Provenance Built on Open Source

Hyperledger In-Depth Webinar with BTP

25 January 2022

01

Provenance

Provenance

The term *provenance* is often associated with the art world. However, recording and sharing provenance data can provide significant value across industries…

Provenance captures the origin and life journey — including the ownership history — of a physical or digital asset.

Why Provenance Matters

Transparency & Trust

Anti-fraud & Anti-counterfeiting

Consumer Safety

Ethical Sourcing

Environmental Sustainability

Resilient Supply Chains

02

Chronicle

Our Value Proposition

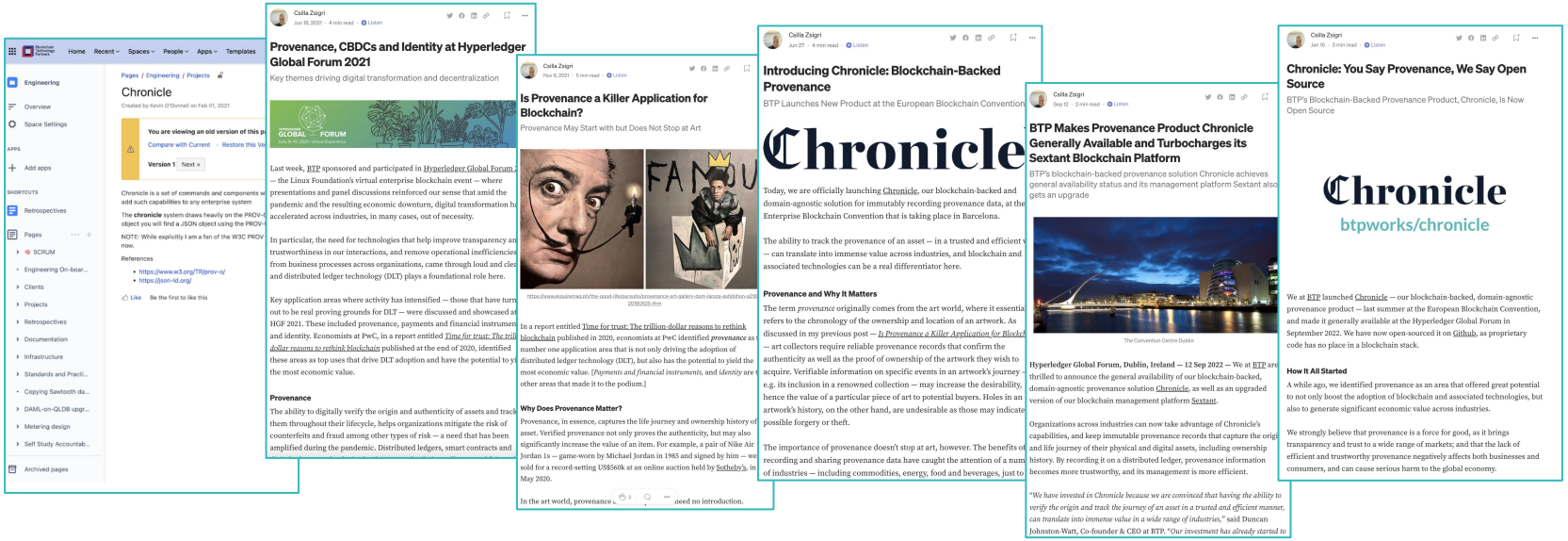
We are a digital provenance company with a mission to make assets trustworthy. We believe that provenance is a force for good, as it brings transparency and trust to a wide range of markets.

We make it easy for organizations to record and query immutable provenance information on a distributed ledger, about any asset, in any domain, across multiple parties.

We do this by providing Chronicle, our open-source, blockchain-backed, domain-agnostic provenance product.

btp.works

The Chronicle Story



Chronicle

btp.works/chronicle

Open-source, blockchain-backed, domain-agnostic product for immutably recording and querying provenance data.

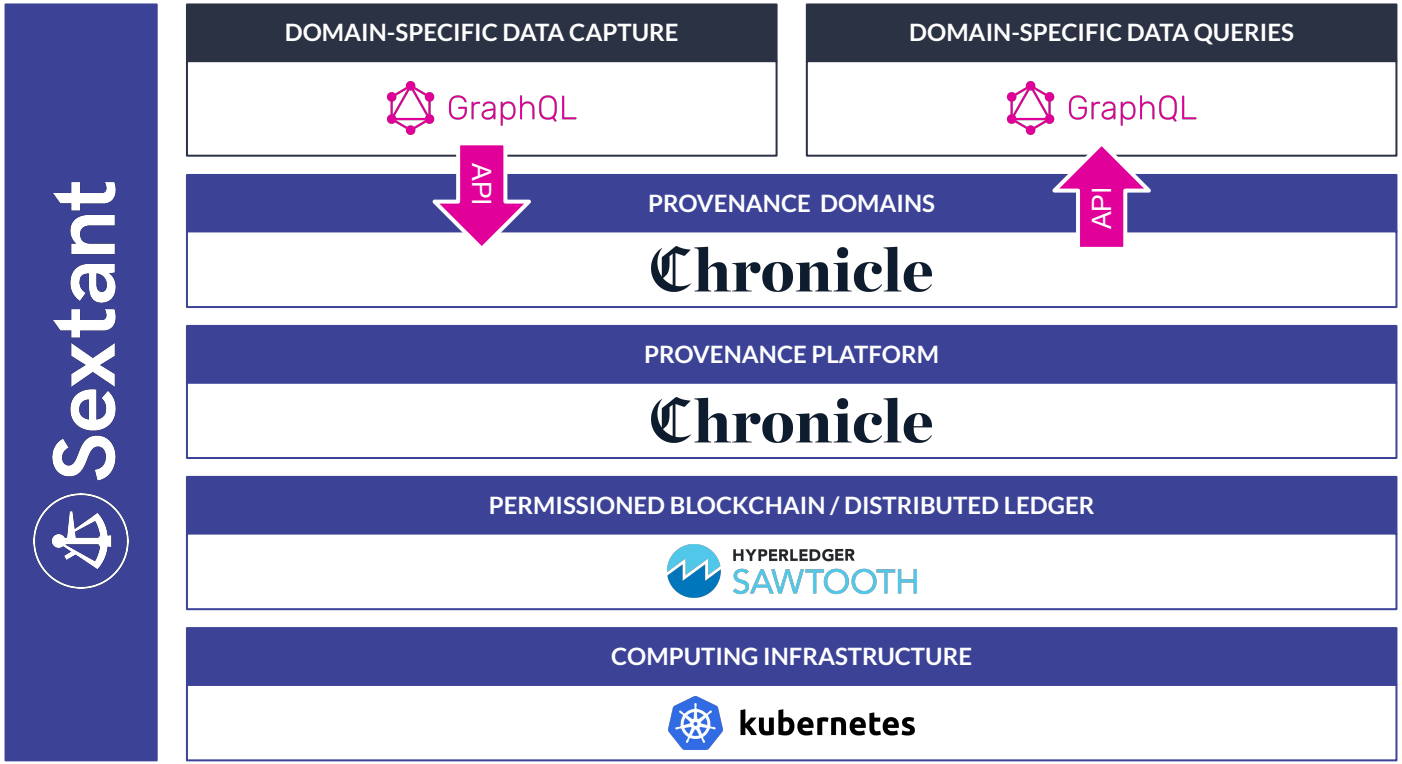
Built on the **W3C PROV Ontology** specification; uses the lightweight **JSON-LD** linked data format, and the data query language **GraphQL**.

Records provenance information of any physical or digital asset on a distributed ledger. Available with **Hyperledger Sawtooth** as its default backing ledger.

Easily configurable to enable users to capture provenance information for a range of domain-specific applications.

Delivered by Sextant, to facilitate its deployment and management, as well its integration with enterprise systems.

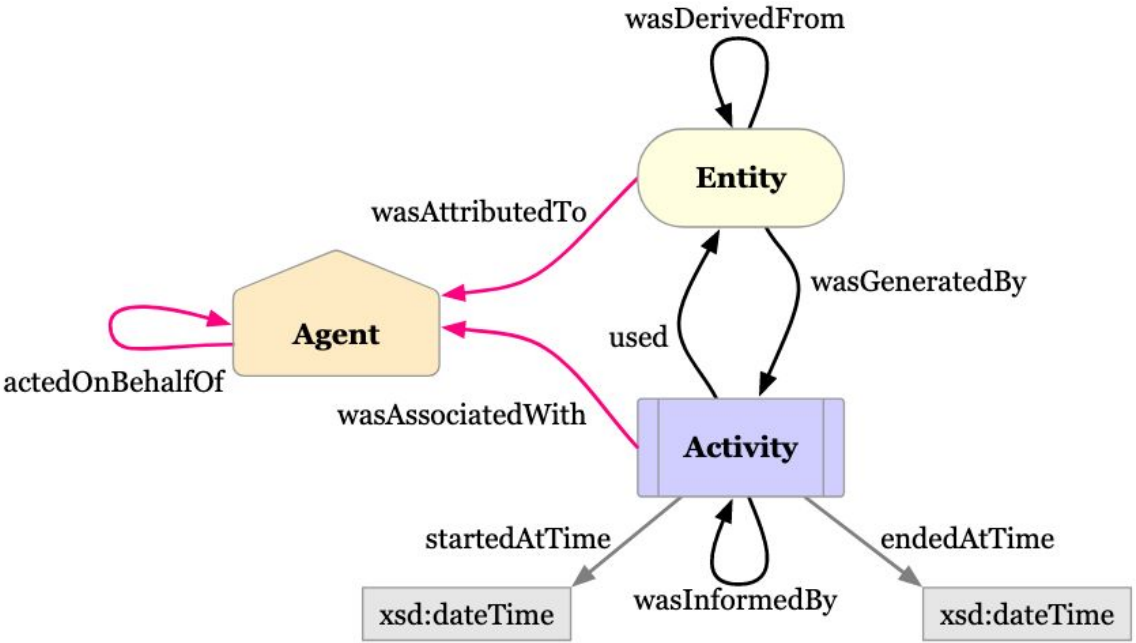
Chronicle Stack



03

Demo

PROV-O: Domain-Agnostic Ontology



Source: PROV-O: The PROV Ontology - At a Glance

Domain-Specific Ontology

Domain: Manufacturing

Assets (aka *Entities* in PROV-O speak): Item, Certificate

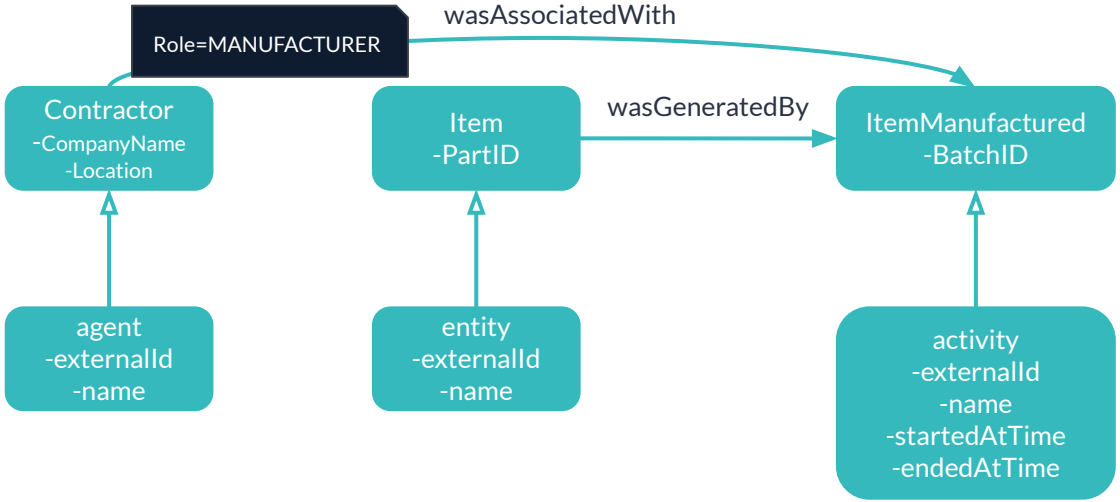
Agents: Contractor

Activities: ItemManufactured, ItemCertified

See the Manufacturing Example Guide

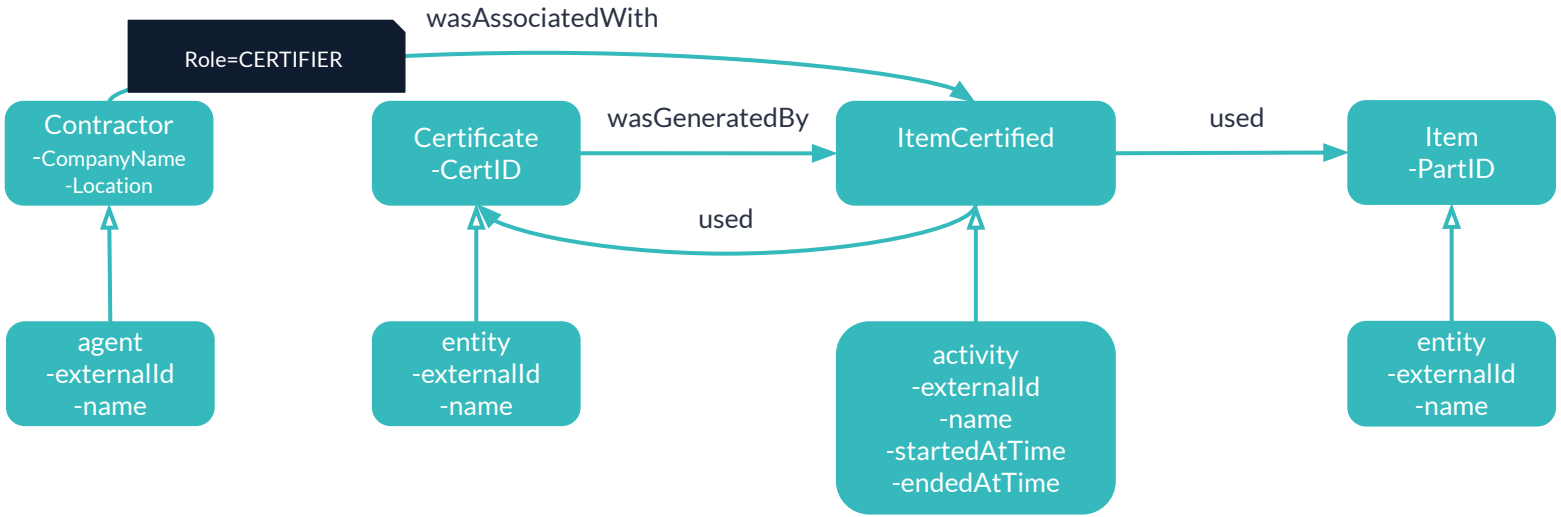
Item Manufacturing

Although various agents may be involved in the *ItemManufactured* activity, in this example, the *Contractor* is the only agent responsible for the manufacturing of the *Item*.



Item Certification

The *Contractor* issues a *Certificate* for each item. The *ItemCertified* activity uses this item.



Resources

Join Chronicle Works on Slack

Github - btpworks/chronicle

Github - btpworks/chronicle-examples

Github - blockchaintp/sawtooth-core

Chronicle: You Say Provenance, We Say Open Source

Is Provenance a Killer App for Blockchain?

btp.works

Edinburgh | Barcelona | London | New York | San Francisco

04

Appendix

Getting Started

Clone Repo

git clone https://github.com/btpworks/chronicle-examples.git
cd chronicle-examples

Build Manufacturing Example

make run-manufacturing

Connect to GraphQL Playground

http://127.0.0.1:9982/

See the README.md

Subscription

```
subscription {
  commitNotifications {
    delta
  }
} # Write your query or mutation here
```

Define Contractors

```
mutation {
  agent1: defineContractorAgent(
    externalId: "helicoptersplc"
    attributes: {
      companyNameAttribute: "Helicopters PLC"
      locationAttribute: "Bristol, England"
    }
  ) {
    context
    txId
  }
  agent2: defineContractorAgent(
    externalId: "acmecorp"
    attributes: {
      companyNameAttribute: "ACME Corp"
      locationAttribute: "Burbank, California"
    }
  ) {
    context
    txId
  }
}
```

Query Contractors

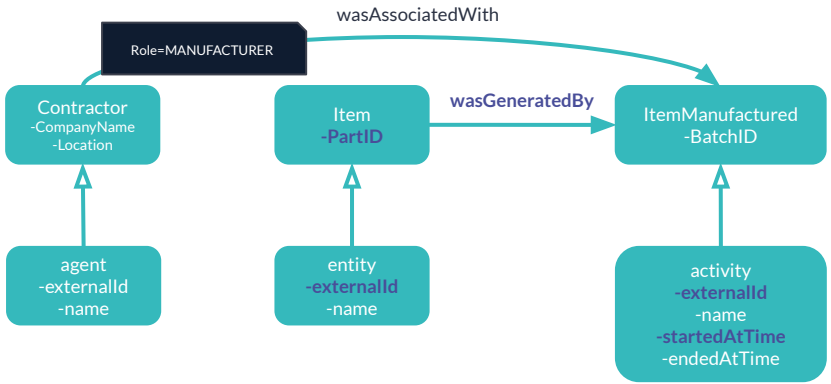
```
query {
  agentsByType(agentType: ContractorAgent) {
    nodes {
      __typename
      ... on ContractorAgent {
        externalId
        companyNameAttribute
        locationAttribute
      }
    }
  }
}
```

Define ItemManufactured Activity

```
mutation {
  defineItemManufacturedActivity(
    externalId: "rotorblademake-run-001"
    attributes: { batchIDAttribute: "run-001" }
  ) {
    context
    txId
  }
  wasAssociatedWith(
    activity: { externalId: "rotorblademake-run-001" }
    responsible: { externalId: "helicoptersplc" }
    role: MANUFACTURER
  ) {
    context
    txId
  }
}
```

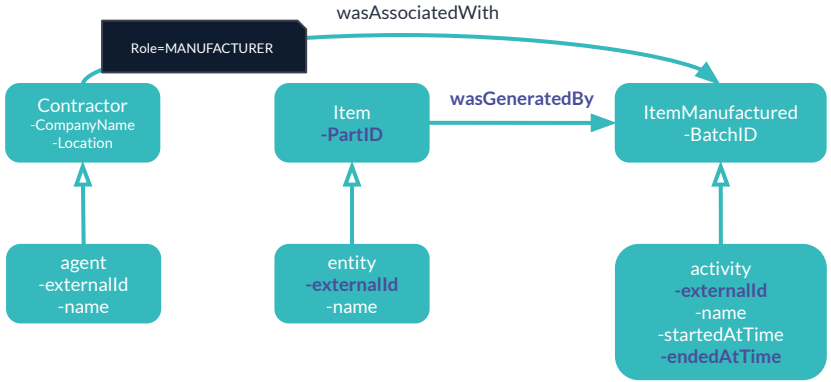
Start ItemManufactured Activity

```
mutation {
  startActivity(id: { externalId: "rotorblademake-run-001" }) {
    context
    txId
  }
  i1: defineItemEntity(
    externalId: "rotorblade-run-001-001"
    attributes: { partIDAttribute: "run-001-001" }
  ) {
    context
    txId
  }
  g1: wasGeneratedBy(
    id: { externalId: "rotorblade-run-001-001" }
    activity: { externalId: "rotorblademake-run-001" }
  ) {
    context
    txId
  }
}
```



End ItemManufactured Activity

```
mutation {
  i2:defineItemEntity(
    externalId: "rotorblade-run-001-002"
    attributes: { partIDAttribute: "run-001-002" }
  ) {
    context
    txId
  }
  g2:wasGeneratedBy(
    id: { externalId: "rotorblade-run-001-002" }
    activity: { externalId: "rotorblademake-run-001" }
  ) {
    context
    txId
  }
  endActivity(id: { externalId: "rotorblademake-run-001" }) {
    context
    txId
  }
}
```

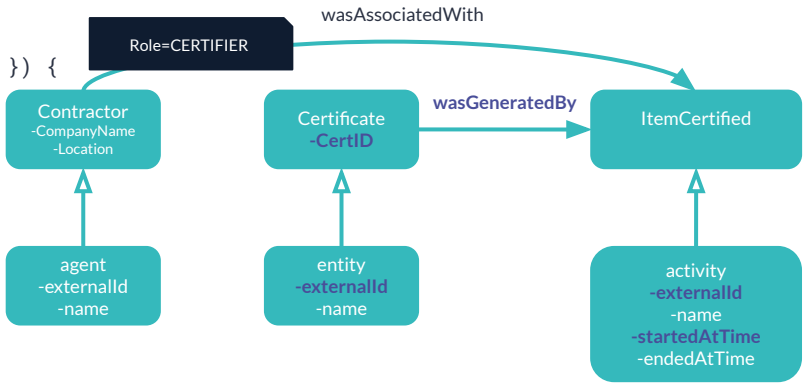


Define ItemCertified Activity

```
mutation {
  defineItemCertifiedActivity(
    externalId: "rotorbladecert-run-001-001") {
    context
    txId
  }
  wasAssociatedWith(
    activity: {
      externalId: "rotorbladecert-run-001-001" }
    responsible: { externalId: "helicoptersplc" }
    role: CERTIFIER
  ) {
    context
    txId
  }
}
```

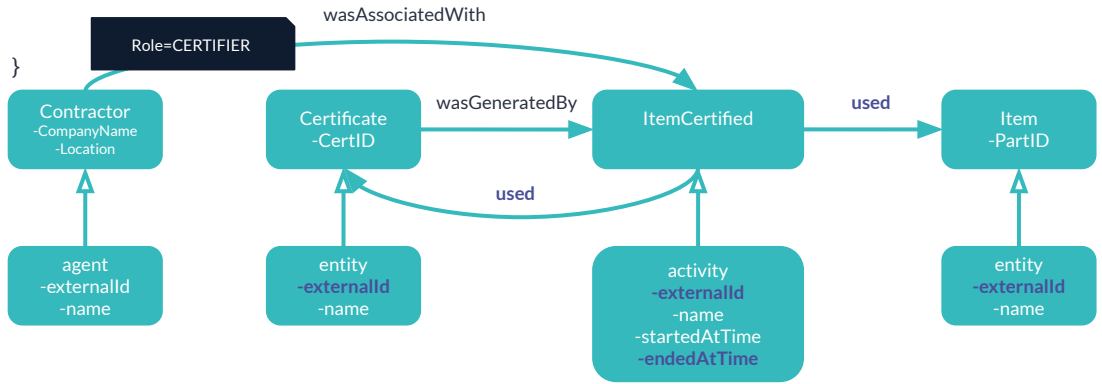
Start ItemCertified Activity

```
mutation {
  startActivity(
    id: { externalId: "rotorbladecert-run-001-001" }) {
    context
    txId
  }
  defineCertificateEntity(
    externalId: "rotorbladecert-run-001-001"
    attributes: { certIDAttribute: "run-001-001" }
  ) {
    context
    txId
  }
  wasGeneratedBy(
    id: { externalId: "rotorbladecert-run-001-001" }
    activity: { externalId: "rotorbladecert-run-001-001" }
  ) {
    context
    txId
  }
}
```



End ItemCertified Activity

```
mutation {
  cert: used(
    id: { externalId: "rotorbladecert-run-001-001" }
    activity: {
      externalId: "rotorbladecert-run-001-001" }
  ) {
    context
    txId
  }
  blade: used(
    id: { externalId: "rotorblade-run-001-001" }
    activity: {
      externalId: "rotorbladecert-run-001-001" }
  ) {
    context
    txId
  }
  endActivity(id: { externalId: "rotorbladecert-run-001-001" }) {
    context
    txId
  }
}
```



Query Item #1

```
query {
  q1: entityById(id: { externalId: "rotorblade-run-001-001" }) {
    ... on ItemEntity {
      partIDAttribute
      wasGeneratedBy {
        ... on ItemManufacturedActivity {
          id
        }
      }
    }
  }
  q2: entityById(id: { externalId: "rotorbladecert-run-001-001" }) {
    ... on CertificateEntity {
      certIDAttribute
      wasGeneratedBy {
        ... on ItemCertifiedActivity {
          id
        }
      }
    }
  }
}
```

Query Item #2

```
query {
  q3: entityById(id: { externalId: "rotorblade-run-001-002" }) {
    ... on ItemEntity {
      partIDAttribute
      wasGeneratedBy {
        ... on ItemManufacturedActivity {
          id
        }
      }
    }
  }
  q4: entityById(id: { externalId: "rotorbladecert-run-001-002" }) {
    ... on CertificateEntity {
      certIDAttribute
      wasGeneratedBy {
        ... on ItemCertifiedActivity {
          id
        }
      }
    }
  }
}
```