## envoy http interface

all URL’s must have the appropriate port appended – typically port 8000.

| method, url | request body | response body |
| --- | --- | --- |
| get /  get /file | empty | html, css, js files |
| put /deploy | {  user: string, //publicKey,string?  term: string,  timestamp: int,  sig: string, // string?  sigAlgorithm: string,  from: string,  phloPrice: int,  phloLimit: int,  nonce: int  } | {  success: boolean,  message: string  } |
| put /block | {  blockHash:  header: {  parentsHashList: [ ],  postStateHash:  deploysHash:  timestamp:  version:  deployCount:  extraBytes:  },  body: {  postState: {  tuplespace:  bonds:  blockNumber:  },  deploys: {  deploy {  Deploy: // expand  },  cost: {  PCost: // expand  },  log: {  [ Event: ] // expand  },  errored:  }  extraBytes:  },  justifications: {  [ Justificaton: ] // expand  },  sender:  seqNum:  sig:  sigAlgorithm:  shardId  extraBytes:  } | {  success: boolean,  message: string  } |
| post /block | { } // or empty? | {  success: boolean,  message: string  } |
| put /show/block | {  hash: string  } | {  status: string,  blockInfo: {  blockHash: string,  blockSize: string,  blockNumber: int,  version: int,  deployCount: int,  tupleSpaceHash: string,  tupleSpaceDump: string,  timestamp: string,  faultTolerance: float,  mainParentHash: string,  parentsHashList: [ string ],  sender: string,  shardId: string  }  } |
| put /show/blocks | {  depth: int  } | [  {  blockHash: string,  blockSize: string,  blockNumber: int, //?  version: int, //?  deployCount: int,  tupleSpaceHash: string,  timestamp: string,  faultTolerance: float,  mainParentHash: string, //?  parentsHashList: string, //?  sender: string //?  } // zero or more  ] |
| put /show/chain | {  depth: int  } | // same as put /show/blocks |
| put /listen/data | {  depth: int,  name: {  Par: // expand  }  } | {  status:  blockResults: {  postBlockData: {  Par: // expand  }  },  blockResults: {  block: {  /show/chain: // expand  }  },  length:  } |
| put /listen/ continuation | {  depth: int,  name: {  [ Par ] // expand  }  } | {  status:  blockResults: {  postBlockContinuations: {  WaitingContinuationInfo:  },  block: {  BlockInfoWithoutTupleSpace:  }  },  length:  } |