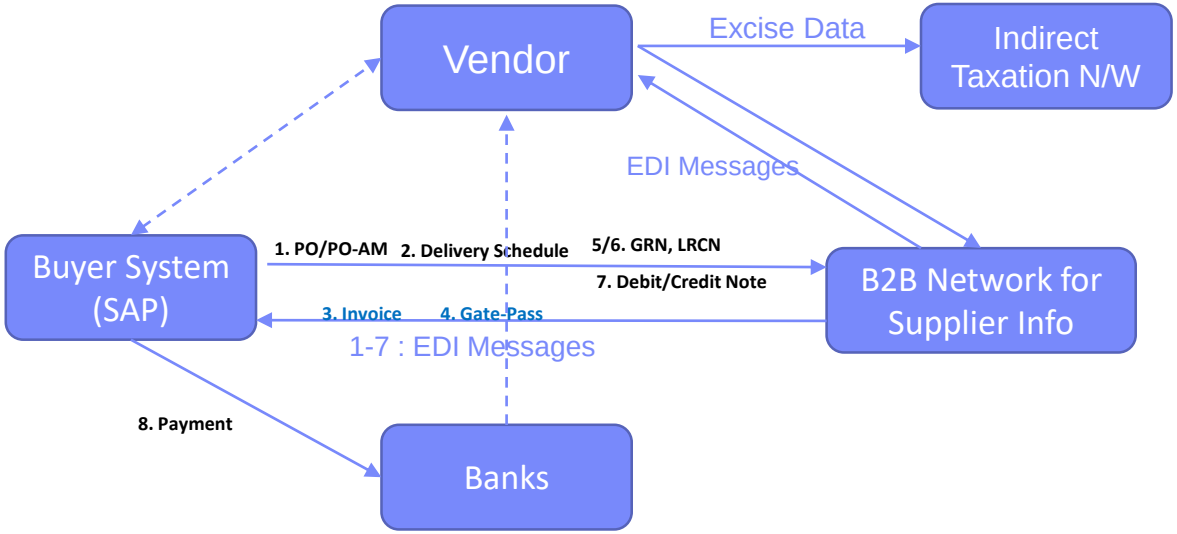
Invoice Reconciliation Using Blockchain

Kameshwaran Sampath, Pankaj Dayama, D. Sai Koti Reddy and Vinayaka D Pandit
IBM Research – India

Invoice Lifecycle Management

IBM **Research**



- ✕ Delayed/ Inconsistent Visibility of Invoice Workflow
- ✕ Asynchronous Dispatch Schedule and Shipping Notice
- ✕ Disputes on Payment Delays/ Amount Paid
- ✕ Complex Reconciliation Process for Input Tax Credits

- Multiple distrusting parties
- Sequence of point-to-point EDI messages – from Purchase Order till Payment
- Multiple blind spots in the process for each of the parties – not all parties have the same state of truth at all times
- Process non-compliance results in delay, disputes, and inefficiencies

© 2020 IBM Corporation

Why Invoice Reconciliation using Blockchain?

Ad-hoc and Non-core Vendors

- OEMs often deal with majority of ad-hoc, non-strategic vendors
- Number of transactions with such vendors could be less (< 40%) but are often source of non-compliance, disputes, and eventual delay in both operations and finance
- Blockchain smart contracts ensures a *standard* invoice lifecycle process is choreographed with all vendors, thus avoiding non-compliances and disputes

Non-compliance in EDI (Electronic Data Interchange)

- Non-compliance in EDI (header, line item errors) are common with ad-hoc vendors
- Disputes due to non-compliance are settled through length off-line resolutions

provides the choreography

- Blockchain smart contracts can flag, validate, and auto-correct non-compliance in EDI

© 2020 IBM Corporation

Why Invoice Reconciliation using Blockchain?

Multi-party Disputes

- EDI are 1-1 message exchange between two parties
- Multi-parties (OEM, vendor, carrier) do not have full visibility on messages exchanged
  - Goods received notice from OEM to Vendor is not visible to carrier and can contest the claim on the delivery time
- Blockchain ensures shared visibility and consensus among multiple parties

Choreography on Messaging Infrastructure

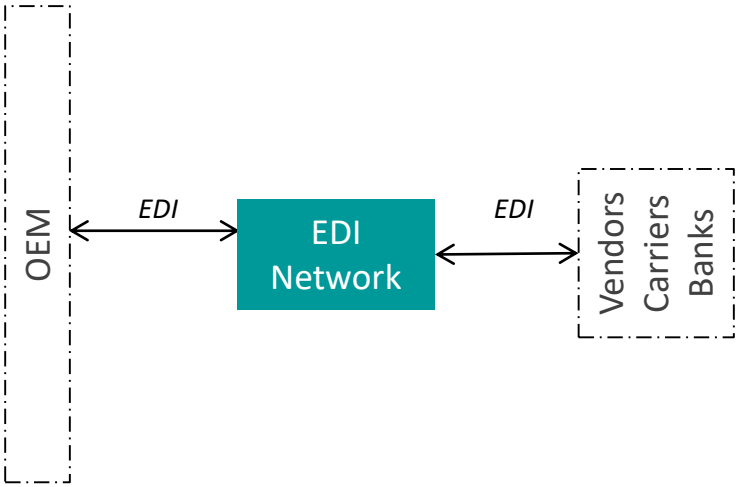
- EDI Business network is a messaging infrastructure that ensures EDI exchanges
- Choreography of processes (like invoice lifecycle) is required for dispute-free, compliant, and efficient workflow
- Blockchain smart contracts on top of EDI Business network provides the choreography

© 2020 IBM Corporation

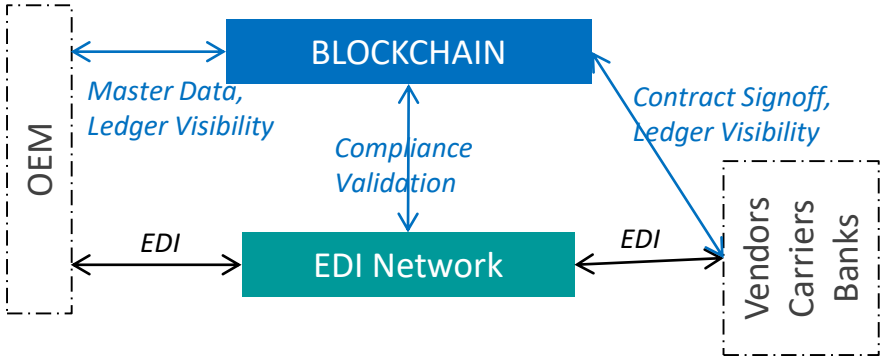
Invoice Reconciliation for OEM

IBM **Research**

As-Is Process



Blockchain Assisted Compliance



ISSUES:

- Collaboration via point to point message infrastructure – no shared state
- PO-Invoice reconciliation is costly, manual, with sometimes lengthy dispute resolution
- Lack of global visibility across the network

IMPROVEMENTS:

- Shared state provided by new blockchain "overlay" infrastructure
- Blockchain enforces: Choreographed process execution, Compliance validation, Single world state, Contract sign-offs and fast audits
- → Invoice reconciliation is rapid and less error prone

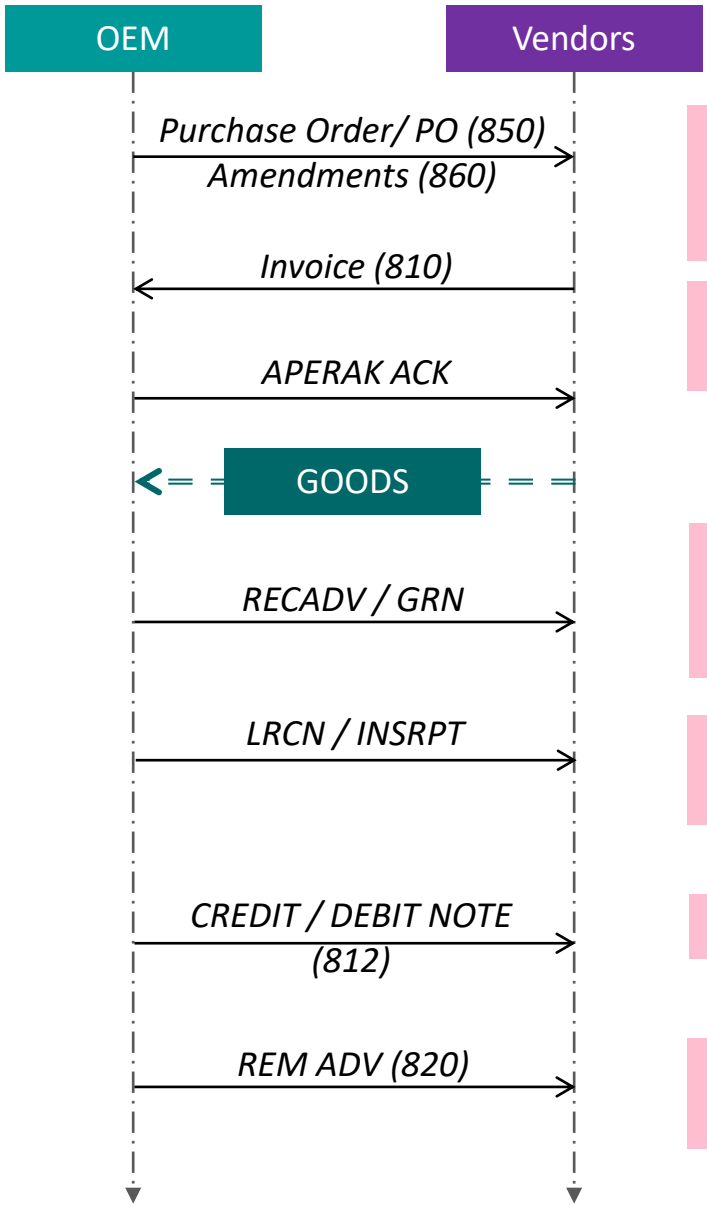
EDI: PO, Invoice, REC ADV / GRN, INSRPT / LRCN, Credit/Debit Note, REM ADV

© 2020 IBM Corporation

Doc Flows & User Actions

IBM **Research**

Typical As-Is Process



Potential Disputes and EDI Non-compliance

- Wrong data (Vendor, Plant, Material)
- Cost conflict

- Wrong invoice line items
- Tax miscalculated

- Disagreement among OEM, vendor, and carrier on the goods delivery time

- Wrong line items on inspection report

- Mismatch on invoice line items

- Mismatch on invoice and credit/debit note line items

Blockchain Assisted Compliance

- Cross Validation with Master Data (Vendor, Plant, Material)
- Contract Sign-Off

- Vendor Acknowledges PO

- Cross Validation with PO
- Tax Validation
- Workflow Creation

- Cross Validation with Invoice
- Goods received time consensus
- Workflow Update

- Cross Validation with Invoice
- Workflow Update

- Vendor Acknowledges Receipt of Rejected Items

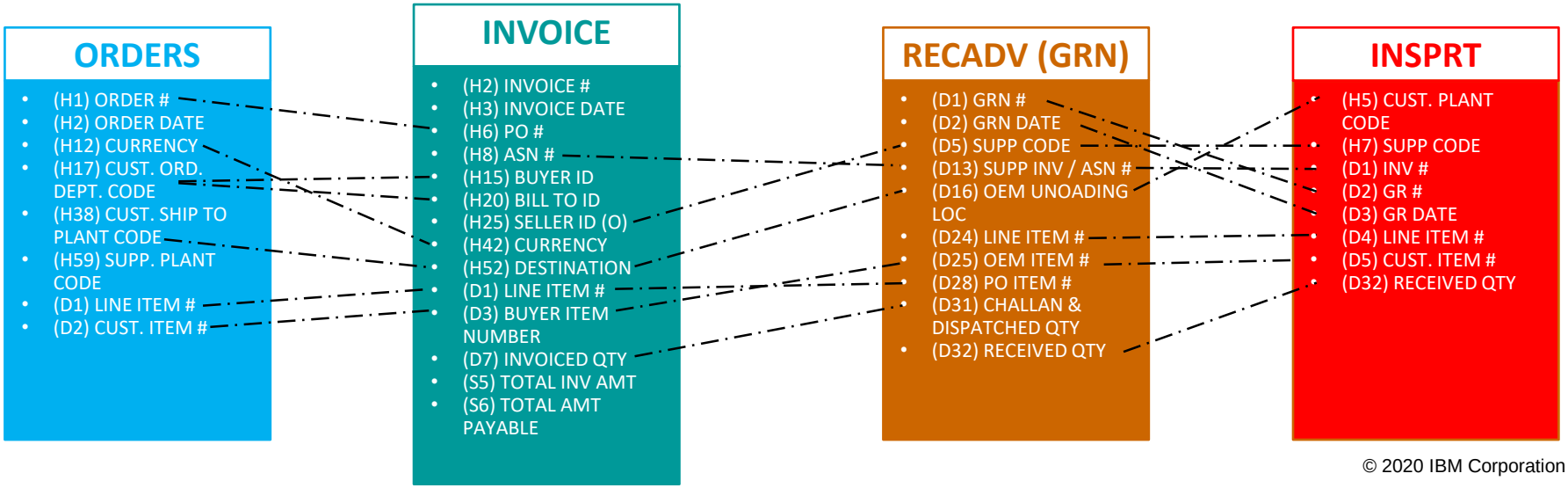
- New ledger entry

- Cross Reference with Invoice
- Cross Reference with Credit/Debit Note
- Workflow Update

Compliance Validation

- Smart Contract Validations
  - **Document**:
    - Duplicate/Updated
  - **Field Level**:
    - Mandatory/Optional
    - Field Value Integrity (Date formats, Integers, Positive, Non-Zero, …)
  - **Intra-Document**:
    - Computational (Tax, Price, Total Amounts, …)
    - Conditional (Only 2 of the 3 tax fields can be non-zero, …)
  - **Inter-Documents**:
    - Precedence Validation (PO and ASN should be received before Invoice)
    - Cross Validation (Cross validating dependent fields across documents)
- Errors and warnings are stored and EDIs with non-zero errors are given INVALID compliance status, which needs to be acted on.

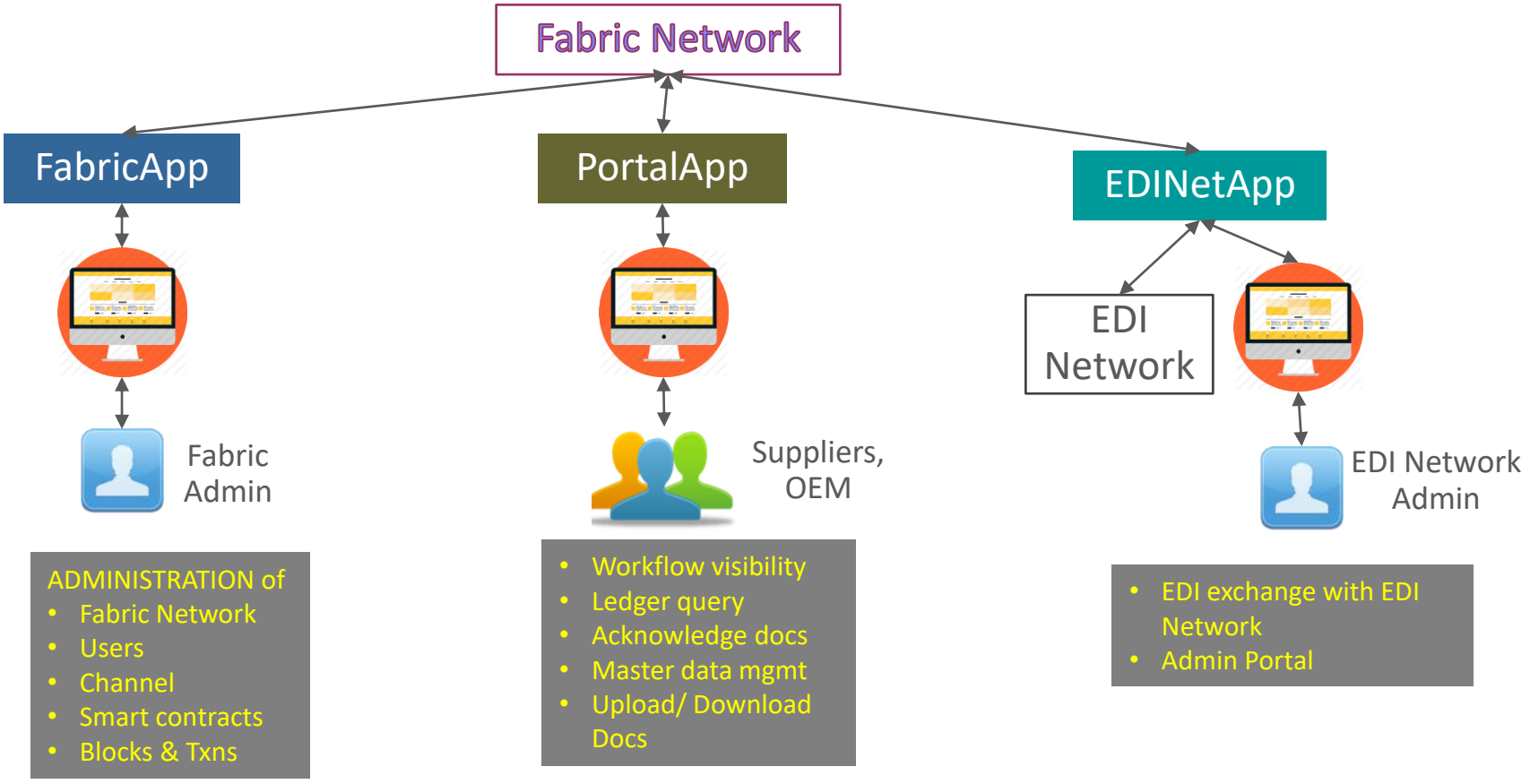
Example: Inter-Document CV

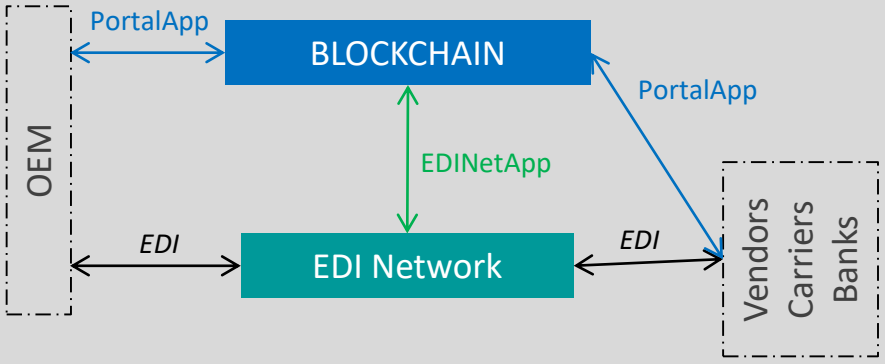


© 2020 IBM Corporation

Architecture

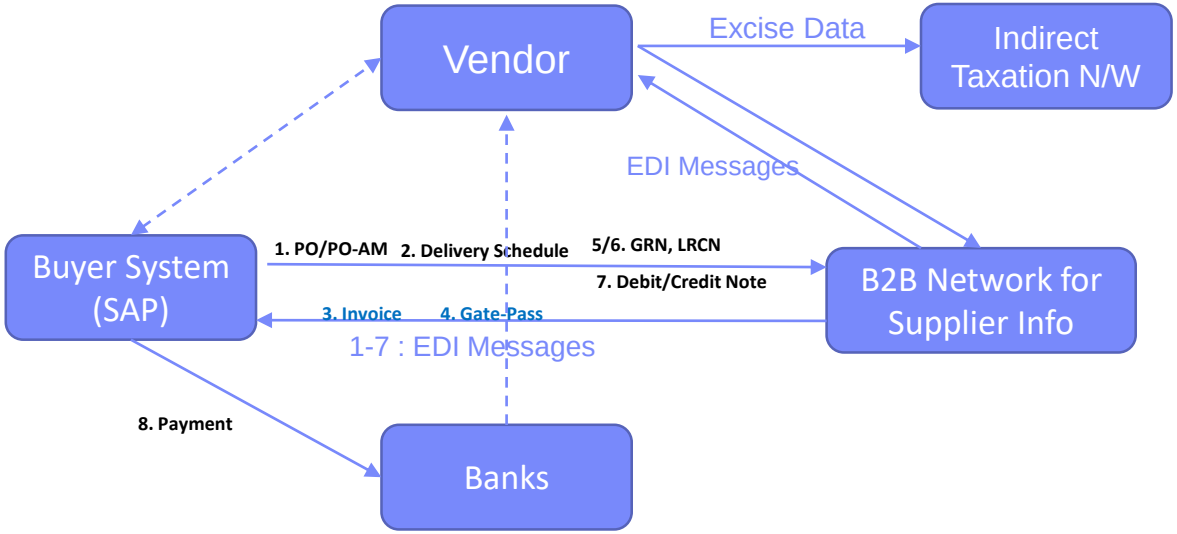
IBM **Research**





Invoice Lifecycle Management

IBM **Research**



- ✗ Delayed/ Inconsistent Visibility of Invoice Workflow
- ✗ Asynchronous Dispatch Schedule and Shipping Notice
- ✗ Disputes on Payment Delays/ Amount Paid
- ✗ Complex Reconciliation Process for Input Tax Credits

- ✓ Real-time Shared Visibility of Invoice Lifecycle
- ✓ Seamless Integration with Different Systems
- ✓ Buyer Payment Lead Time Visibility
- ✓ Automated Payment/Tax Reconciliation
- ✓ Trusted KPIs
- ✓ Identification of common errors
- ✓ Can also handle remote product assembly lifecycle

Thank You!

Email : d.saikotireddy@in.ibm.com

© 2020 IBM Corporation