Hyperledger Quilt

⚠ Hyperledger Quilt was moved to dormant status at the request of the maintainers on 17 JUN 2021. You are welcome to use and contribute to this code, although the maintainers may or may not be responsive to any questions you have. You would also be welcome to help reactivate this project if you are interested in continuing development of this code.

Project	HYPERLEDGER QUILT
Status	DORMANT
CII Badge	
Description	An interoperability solution for any fungible assets - fiat, crypto, DLT, in-game tokens, or any other type of asset tracked on a ledger.

Hyperledger Quilt is a Java implementation of Interledger, a suite of open protocols and standards that allows payments interoperability across any currency - fiat or crypto. Interledger standards and specifications are being defined by the open source community at Interledger.org and at various community groups in the W3C, including Web Monetization and Web Payments.

News and PR

- **Announcing Quilt v1.0:** (https://www.hyperledger.org/blog/2019/11/01/announcing-hyperledger-quilt-v1-0-interledger-for-the-java-platform)

Key Characteristics

By implementing the Interledger protocol, Quilt provides:

- **Foundations**: A set of protocols and primitives for enabling value interoperability across any fungible asset system, including crypto, fiat, in-game, and other token currencies.
- **Identity**: A standardized, globally unique identifier for payment accounts called Payment Pointers.
- **Liquidity**: A peer-to-peer packetized payments protocol specified under ILPv4.
- **Sending & Receiving**: A Java implementation of STREAM, which is a packetized payment protocol inspired by the QUIC Internet Transport Protocol. STREAM operates on top of core Interledger protocols by allowing a sender and receiver to coordinate a payment over any arbitrary Interledger topology.
- **Extensibility**: Quilt provides a framework for designing higher layer use-cases on Interledger to enable payment systems interoperability.

Repositories

- **Github**: hyperledger/quilt

Documentation

- **Java Project Documentation**
  - Project overview
  - Github Issues
  - Javadoc Links
  - Java Examples
- **Project Artifacts**

    - Releases & Changelogs
    - Current Milestones
    - Maven Central Coordinates

- **Interledger Protocols**
    - Interledger Addresses: a hierarchical identifier for Interledger network nodes that enables efficient routing of payment packets.
    - ILPv4: The lowest-layer Interledger protocol that enables packets to multi-hop from peer to peer across the Interledger.
    - Payment Pointers: a standardized identifier for end-user payment accounts.
    - ILP-over-HTTP: An HTTP protocol that allows two peers to transmit ILPv4 packets to each other using HTTP.
    - SPSP: Simple Payment Setup Protocol, used to allow a sender and receiver to negotiate the security parameters of an Interledger payment.
    - STREAM: A protocol for reliably sending money and data over ILPv4.

Security

The Quilt project tries to maintain best-practices when it comes to the security of the library. To this end, we employ the following automated scans:

- Github Automated Dependency Scans (announcement & docs)
- LGTM (acquired by Github in 2019): LGTM's security analysis is powered by findings from a dedicated team of security researchers, and by contributions from security teams at a number of top tech companies.

Interledger TestNet

The Quilt project implements Interledger sender and receiver primitives, but requires a server infrastructure in order to operate. The current Interledger Testnet is operated by Xpring, and can be accessed at https://xpring.io.

Community

Mailing Lists

- Quilt Maling List
- W3C Interledger Mailing List

Chat/Communications (for questions and ephemeral discussions)

- Hyperledger Rocket Chat: #quilt
- Interledger Slack.
- Interledger Community Forum.
- Other ways to get involved in the Interledger community.

Community Meetings

We have open community calls **every other Wednesday at 5pm UTC** to discuss the latest in Interledger Java projects, including Interledger spec development, the Quilt implementation, and to answer any questions people have. Agendas are sent out via the mailing lists and anyone can suggest an agenda item by adding to the topic created for that purpose in the Interledger forum (or by reaching out on any channel in the community).

Previous meeting recordings can be found here (older archives here).

Compatible Software

At its core, Interledger is a protocol that can be used in a nearly infinite number of use-cases. The protocol has been designed using a layered design approach, similar in fashion to the Internet suite of protocols. To this end, Quilt should be interoperable with any other Interledger-compatible implementation. This following are known implementations of Interledger in various communities and programming languages:

- Interledger-rs: An implementation of the Interledger suite of protocols in Rust.
- Interledger-js: An implementation of the Interledger suite of protocols in Javascript.
- Rafki: A modular approach to Interledger components, written in Javascript (blog announcement here).

History

- Proposed by Takahiro Inaba – NTT DATA, Adrian Hope-Bailie – Ripple, and Isaac Arruebarrena – Everis, an NTT DATA Company.
- Approved by the TSC on 2017-10-12.
- Approved maintainer change to David Fuelling on 2019-09-19.

Recent space activity

David Boswell

Hyperledger Quilt updated Jan 10, 2022 • view change

Ry Jones

Hyperledger Quilt updated Jan 10, 2022 • view change

Arnaud J Le Hors

Hyperledger Quilt updated Jun 23, 2021 • view change

Ry Jones

2019-11-06 Meeting notes updated Jun 15, 2021 • view change

David Fuelling

2020-06-02 Meeting notes updated Jun 03, 2020 • view change

Space contributors

- David Boswell (57 days ago)
- Ry Jones (57 days ago)
- Arnaud J Le Hors (259 days ago)
- David Fuelling (643 days ago)
- Sudheesh Singanamalla (840 days ago)
- ...