

# DEX Contract Interface

TheCrocSwapDex contract is a thin low-level interface. It's designed with the principal of maximizing flexibility, efficiency, and future upgradeability on-chain, by delegating more complexity to off-chain clients.

The DEX contract ABI exposes five Solidity methods:

- swap()
  - - The sole "vanilla" Solidity method that accepts standard argument types. Because swaps are the most common operation this method runs purely in the "hot-path" without any calls to proxy sidecar contracts.
- userCmd()
  - - Single point of entry for all possible user commands. Accepts acallpath index
  - which maps to the specific proxy sidecar contract and an arbitrary byte string which is specifically structured for that specific sidecar.
- userCmdRouter()
  - - Same as userCmd()
  - but called by an external router contract on behalf of an end user. The user must have previously approved the router contract.
- userCmdRelayer()
  - - Same as userCmd()
  - but executes a command signed by a user off-chain with the EIP-712 standard.
- protocolCmd()
  - - Executes a protocol administrative command. This call can only be made the protocol authority contract.
- readSlot()
  - - Low-level view method for reading an arbitrary slot location in DEX contract storage. This only accepts the raw slot location, in most cases ordinary users are better off using the higher levelCrocQuery
  - contract to query information.
- 

[Previous](#) [Type Conventions](#) [Next](#) [userCmd](#) [Callpaths](#) Last updated 9 months ago On this page