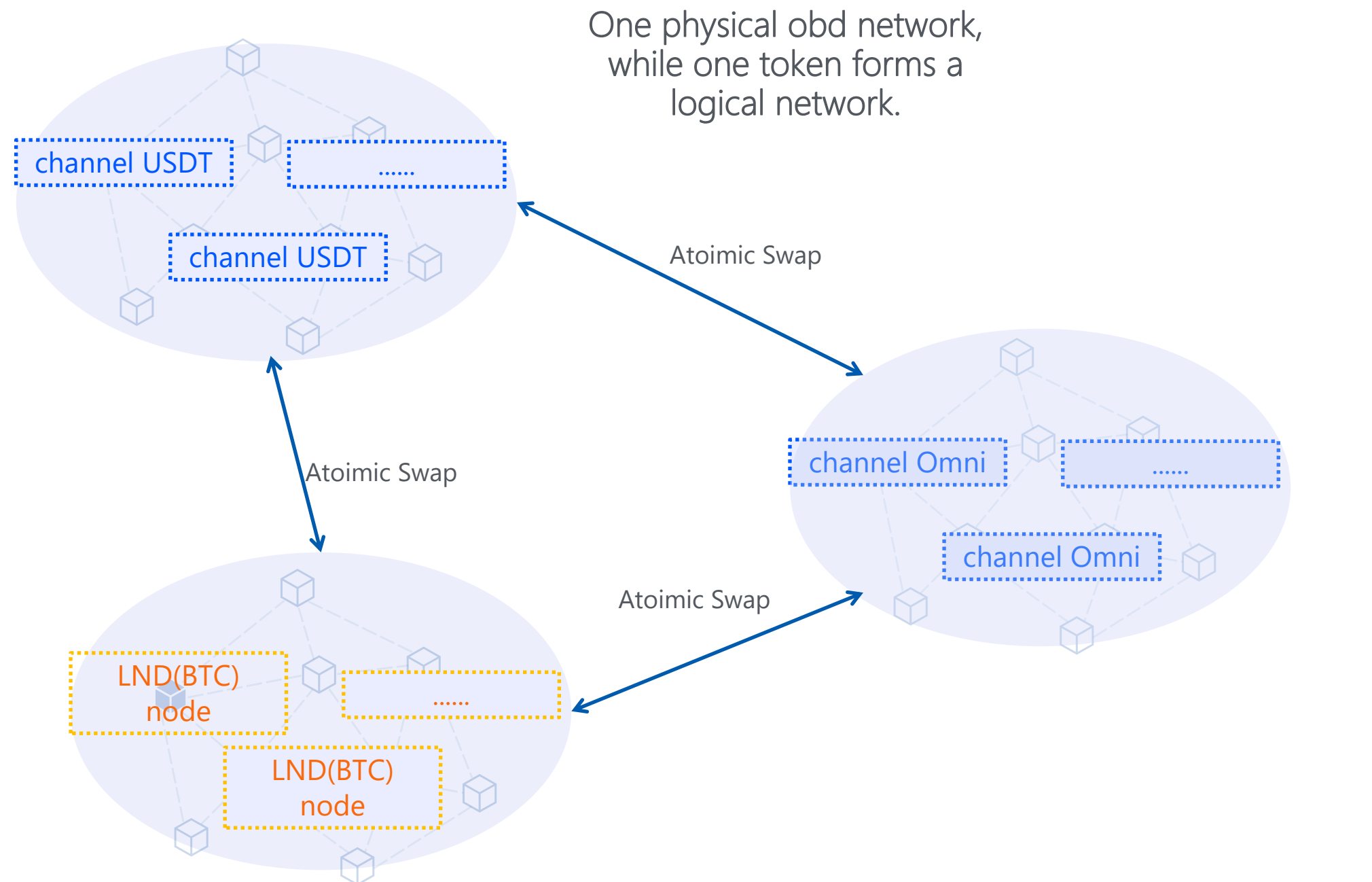
onchain protocol



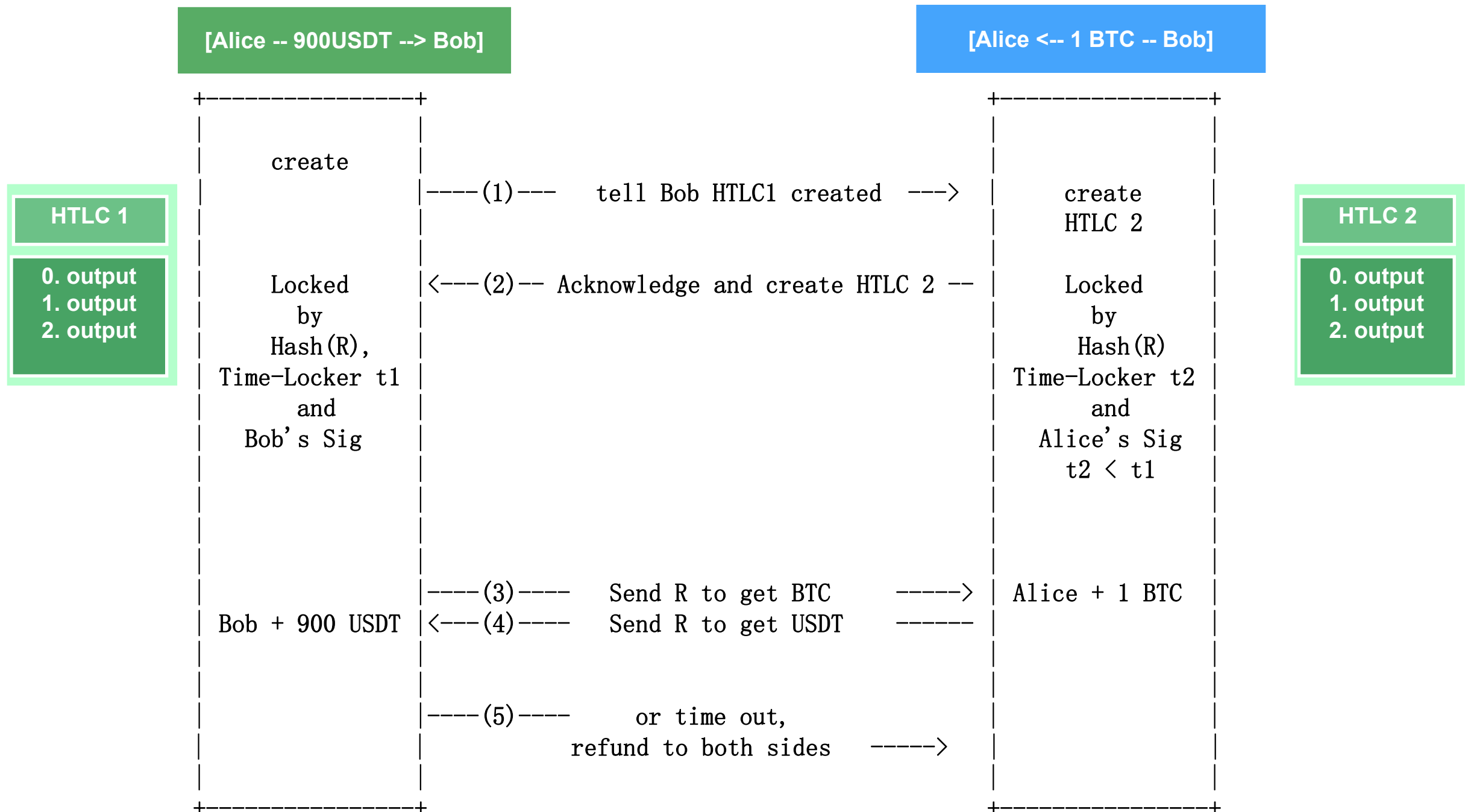
OmniBOLT #07: Construct transactions on OmniLayer

<https://github.com/omnilaboratory/OmniBOLT-spec#chapters>

Network and Atomic Swap



Atomic Swap



<https://github.com/omnilaboratory/OmniBOLT-spec/blob/master/OmniBOLT-05-Atomic-Swap-among-Channels.md>

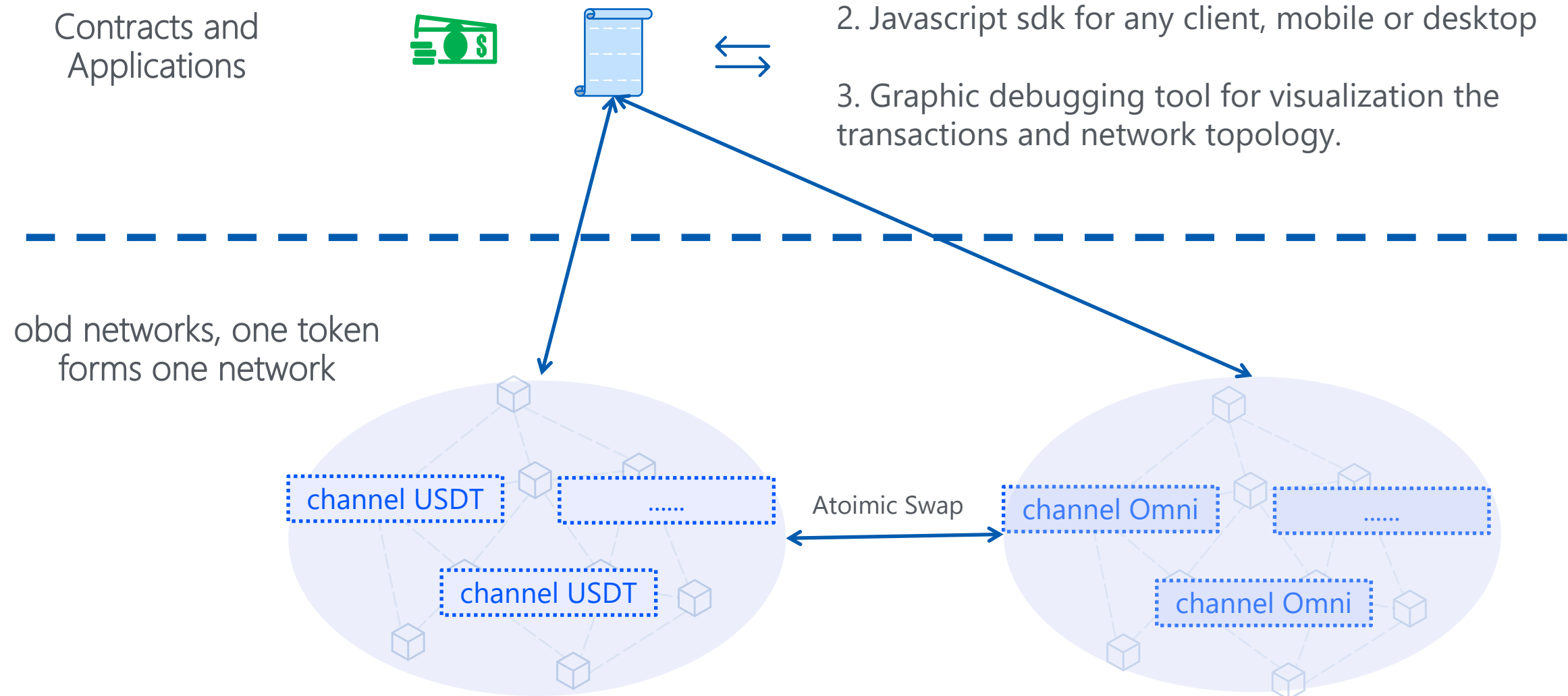
OBD Client SDK

Light Client Adaption

1. Websocket for web client

2. Javascript sdk for any client, mobile or desktop

3. Graphic debugging tool for visualization the transactions and network topology.



API documentation
<https://api.omnilab.online>

Tool kits
<https://github.com/omnilaboratory/DebuggingTool>

Light Client Adaption

1. Currently being released are websocket and javascript API package;
2. Function names are similar to LND clinet API, but not 100% the same, because OmniBOLT supports smart assets circulation and users shall be enabled to operates.
3. Learning curve will be smooth if one is already familiar with LND.
4. For new comer, the materials from BOLT/LND community are excellent to be learned from, and all knowledges apply to OmniBOLT.
5. General knowledge of OmniLayer, Bitcoin and lightning network can be found in there own communities.

Currently being Released JS and Websocket API

Function names
similar to LND client

Arguments similar to
LND client

Examples for App
developers

The screenshot shows the OpenChannel API documentation page at <https://api.omnilab.online/#openchannel>. The page is divided into three main sections, each indicated by a blue arrow from the text above:

- Function names similar to LND client:** Points to the left sidebar menu, which lists various API endpoints such as `OmniBOLT Daemon Websoc...`, `SignUp`, `Login`, `OpenChannel` (highlighted), `Simple Type -32 Protocol`, `Websocket Request: Message...`, `Websocket Response: Messag...`, `AcceptChannel`, `FundingBTC`, `BTCFundingCreated`, `BTCFundingSigned`, `FundingAsset`, `AssetFundingCreated`, `AssetFundingSigned`, `Commitment Transaction Cr...`, and `Revoke and Acknowledge C...`.
- Arguments similar to LND client:** Points to the main content area, which displays the `Simple Type -32 Protocol` and a table for `Websocket Request: Message Type -32`.
- Examples for App developers:** Points to the right sidebar, which shows code examples for `shell`, `javascript`, and `golang`. The `javascript` tab is active, showing a JSON request object.

The `Websocket Request: Message Type -32` table is as follows:

Parameter	default	placement	Description
funding pubkey	-----	data	public key of funder, who wish to deposit BTC and other tokens to the channel
recipient_peer_id	-----	body	peer id of the fundee.

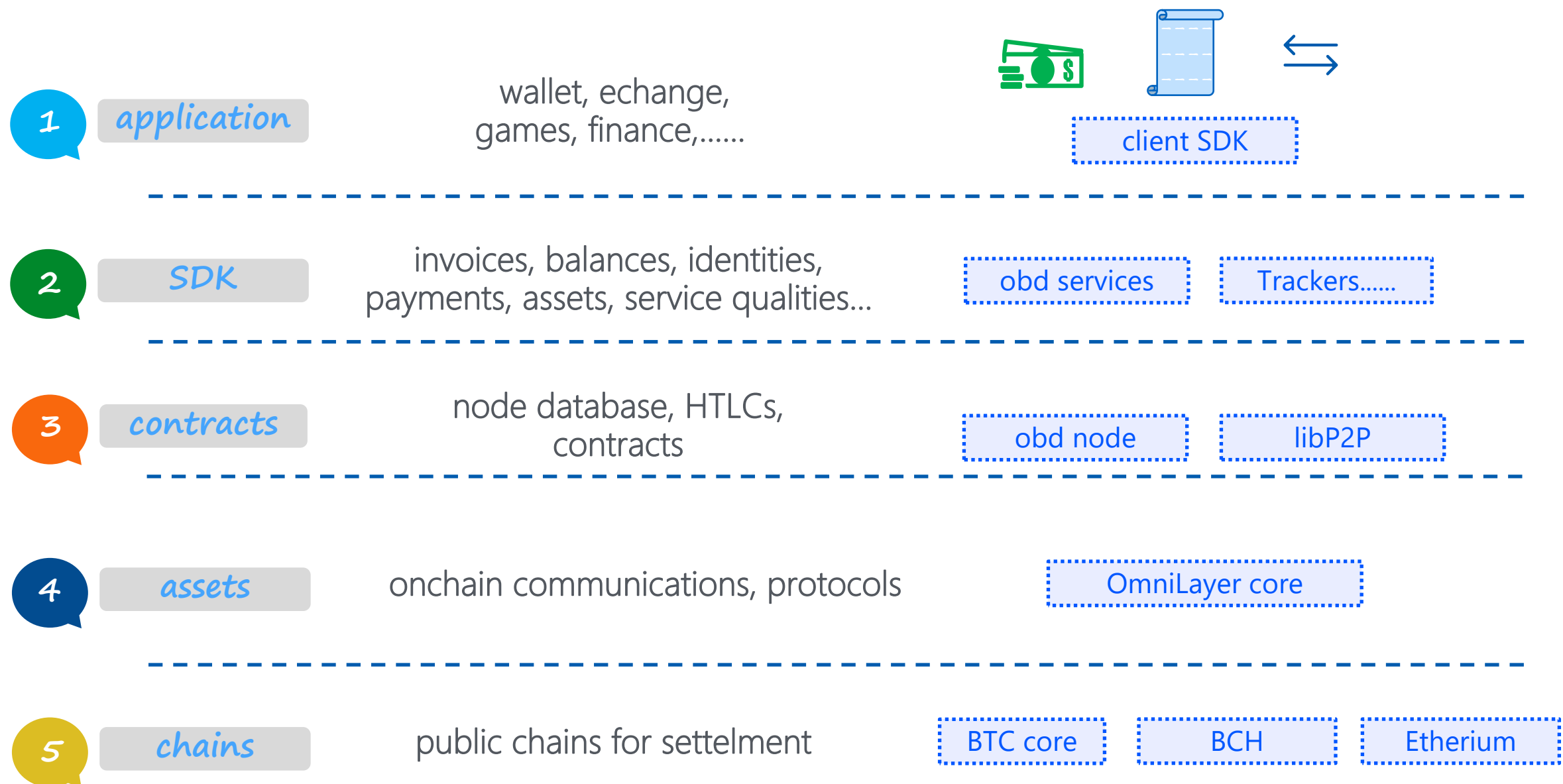
The `Request:` JSON example in the `javascript` tab is:

```
{
  "type": -32,
  "data": {
    "funding_pubkey": "021d475729c52f86df24...",
    "recipient_peer_id": "<user_id>"
  }
}
```

<https://api.omnilab.online>

Integration Stack

Integration Stack



Related Projects

<https://github.com/omnilaboratory/obd>

<https://github.com/omnilaboratory/DebuggingTool>

<https://github.com/OmniLayer/omniwallet>

<https://github.com/OmniLayer/omnicore>

<https://github.com/OmniLayer/OmniJ>

<https://github.com/OmniLayer/spec>

<https://github.com/lightningnetwork/lightning-rfc>

<https://github.com/lightningnetwork/lnd>

<https://github.com/omnilaboratory/OmniWalletMobile>



Thank you