プロジェクト・ステラ：
DLTと決済インフラの未来の探究
Project Stella: Search for future of DLT and financial market infrastructure

FIN/SUM 2019

2019年9月3日 16:00～17:30

日本銀行 決済機構局

参事役　岸　道信

プロジェクト・ステラとは

- プロジェクト・ステラ（ Project Stella ）は、2016年12月に日本銀行と欧州中央銀行が立ち上げた、分散型台帳技術（DLT）の金融市場インフラへの応用可能性に関する共同調査プロジェクト。
  - なお、本プロジェクトは、中央銀行が運営する決済システムを含む既存の仕組みを置き換えたり、補完することを意図したものではない。また、本調査は法律や規制面には立ち入っていない。

プロジェクト・ステラとは

Ph. 1

- From December 2016 to September 2017
- Replicates liquidity saving mechanisms
- Presents quantitative results of tests on both efficiency and safety aspects (the first of its kind, at the time of publication)

Ph. 2

- Until March 2018
- Applicability of DLT to securities delivery versus payment

Ph. 3

- Until June 2019
- Synchronised cross-border payments

実験に用いた基盤

Ph. 1

- Hyperledger Fabric v0.6.1

Ph. 2

- Corda release-V2
- Elements v2.14.1.1
- Hyperledger Fabric v1.1.0-alpha

Ph. 3

- Hyperledger Fabric v1.2.1

RTGSの一部機能の実現可能性に関する調査

Stella Phase 1

フェーズ 1の目的

- 「銀行間資金決済システムの擬似環境」を用いて、技術の有効性や課題を評価。

- スマートコントラクト（一定の条件が満たされた場合に自動的に処理を執行するDLT台帳上のプログラム実行機能）を用いて、日銀ネットやTARGET2（ECBが運営するReal Time Gross Settlementシステム）の「流動性節約機能」を組み込むことができるかを確認。

フェーズ 1の評価項目など

- 主な評価項目：

評価項目	評価内容
効率性 ⇒ 処理性能（パフォーマンス）	検証ノード数や取引指図の数を増加させても、速やかに処理を実行できるか。
安全性 ⇒ 可用性（アベイラビリティ）	検証ノードの障害発生時や障害復旧後に、迅速に業務を再開したり、システムの継続性を確保できるか。

- 実験環境：実験当初はスタンドアロン端末、その後はクラウド環境上で実施。

フェーズ1 結果のまとめ

処理性能からみた効率性の評価

RTGSサービスのうち実験の対象となった部分については、DLTによって、現行の大口資金決済システムが求めるパフォーマンスとほぼ同等の水準を満たす可能性。

可用性からみた安全性の評価

検証ノードにおける障害や、フォーマットの誤った取引指図といった課題に対処する可能性を持ち合わせていることを確認。

結 論

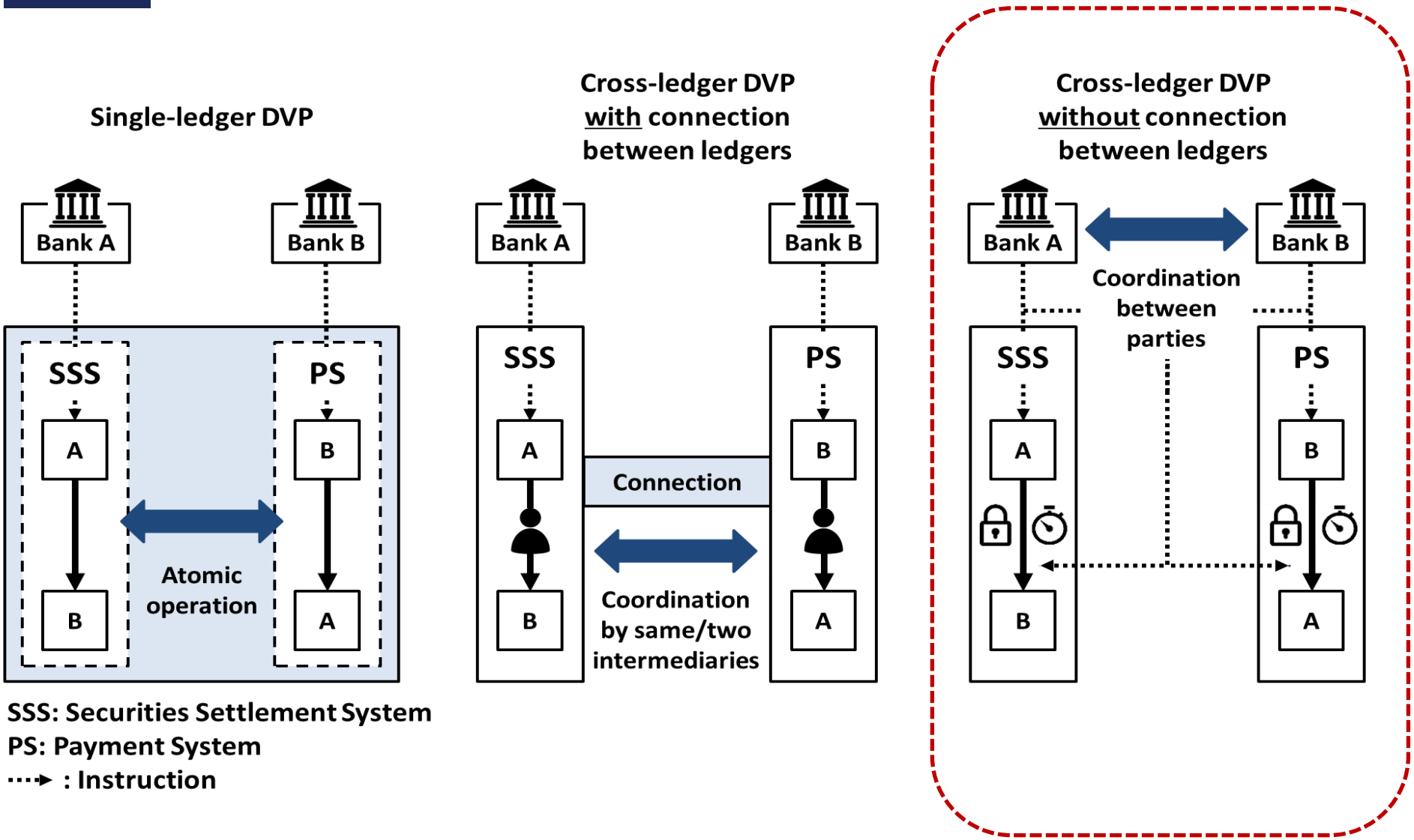
DLTを資金決済システムに応用する可能性について、前向きに捉えるに値する検証結果が示された。

もっとも、今次調査の検証結果は、あくまで実験環境で行われたものであり、本番環境におけるDLTの実用可能性について評価するものではない。

Delivery vs. Paymentの実現可能性について

Stella Phase 2

DLT環境下でのDvP実現

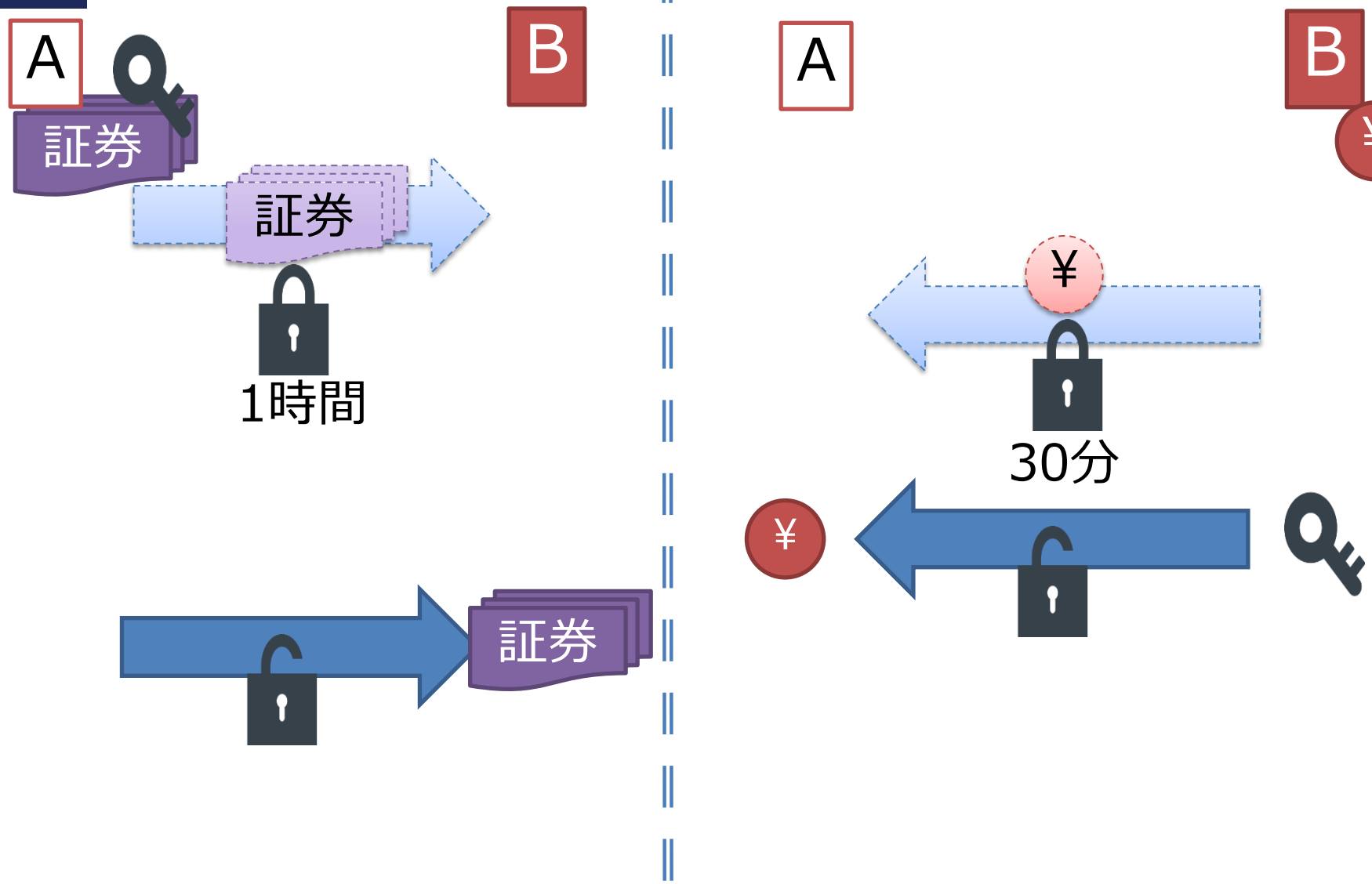


Hashed Timelock Contract (HTLC)

HTLC（ハッシュ・タイムロック・コントラクト）とは、①暗号学的ハッシュ関数と、②タイムアウト機能を組み合わせた、取引の同期に資するプログラム。

暗号学的ハッシュ関数（H）では、Y=H(X)の関係が成立するとき、Xを「原像(Preimage)」、Yを「ハッシュ値」という。原像からハッシュ値はただちに計算できるが、ハッシュ値から原像を推定することは困難。

Hashed Timelock Contract (HTLC)



分析結果

	単一台帳方式	ネットワーク間の接続のない 複数台帳方式 （複数台帳HTLC方式）
インフラ全体のデザイン	1つのネットワーク上で 様々な資産のやり取りが 行われる	複数のネットワークが 互いに一切の接続なく併存
メリット	流動性の利用効率 処理時間	柔軟性
課題	柔軟性 スケーラビリティ 頑健性	流動性の利用効率 処理時間

クロスボーダー取引における支払の同期化

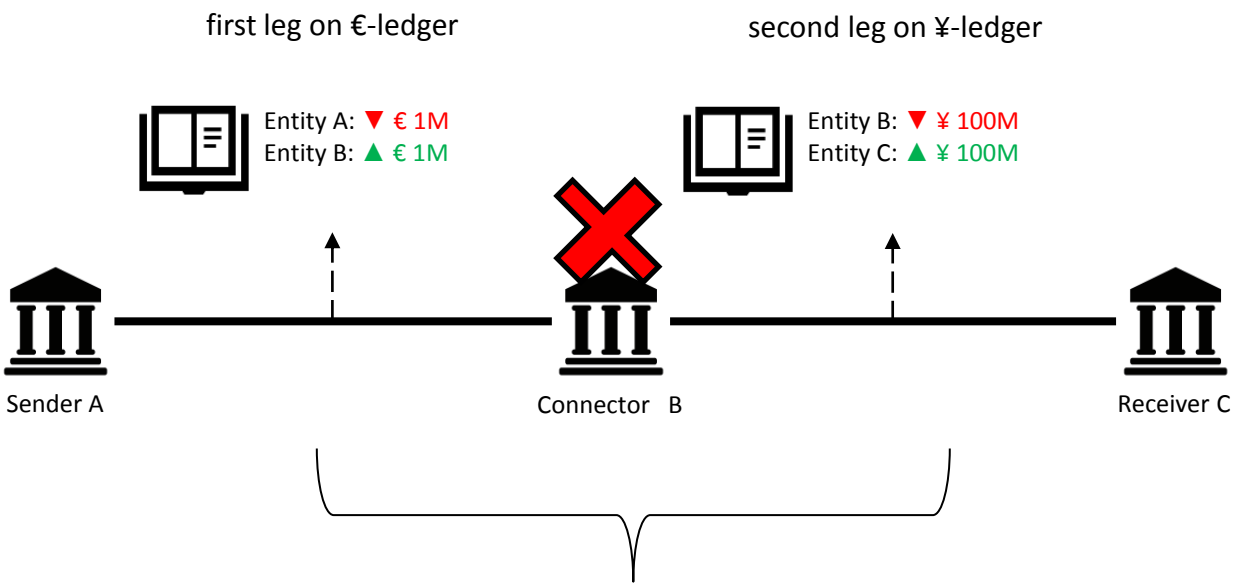
Stella Phase 3

フェーズ3の問題意識

- クロスボーダー送金（異なる通貨圏への送金）は、複数の法域を跨いだ様々な機関が関与するため、国内送金と比べて時間とコストがかかっている。
- こうした課題につき、現在、**効率性**の改善を企図した取組みが行われている。
- 他方、送金の**安全性**の観点からは、複数の台帳を経由することに伴う課題が残る。

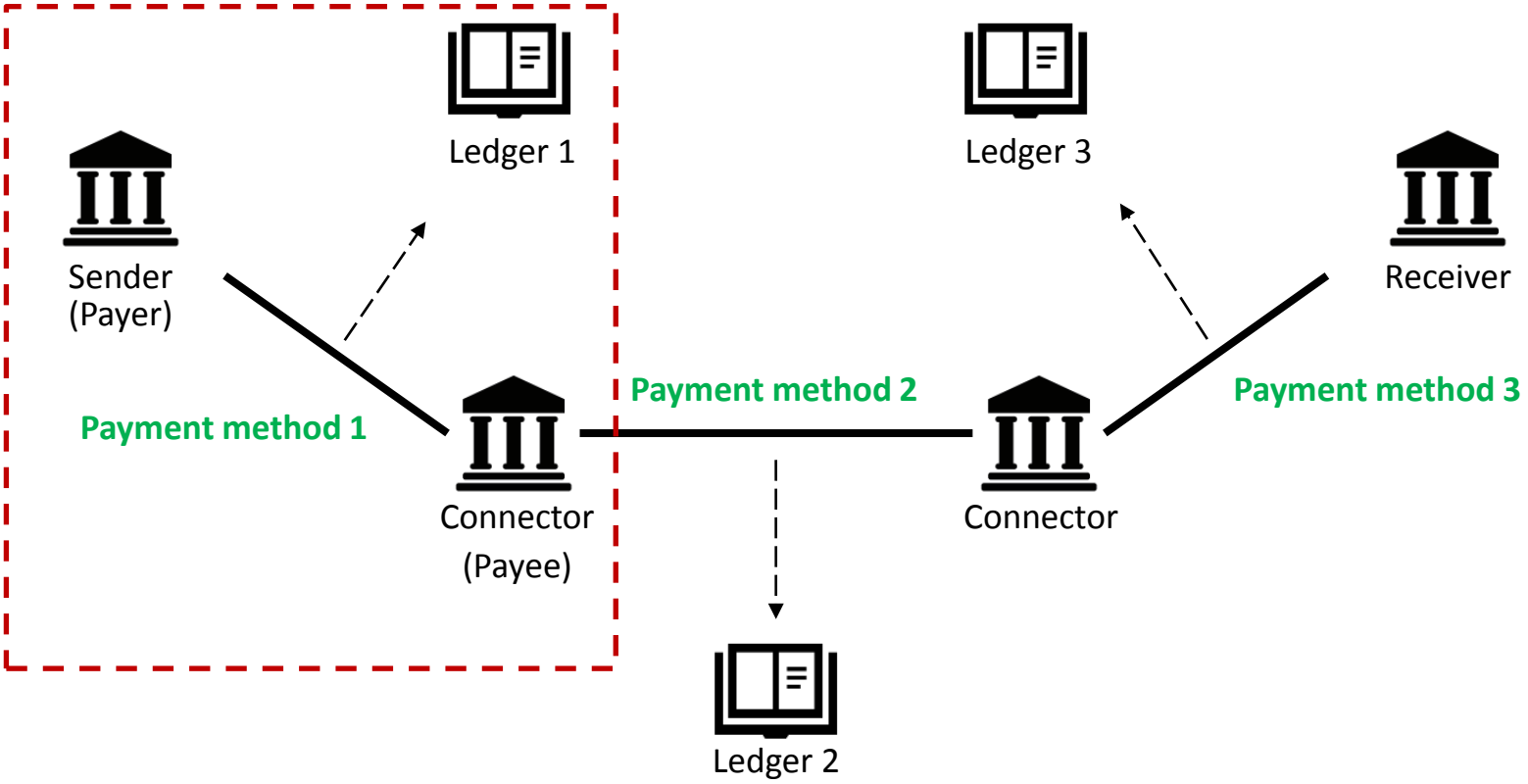
フェーズ3の問題意識

- 送金者Aが受領者Cに、中継銀行Bを経由して送金する場合、Aが中継銀行Bに送金したにもかかわらず、中継銀行BがCに送金する前に破綻するリスク（信用リスク）が存在。
  - 複数台帳間の送金（下図のＡ・Ｂ間の送金とＢ・Ｃ間の送金）が、ハッシュ関数を利用して同期されて行われれば、リスクは低減され得る。



フェーズ3で考えるユースケース

- 本報告書でのクロスボーダー送金は、**送金者**から**受領者**への送金が、**中継者**を経由して行われるケースを想定。
  —— 点線内のブロックは、支払人（一つの台帳での支払に注目する際には、送金者をこのように呼び換える）、受取人（中継者を呼び換える）と台帳で構成される。



フェーズ3のアプローチ

- フェーズ2対比、検討範囲を拡げた。

	Phase 2	Phase 3
Synchronisation of ledgers	HTLC	**Both HTLC and methods not dependent upon HTLC are considered.**
Type of ledger for synchronisation	Between DLT ledgers	**（i） Between a centralised ledger and a DLT ledger, (ii) between DLT ledgers, and also (iii) between centralised ledgers.** **Our experiments confirmed that synchronisation is achieved in all the three patterns.**

安全な支払方法(payment method)の例：HTLCを用いたもの 1ブロック内でのやりとり

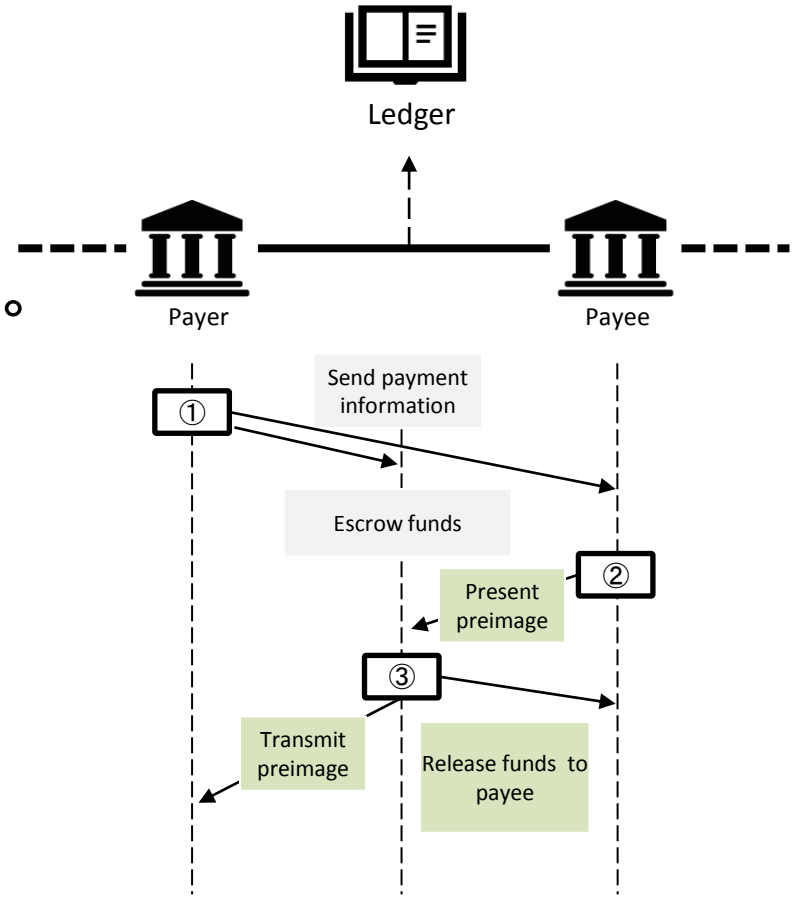
Prepare

① 支払人は台帳で資金を拘束状態にするよう、台帳に依頼。台帳は支払人の信用リスクから隔離した状態で資金を固定する。

Execute

② 受取人が、タイムアウト前に、台帳に対して原像を提示する（事前に設定された条件が満たされる）。

③ 台帳が資金を受取人に送る。時間内に満たされなければ、当該資金は支払人に戻る。

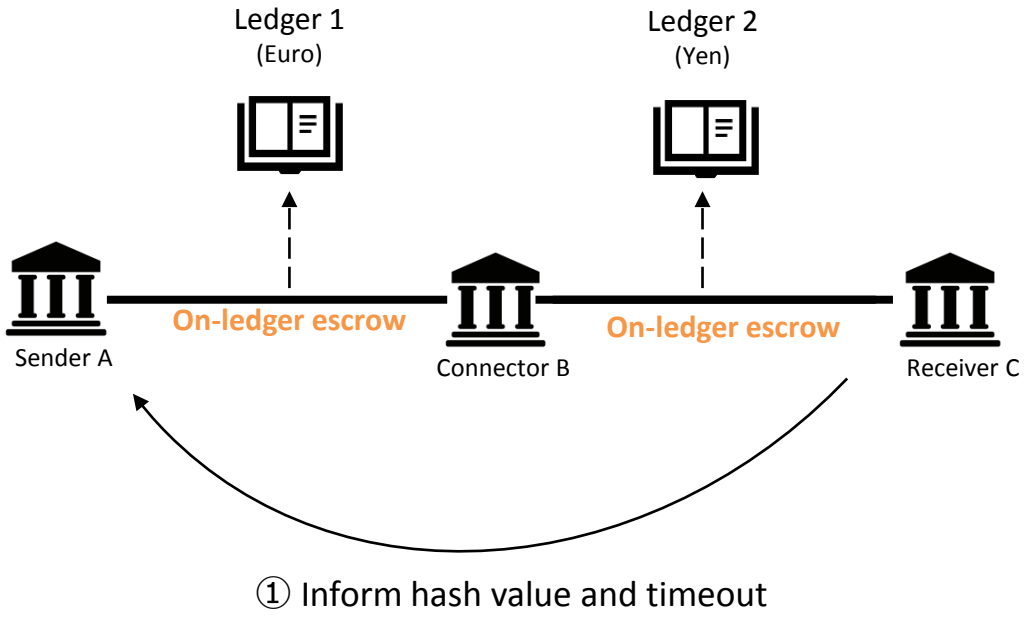


フェーズ3では、複数台帳間で同期をとった支払を行うことを目指して制定された規約（プロトコル、例えばInterledger Protocol(ILP)）に沿って、クロスボーダーでの支払が行われるケースを分析。

安全な支払方法の例：HTLCを用いたもの クロスボーダー送金

Prepare

① 受領者Cが原像を生成。原像とハッシュ関数を用いてハッシュ値を計算。そのハッシュ値やCへの支払時限T（例えば本日15時）を送金者Aに連絡。

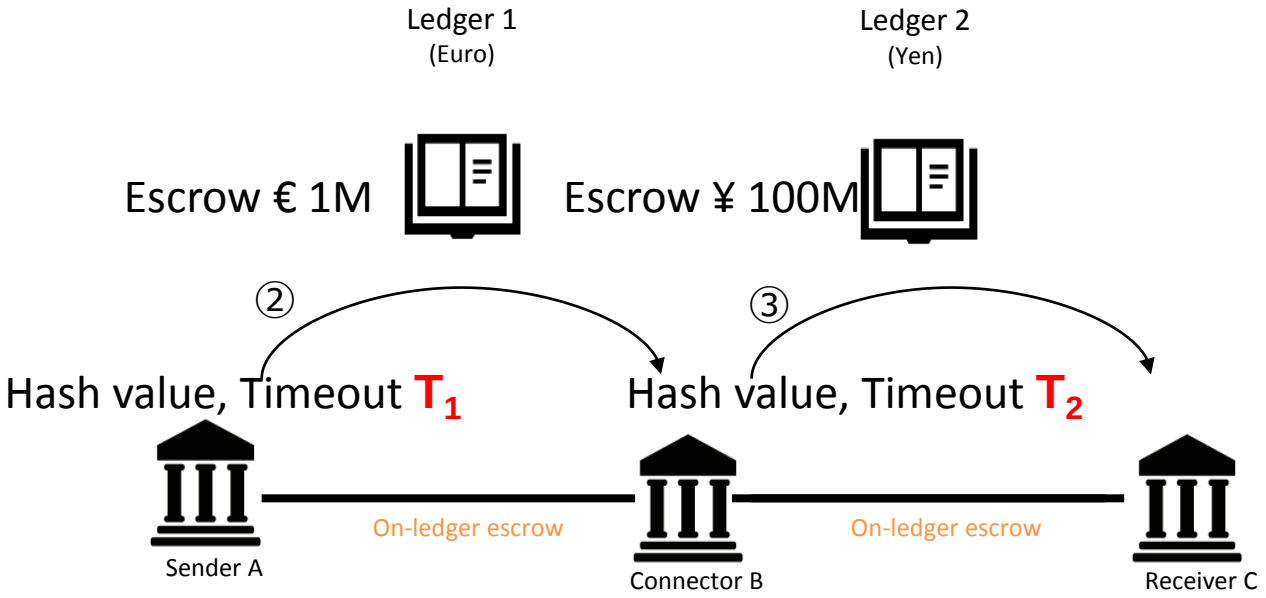


安全な支払方法の例：HTLCを用いたもの クロスボーダー送金

Prepare

② Aは台帳1に、Aの持っている100万ユーロを固定してもらう。また、Aは中継銀行Bと台帳1にハッシュ値を伝えるとともに、ある「時限T_1」（例えば本日17時）までにBがハッシュ値と整合的な原像を提示すればBに100万ユーロを支払うよう台帳1に指示。

③ Bは台帳2に、Bの持っている1億円を「時限Ｔ」までに固定してもらう。また、BはCと台帳2にハッシュ値を伝えるとともに、ある「時限T_2（T_1より早い時間）」（例えば本日13時）までにCがハッシュ値と整合的な原像を提示すればCに1億円を支払うよう台帳2に指示。

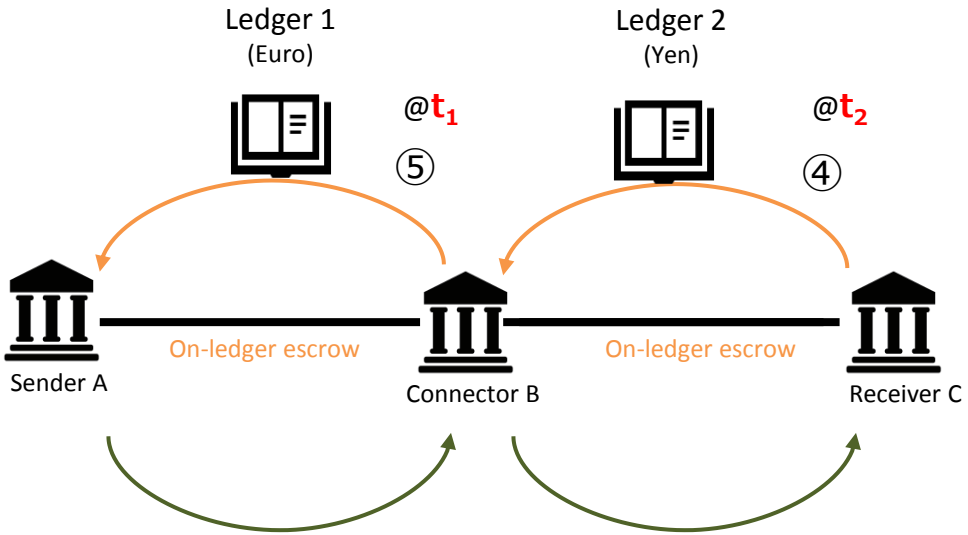


安全な支払方法の例：HTLCを用いたもの クロスボーダー送金

Execute

④ CがT$_2$より前（時刻t$_2$）に台帳2に原像を提示し、Cに資金が渡る（④′）。

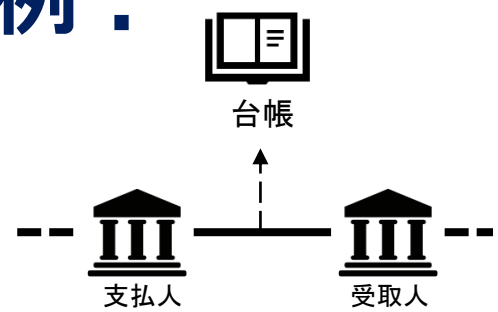
⑤ Bが T$_1$より前（時刻t$_1$）に台帳１に原像を提示し、Bに資金が渡る（⑤′）。



「同期された安全な支払」の特徴

- t2 ～ t1 の時間ラグはあるが、台帳上で資金が支払人の信用リスクから隔離された状態で固定されているため、送金は安全に実行される。

安全ではない支払方法の例：トラストライン



安全ではない支払方法の例：Trustline

- Set-up (Trustline作成)
  - 決済を行うことなくつけ払い可能な金額の上限を、支払人、受取人間で合意。支払人の資金は固定されない。
- State update(支払中)
  - Prepare（準備）：支払人は受取人に「受取人が時限内に原像を提示すれば支払う（つけ払いを行う）」と約束する。
  - Execute（実行）：原像を受取人がタイムアウト前に提示した時、支払人はつけ払いを行うことが想定されている。
- Settlement (決済)
  - 決済を行うことなく支払可能な金額の上限を超える前に、支払人は資金移動を台帳に依頼する。台帳は支払人から受取人に資金を移動する。

「安全ではない支払」の特徴

・　支払人の資金が固定されていないため、つけ払いの段階（決済前）に支払人が破たんすると、受取人は資金を受け取れない。

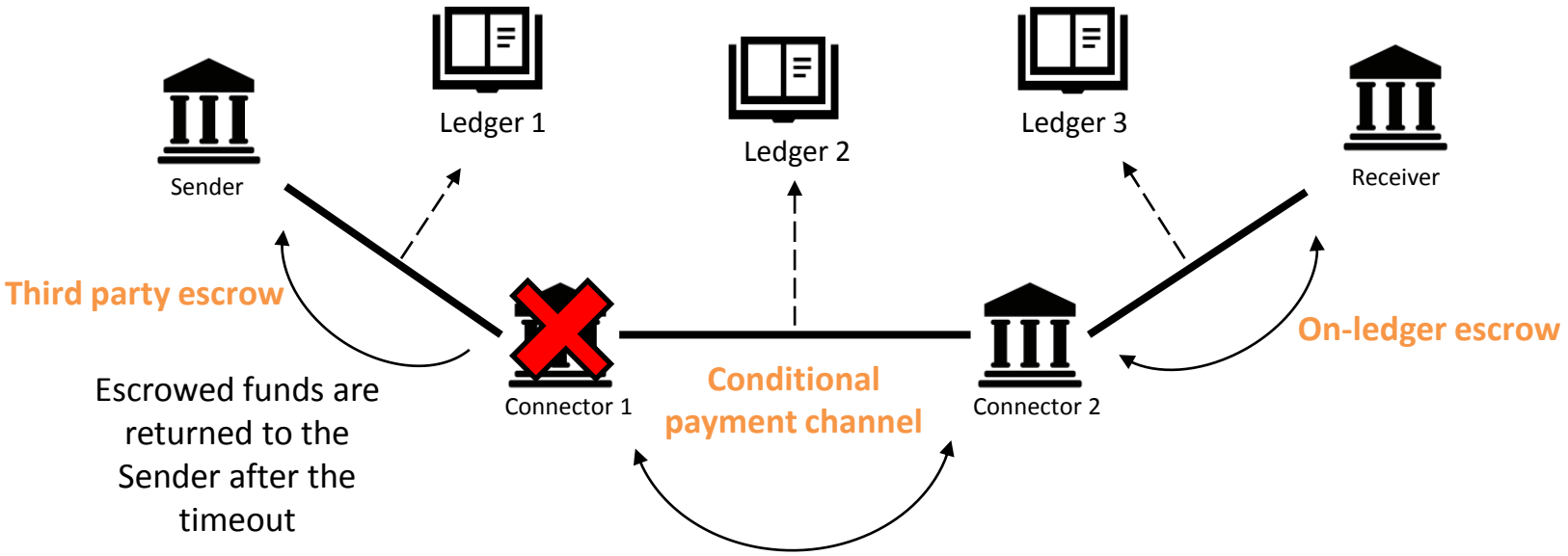
今回検証を行った5つの支払方法(Payment method)

Payment method	On-ledger/ Off-ledger	Escrow/Lock	Enforcement of conditional payment	Specific ledger requirements
Trustline	Off-ledger	No	No enforcement	No
★ On-ledger escrow using HTLC	On-ledger	Yes	Enforced by ledger	Yes
★ Third party escrow	On-ledger	Yes	Enforced by third party	No
Simple payment channel	Off-ledger	Yes	No enforcement	Yes
★ Conditional payment channel with HTLC	Off-ledger	Yes	Enforced by ledger	Yes

「条件付支払を確実に行う仕組み」がある3つの支払方法は、資金移動が確実であり、**安全**。

送金経路内で複数の支払方法が用いられる場合

- 安全な支払方法を採用した取引参加者については、自らの責務（例：タイムアウト内の原像の提示）を完全に果たせば、信用リスクに晒されることはない。
- 中継者2から受領者に資金が受け渡された後に、中継者1が倒産した場合であっても、安全な支払方法を用いた中継者2は資金を受け取れる。
  - 送金者は、安全な支払方法（Third party escrow）を用いているため、タイムアウト後に固定されていた資金が返金される。

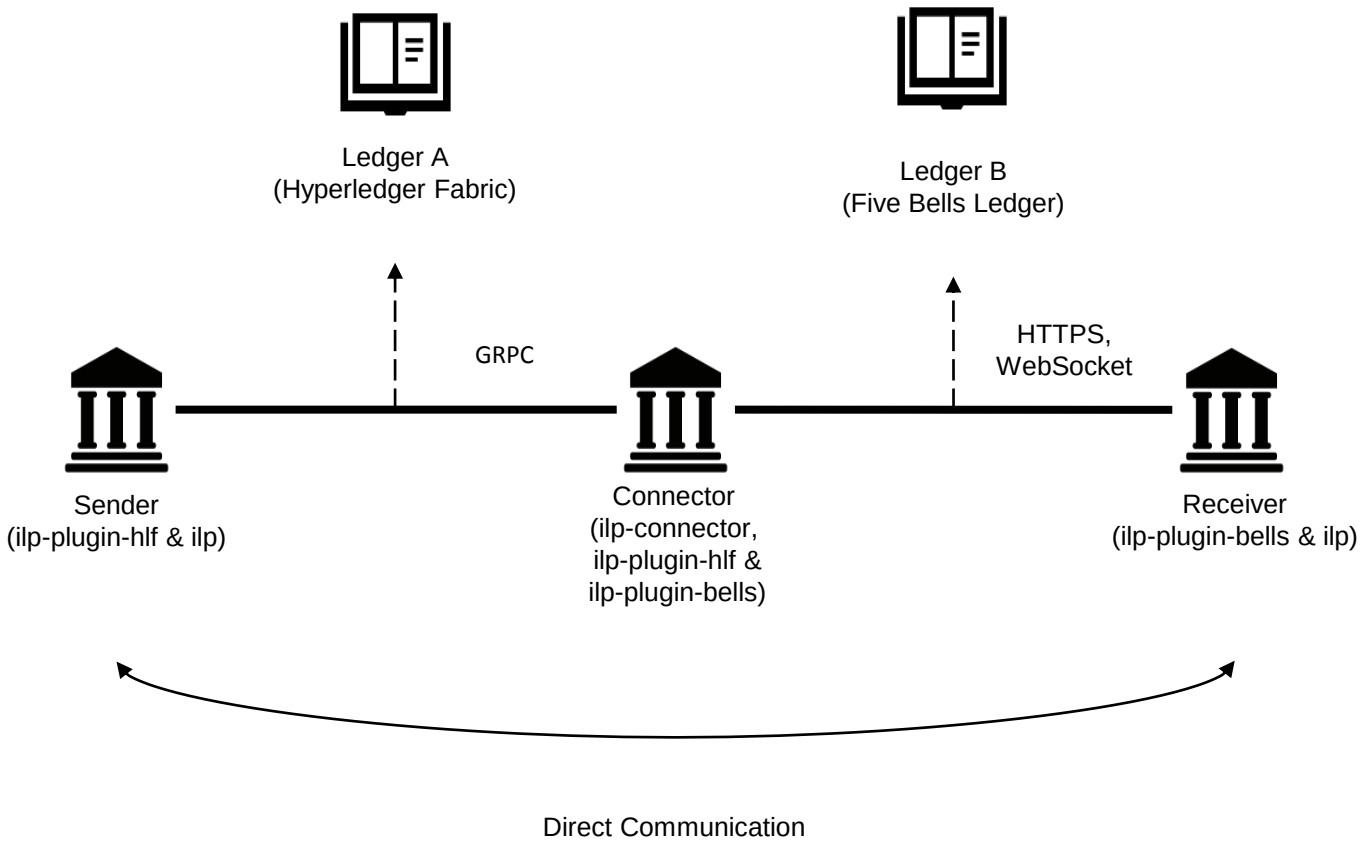


実機検証（Experimentation)

- ILPのオープンソース実装は、Interledger.jsを採用
- 中央集権型台帳の実装を利用するため、（最新版とは異なる）ILPv3を採用
- Hyperledger FabricのILPプラグインは、ステラチームで新規に実装。

実機検証（Experimentation)

- 分散型台帳（Hyperledger Fabric） — 中央集権型台帳（Five Bells Ledger）間の実験時の構成



まとめ①

- フェーズ3では、「DLT台帳と中央集権型台帳間」といったように、異なる種類の台帳間の支払を行う際に用い得る支払方法を、いくつか取り上げたうえ、安全性の観点を重視しつつ、分析（実験を含む）を行った。

- 台帳の様々な組み合わせに対して、送金資金を信用リスクから隔離した状態で固定しながら、各ブロックの支払を同期化することによりクロスボーダー支払の安全性を確保できることを確認した。

- HTLCに基づいたクロスボーダー支払について、DLT台帳のみならず、中央集権型台帳でも実装できることを実験により確認した。

まとめ②

- **フェーズ3は、資金を固定しながら一連の支払を同期化できる支払方法を利用することにより、現在普及しているクロスボーダー送金スキームの安全性を技術面から改善することが可能であることを示唆している。**

- 今後は、そうした新しい支払方法に関する法律面やコンプライアンス面の検討、技術の成熟度やコストベネフィットに関する評価を進めていくことが望まれる。

Ph. 1

Ph. 2

Ph. 3