Internet Financial EXchange (IFEX)

Searc

Our Proposals >

IFEX Protocol

The **Internet Financial EXchange (IFEX) protocol** facilitates the negotiation of financial transactions between internet-based financial endpoints.

Status: *Currently under development.* ***Development recommenced 2017-04-08.***

Problem Statement

Certain functionality required by modern, forward-looking financial systems is not presently available in open, legacy-free, adequately globalized protocols.

This functionality includes:

- **Negotiation**
  - Prescriptive quotations
  - Settlement and reversal / cancellation / contract exception terms
  - Exchange rate negotiation / hedging
  - Fee, tax and discount calculation / negotiation
  - Trust facilitation (eg. escrow, hedging via third party derivatives)
- **Arbitrary currency / asset support**
  - Conventional currencies, emerging currencies, currency-like commodities, commodities, even services
    - Significant latent interest in the emerging 'cloud', 'smart' or 'distributed' manufacturing sector, partly driven by the growth of 3D printing and partly driven by calls for increased efficiency within existing manufacturing supply chains
  - Multi-currency / asset transaction support
    - Reportedly widely used by agents in the global maritime shipping industry to hedge exchange rate risk across multiple conventional currencies
- **Arbitrary / redundant settlement path support**
  - Multiple concurrent settlement paths
    - Provide for increased availability through fault-tolerance in financial settlement
  - Support for in-band settlement (sometimes known as DVP)
  - Settlement latency calculation / negotiation
  - Multi-hop settlement path support
  - Arbitrary financial settlement topology support (including multi-hop, multi-party)
- **Hard to find features**
  - High precision decimal value support
  - Arbitrary communications topology support
    - Full support for *n:n* topologies such as those used in high availability or high frequency trading (HFT) systems
- **Security**
  - Integrated cryptography for message authentication and integrity
    - Enables reputation metrics, publishing contract excerpts for 'name and shame'

Given this situation, it makes sense to propose an open, legacy-free, adequately globalized, extensible protocol for internet-based financial exchange.

Work in Progress

General approach (2017):

- **Valid IFEX transaction states and transitions** can be viewed as a Mealy-type finite state machine (from George H. Mealy's *A Method for Synthesizing Sequential Circuits* (1955).).
- **Implement definition** in a formal / declarative environment of some type, ideally capable of theoretical validations. Currently evaluating:
    - **IETF ABNF** ("a modified version of Backus-Naur Form(BNF), called Augmented BNF (ABNF), has been popular among manyInternet specifications. It balances compactness and simplicity withreasonable representational power.")
        - **50% done.** Here are the 2012 work's IFEX transaction state definitions (missing state transitions):

```
txstate    = prestate / livestate / poststate
prestate   = 'INITIAL'
livestate  = 'PENDING' / 'SETTLED'
poststate  = 'SUCCESS' / 'FAILURE' / 'PARTIAL'
```

    - **Isabelle** ("generic proof assistant [that] allows mathematical formulas to be expressed in a formal language and provides tools for proving those formulas in a logical calculus")
        - **Rather mathematical and opaque.** Apparently we need to define a set, then a series of transformation functions.
    - **ohm** ("a parser generator consisting of a library and a domain-specific language. You can use it to parse custom file formats or quickly build parsers, interpreters, and compilers for programming languages. The Ohm language is based on parsing expression grammars (PEGs), which are a formal way of describing syntax, similar to regular expressions and context-free grammars.")
        - **50% done.** Here are the IFEX transaction state definitions (missing state transitions):

```
IFEXTransactionStates {

        InitialState = "INITIAL"
        PendingState = "PENDING"
        SettledState = "SETTLED"
        SuccessState = "SUCCESS"
        FailureState = "FAILURE"
        PartialState = "PARTIAL"

        PreStates = InitialState
        LiveStates = PendingState | SettledState
        PostStates = SuccessState | FailureState | PartialState

        IFEXTransactionState = PreStates | LiveStates | PostStates

}
```

    - **OMeta** ("a new object-oriented language for pattern matching. It is based on a variant of Parsing Expression Grammars (PEGs) which we have extended to handle arbitrary data types. OMeta's general-purpose pattern matching facilities provide a natural and convenient way for programmers to implement tokenizers, parsers, visitors, and tree transformers, all of which can be extended in interesting ways using familiar object-oriented mechanisms.")
        - **Very similar to ohm.**
- **Generate model implementations** from this definition, ideally capable of concurrent interoperability testing, potentially using existing databases or state machine libraries such as:
    - **sqlite**: state transitions can be limited through triggers.
    - **ruby**: state_machine
- **Develop an external, pluggable-risk-or-cost-model-capable decision making system utilizing the formally defined transactions.** Should be able to optimize for any combination of factors including but not limited to:
    - cost
    - specific risks
    - overall risk / reliability
    - liquidity of various assets
    - complex scheduling

  This can probably utilize existing mathematical and operations research algorithms to meet common requirements (eg. system scheduling) such as:

- generalized assignment problem
- combinatorial optimization
- simulated Poisson distribution and Bartlett's theorem

General approach (2012): A stateful message-oriented protocol, based upon JSON, that normalizes financial transaction identification and state transitions (initial, pending, settled, success, failure, partial) across disparate settlement systems. Supports arbitrary topologies and transports, aiming for significant extensibility, and keeping the overly specific stuff (used only in some financial messaging scenarios) out of the core specification.

Settlement paths over arbitrary settlement networks can be compared by individual IFEX Nodes based upon hard data regarding the settlement path properties that is supplied by the settlement provider(s) in `QUO` (quotation) messages. Such messages may be sent in response to `RFQ` (request for quotation) messages.

An IFEX Node might then send execution (`EXE`) messages to initiate the settlement of a financial transaction, tracking its progress through the normalized state transition mentioned above.

In computer science terms, the protocol hopes to implement something akin to the Paxos algorithm or or Chandra–Toueg consensus algorithm, optimized for a financial transaction use case.

In this sense, the IFEX Protocol hopes to create a fair and open market for financial services with the capacity for real time, intelligent routing based upon real route (settlement path) characteristics - fostering interoperability and removing barriers to communication, something like IP did for networking, or SMTP did for electronic mail. It also serves to support real time redundancy and failover for high availability financial services.

Status:

- 2017-04-28: Development recommences.
- 2012-11-27: **Draft proposal** on GitHub.
- 2012-04-11: (OLD) **Partial draft proposal from Payward, Inc.**
  - **Draft JSON Schema for the Proposal**
- Open Issues List

Related Projects, Systems and Information

The following references provide information around conventional, emerging and proposed payment protocols and systems that are of potential relevance to IFEX development.

- **Bitcoin** (Wikipedia)
  - **General**: Bitcoin is the first successful cryptographic currency implementation with global reach and relative stability.
  - **Feature set**: Blockchain-based cryptographic currency (X-ISO4217-A3: `XBTC`) and settlement system. Historical attempts to expand scope have largely failed.
  - **Limitations**
    - **Throughput**: Limited
    - **Fees**: Required to effect a relative guarantee of timely processing
    - **Settlement latency**: Not possible to accurately pre-calculate
  - **Quirks**
    - Full network history is stored within the shared database of the blockchain.
    - Network splits are possible, and nominally countered through waiting for multiple confirmations before treating a given transaction as 'confirmed'.
    - A large proportion of network mining capacity, central to the operation of the system, is currently centralized in China due to the availability of cheap hardware and electricity.
    - Ease of mining may be significantly enhanced through the development of alternative hardware.
  - **Abandoned scope / features**
    - ***Bitcoin Payment Messages***, Gavin Andressen. Proposed circa December 2012.
      A somewhat controversial effort by the Bitcoin development community to resolve some of Bitcoin's quirks.
      "This document proposes protocol buffer-based formats for a simple payment protocol between

a customer's bitcoin client software and a merchant. Separate documents will propose an extension to the Bitcoin URI syntax and new MIME types to support them."

- Status: The proposal was criticized for its complexity and ill-considered scope and failed to gain momentum.

- **Ethereum** (Wikipedia)
    - **General**: Ethereum was the second major cryptographic currency implementation to achieve significant adoption.
    - **Feature set**: Ethereum provides similar features to Bitcoin in addition to a decentralized Turing-complete virtual machine, allowing "smart contracts" to be hosted on the platform which could eventually enable "decentralized autonomous organizations".
    - **Limitations**
        - Throughput: Limited (~25 transactions per second)
        - Volatility: Ethereum is yet to reach Bitcoin's level of stability.
    - **Quirks**
        - There have been significant security incidents regarding unnoticed code paths in smart contracts.

- **Ripple** (Wikipedia)
    - **General**: Ripple is a protocol for the connection between various ripple exchanges. In late 2012, it was announced that the previous ripple-project.org was being commercially backed by OpenCoin, Inc. in San Francisco and development moved to the commercial domain ripple.com.
    - **Feature set**: Originally decentralized with explicit support for arbitrary nodes to issue their own assets, commercialization has apparently shifted focus toward the built-in Ripple asset which was apparently originally intended to function as a network operations / transaction fee mechanism.
    - **Limitations**
        - Limited user base.
    - **Quirks**
        - Historically broad feature set.
        - Inter-node trust model is apparently less decentralized than Bitcoin and similar blockchain-based systems, creating nominal opportunities for abuse.

- **W3C Web Payments** (W3C / Web Payments / Payswarm)
    - **General**: A World Wide Web Consortium (W3C) effort focusing on removing barriers to and enhancing the user experience of web based commerce.
    - **Feature set**: Never completed.
    - **Limitations**
        - Web-oriented, B2C, point-to-point payment focus.
    - **Quirks**
        - Incomplete.

- **Logistics, Operations Research (OR) and Supply Chain Management (SCM)**
    - **General**: The requirements to move and consume physical goods is ancient.
    - **Feature set**: Logistics deals with *"the flow* [and processing, and storage/warehousing] *of things between the point of origin and the point of consumption* [and related information] *in order to meet requirements of customers or corporations*". Supply chains are, roughly speaking, a subcategory of logistics systems focused on moving goods or components for further processing (production / value add) or consumption. Note, however, that even if a supply chain or logistics system is typically consumption-oriented, they are usually at least nominally bidirectional (and not necessarily utilizing bidirectionally symmetric paths or entities) due to the occasional need to return or repair faulty parts or products. Operations Research (OR), which grew out of military logistics, focuses on the use of mathematical models, statistics and algorithms to aid decision-making in complex real-world systems, typically with the goal of optimizing performance.
    - **Limitations**
        - Conventional systems traditionally rely upon contracts based upon extensive, somewhat subjective (ie. non machine-parseable) legalese. Therefore, in the event of real or perceived breach of contract, resolution is typically made through professional trade arbitration or government courts of law, which can be slow and expensive.
    - **Quirks**
        - Frequent outsourcing of transportation, warehousing and production/value-add components
        - International logistics systems incur overheads due to currency exchange and related market risk
        - Many types of goods require special transportation, handling and storage considerations

(security, spoilage, etc.)

- Broad range of potential customs, contractual and payment scenarios vary by jurisdiction and type of goods, and are typically defined through a combination of legal contracts, regulation, and case law

- **Other references**
    - ***Beyond IP Transactions: towards a payment protocol***, sipa, ~late 2011.
      "What current addresses are, is a reference to a public key. The way they are used is as a template for a transaction. If you do not need complex transactions, this suffices indeed, given that all other negotiation about the payment occurs out-of-band already (e.g., a webshop interface that after clicking 'pay' gives you a freshly generated bitcoin address and stores it so it can track your payment). What I want to do is to standardize part of that out-of-band communication inside a protocol. [...] Summarized: addresses are a limited method for defining payments, and as soon as you move to a protocol instead of a static template, a lot of possibilities open up."

    - ***Homomorphic Payment Addresses and the Pay-to-Contract Protocol***, Ilja Gerhardt & Timo Hanke, 2012-12-13.
      "We propose an electronic payment protocol for typical customer-merchant relations which does not require a trusted (signed) payment descriptor to be sent from the merchant to the customer. [...] The protocol is specifically designed with bitcoin in mind as the underlying payment system."

Subpages (2): 2012-04-11 Partial Draft Open Issues List

Comments

Sign in | Report Abuse | Print Page | Powered By **Google Sites**