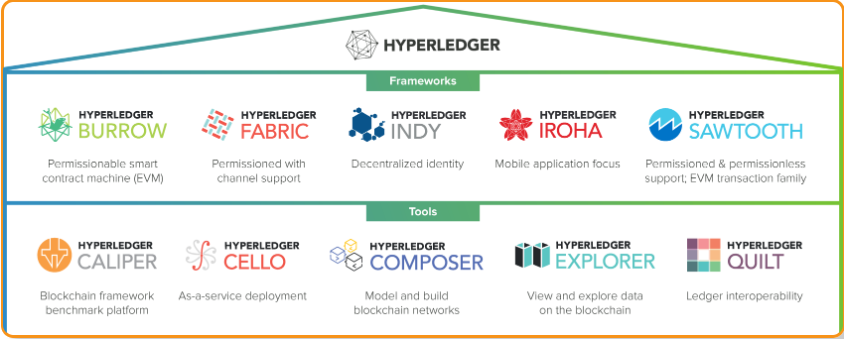
# **Nghiên cứu block trên trong quản lý công văn**

1. Cài đặt hệ thống blockchain hyperledger, cài này là quan trong nhất.
2. Tìm hiểu về quy trình quản lý công văn, cách triển khai chương trình quản lý công văn (web) triển khai trong hệ thống blockchain hyperledger

# **Hyperledger là gì?**

* [**Hyperledger**](https://bitcoinvietnamnews.com/hyperledger-la-gi) (hoặc dự án Hyperledger) là dự án mã nguồn mở được tạo ra để thúc đẩy công nghệ blockchain xuyên suốt ngành công nghiệp. Được ra đời vào tháng 12/2015 bởi Linux Foundation và được hỗ trợ bởi các công ty lớn như IBM, Intel và SAP.
* Linux Foundation là một tổ chức phi lợi nhuận cho phép đổi mới hàng loạt thông qua mã nguồn mở, lưu trữ Hyperledger. Linux Foundation cũng cho phép cộng đồng các nhà phát triển trên toàn thế giới làm việc cùng nhau và chia sẻ ý tưởng, cơ sở hạ tầng và mã code.

# **Các dự án và công cụ của Hyperledger**



# **Assets**

-Trong hệ thống phân tán cho việc trao đổi giá trị Thay mặt cho giá trị là Assets

-Tài sản hữu hình: ô tô, ngôi nhà

-Tài sản vô hình: giấy phép đky xe, bìa đỏ

-JSON hoặc Binary

# **Hyperledger Fabric**

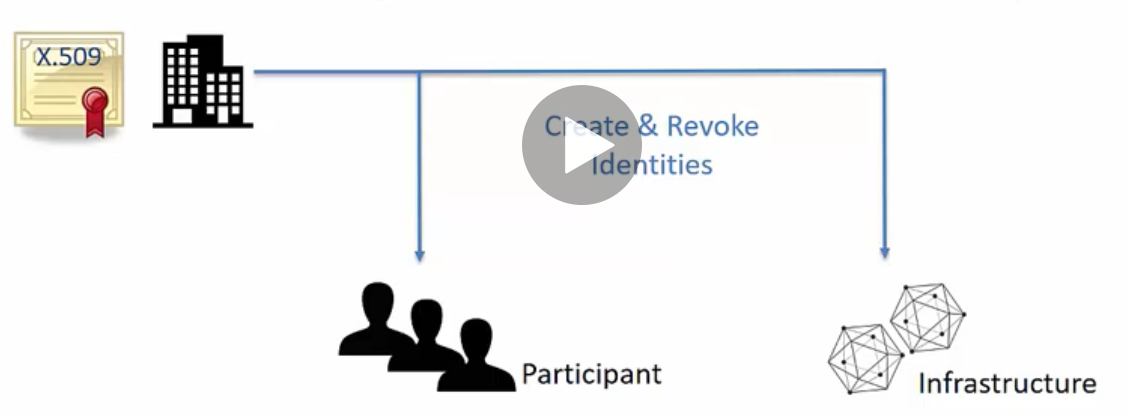
Permissioned Network: Tất cả danh tính được biết đến

Membership service provider: (MSP) vai trò + giới hạn truy cập

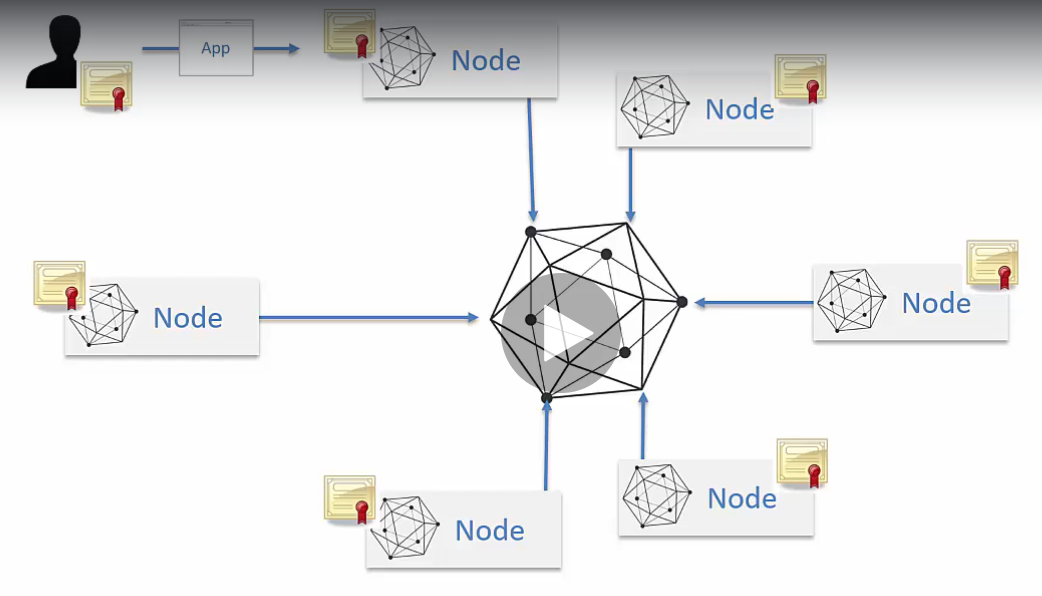
Member là các thực thể tách biệt hợp pháp

Identity được quản lý bởi chứng thực

Chứng thực được cấp và thu hồi bởi CA



A mạng HyperledgerFabric có thể có 1 hoặc nhiều MSP



Hãy nghĩ về nút như một điểm cuối giao tiếp trong mạng blockchain. Các nút kết nối với nút khác và đó là cách mạng blockchain được hình thành.

Trong các mạng blockchain công khai như Ethereum và Bitcoin, tất cả các nút đều bằng nhau. Để tham gia vào các mạng công cộng này, người ta chỉ cần tải xuống phần mềm nút thường được gọi là Ví, Tạo tài khoản và Thực thi nút.

Có ba loại nút riêng biệt.

- Đầu tiên là Client (nút khách hàng): Đây là nút mà các ứng dụng sử dụng để bắt đầu các giao dịch.

- Peers (các đồng nghiệp) là các nút giữ sổ cái đồng bộ trên mạng.

- Orderers (Người đặt hàng) là xương sống truyền thông cho mạng blockchain. Họ chịu trách nhiệm phân phối của các giao dịch.

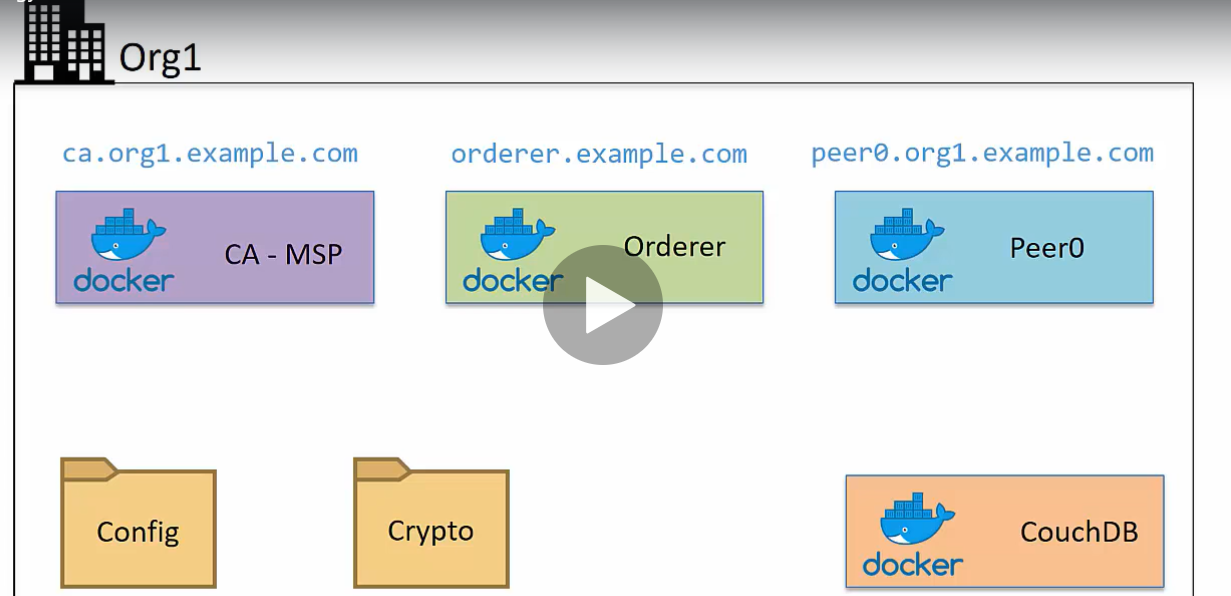
Transactions trong mỗi mạng là độc lập  
Members có thể tham gia trên nhiều mạng Hyperledger Blockchain

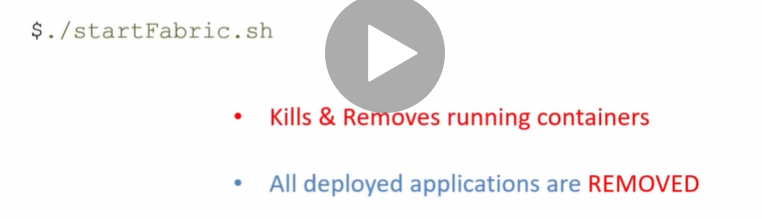
Kênh (Channel)

* Peers kết nối tới kênh
* Sổ cái độc lập trong mỗi kênh

# **Hyperledger Composer**

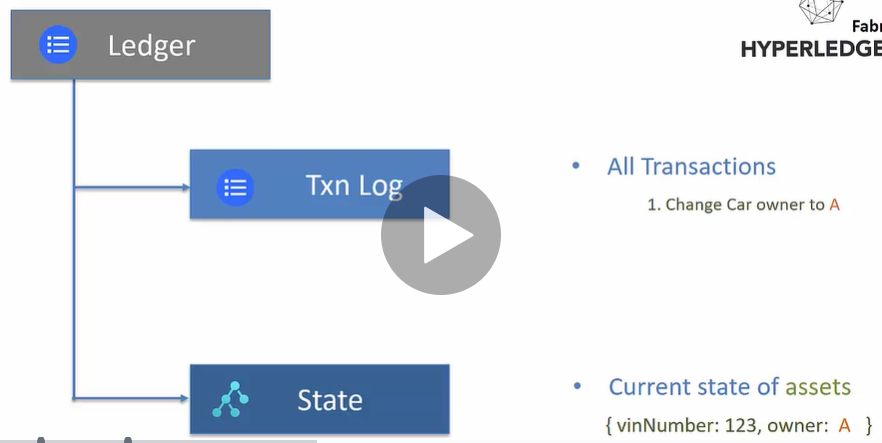
* Là một bộ công cụ để thúc giục sự phát triển của ứng dụng blockchain



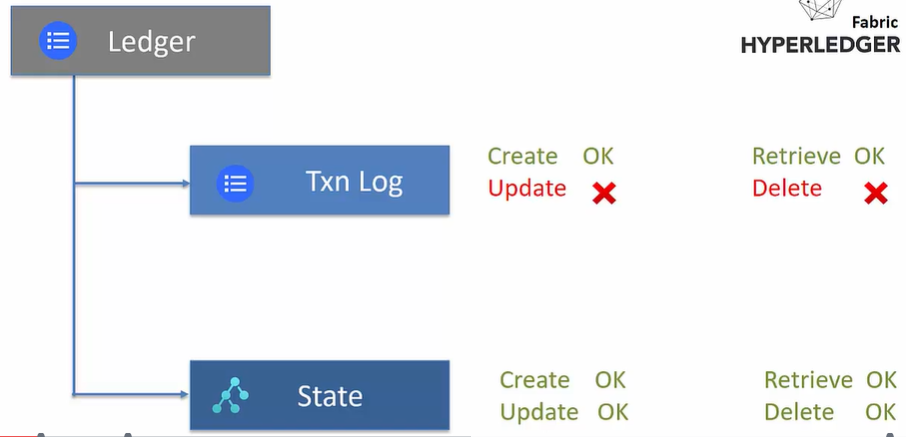


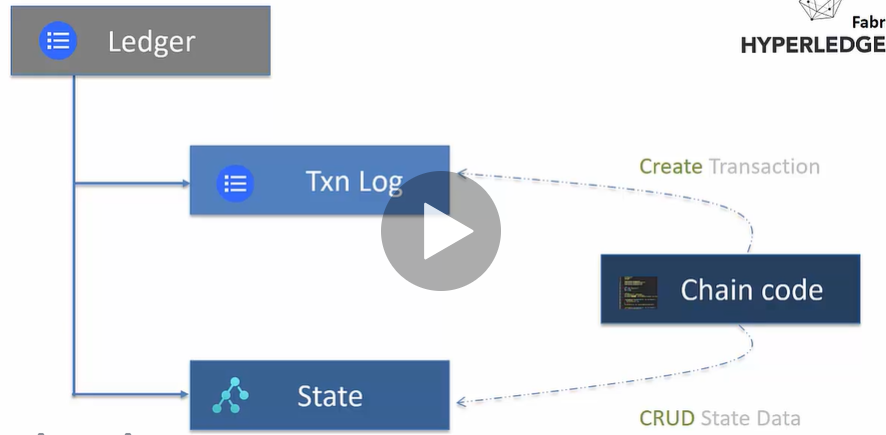


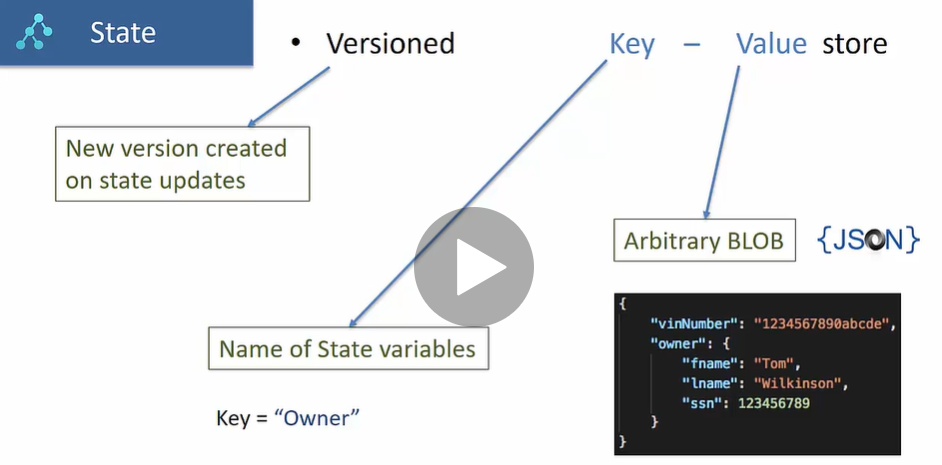
# **Ledger**



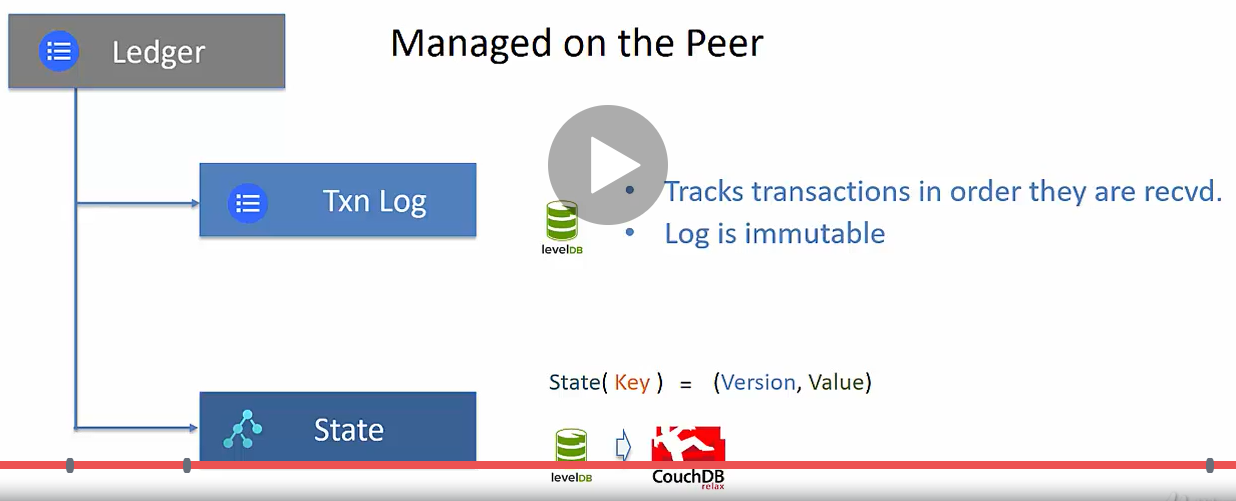


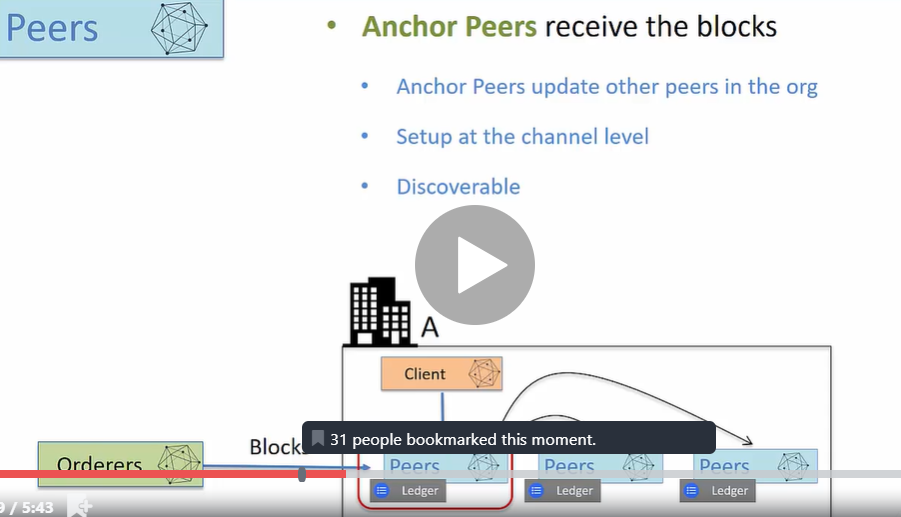


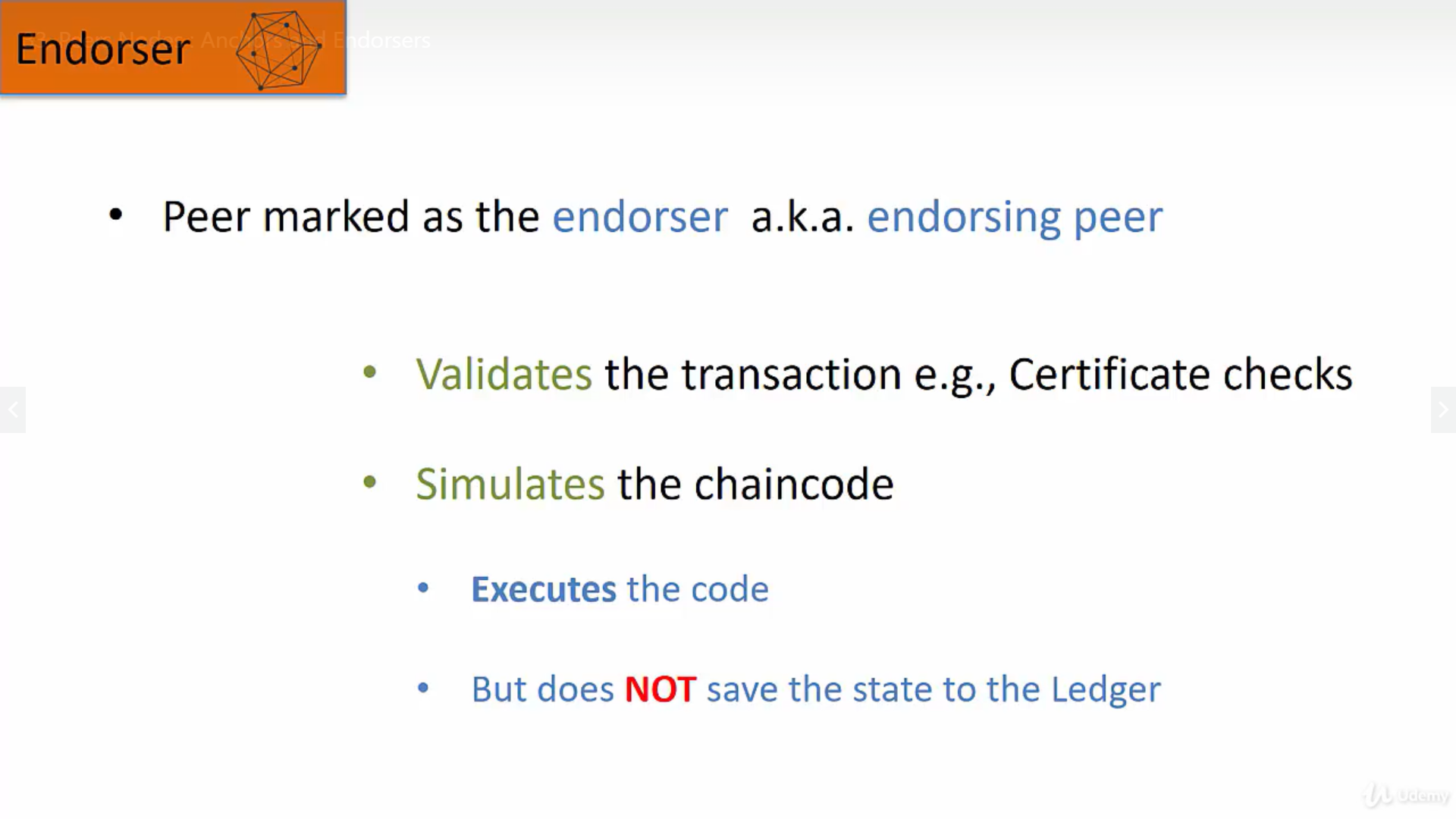


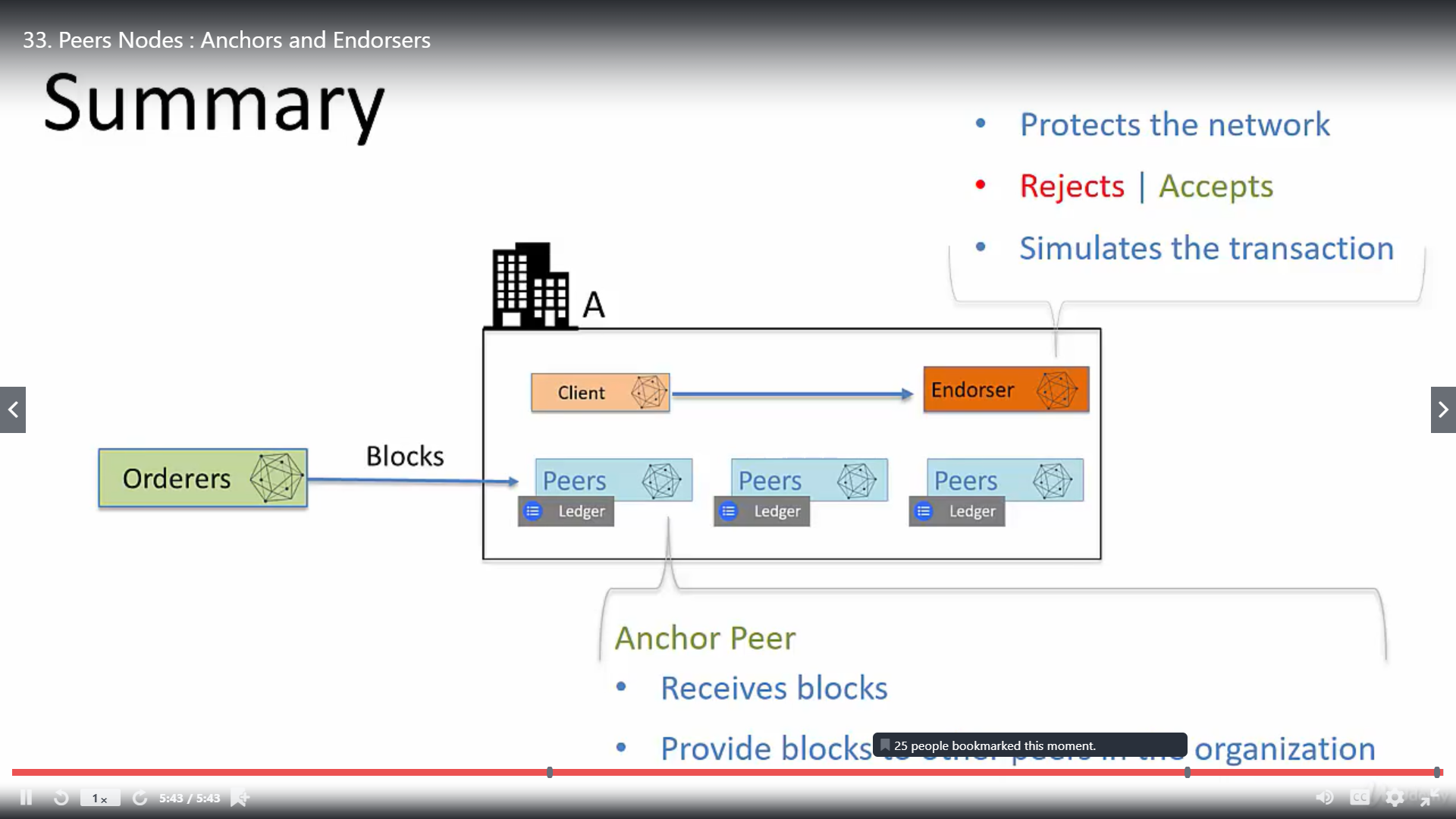


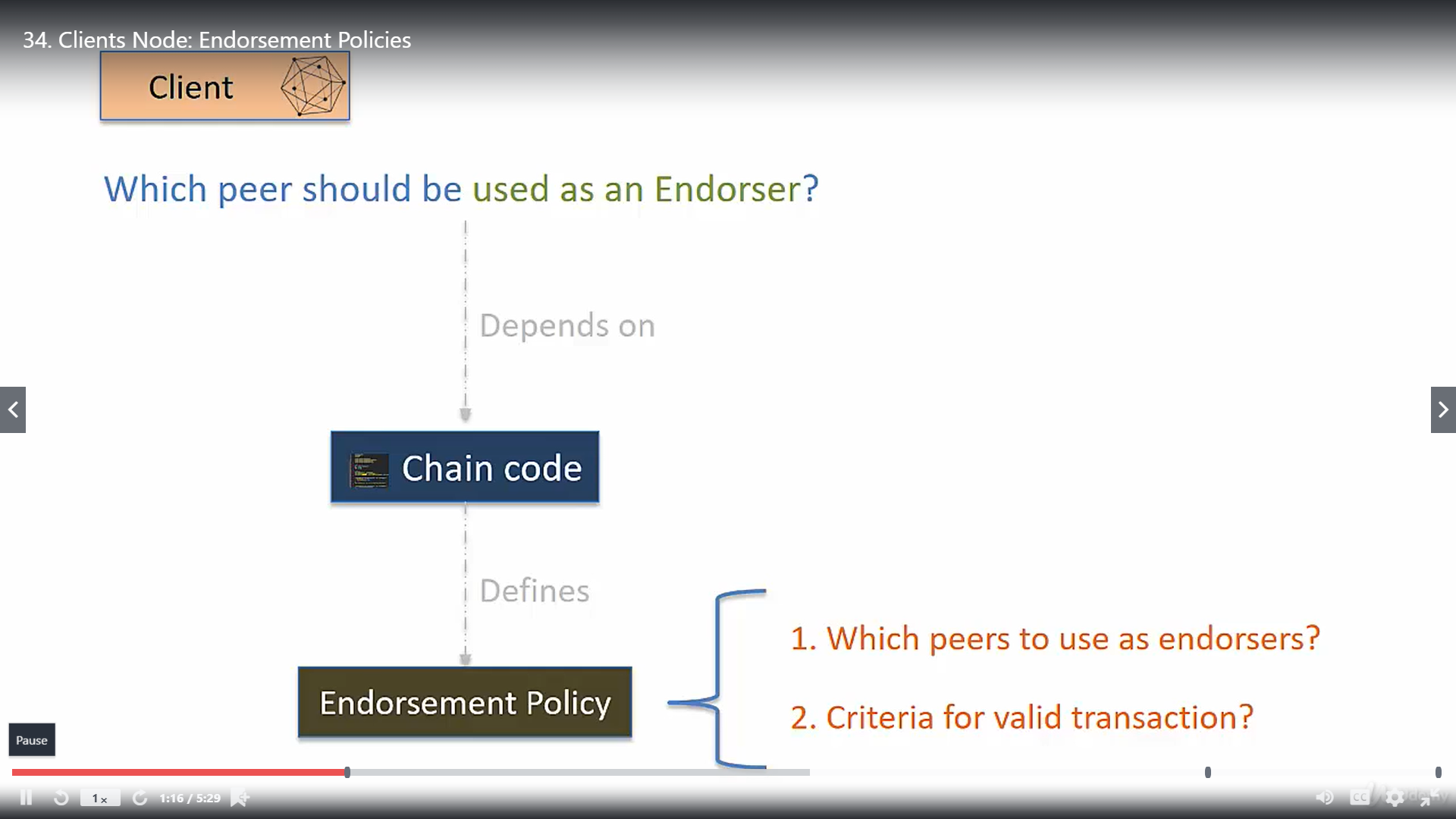


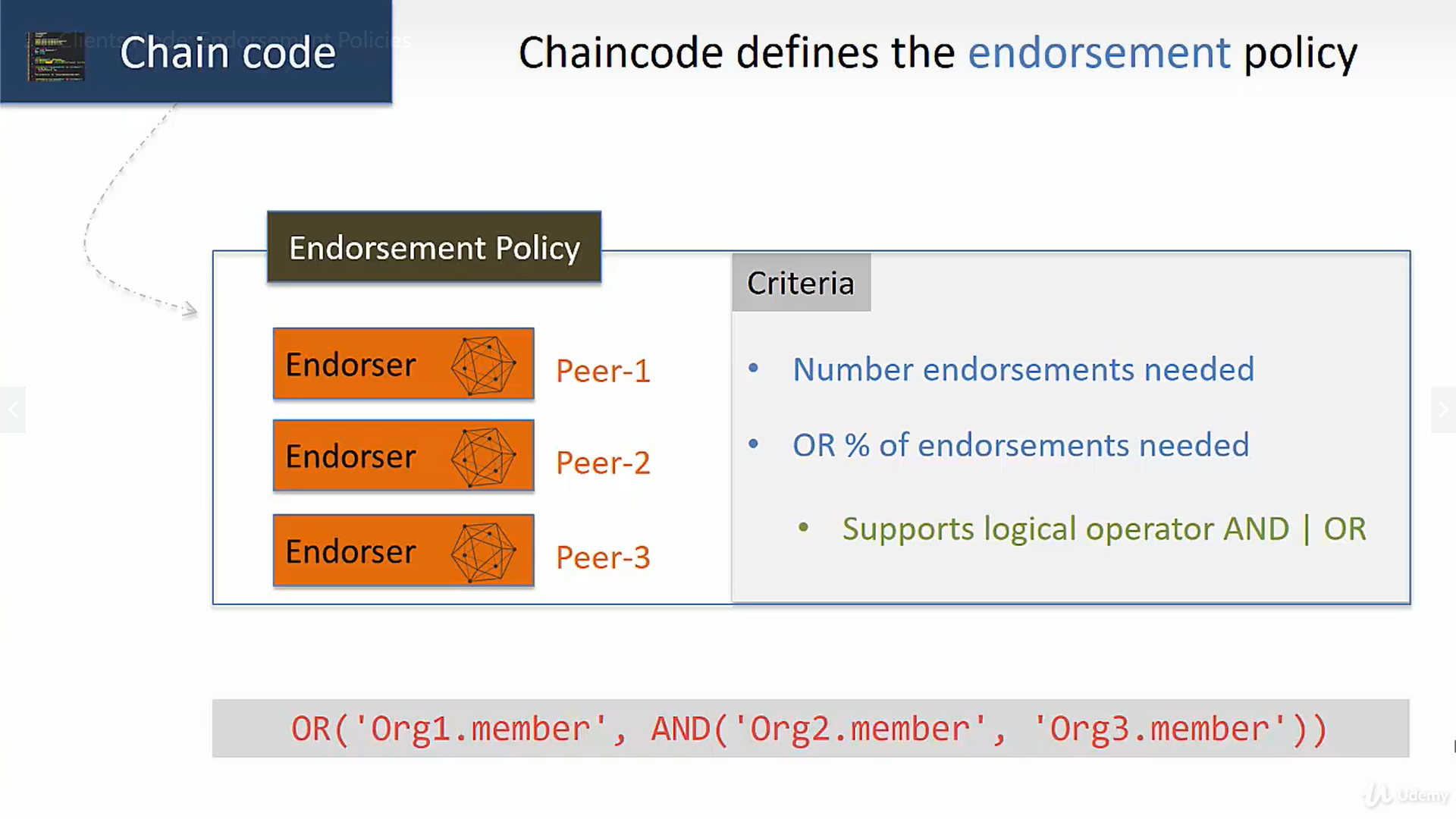


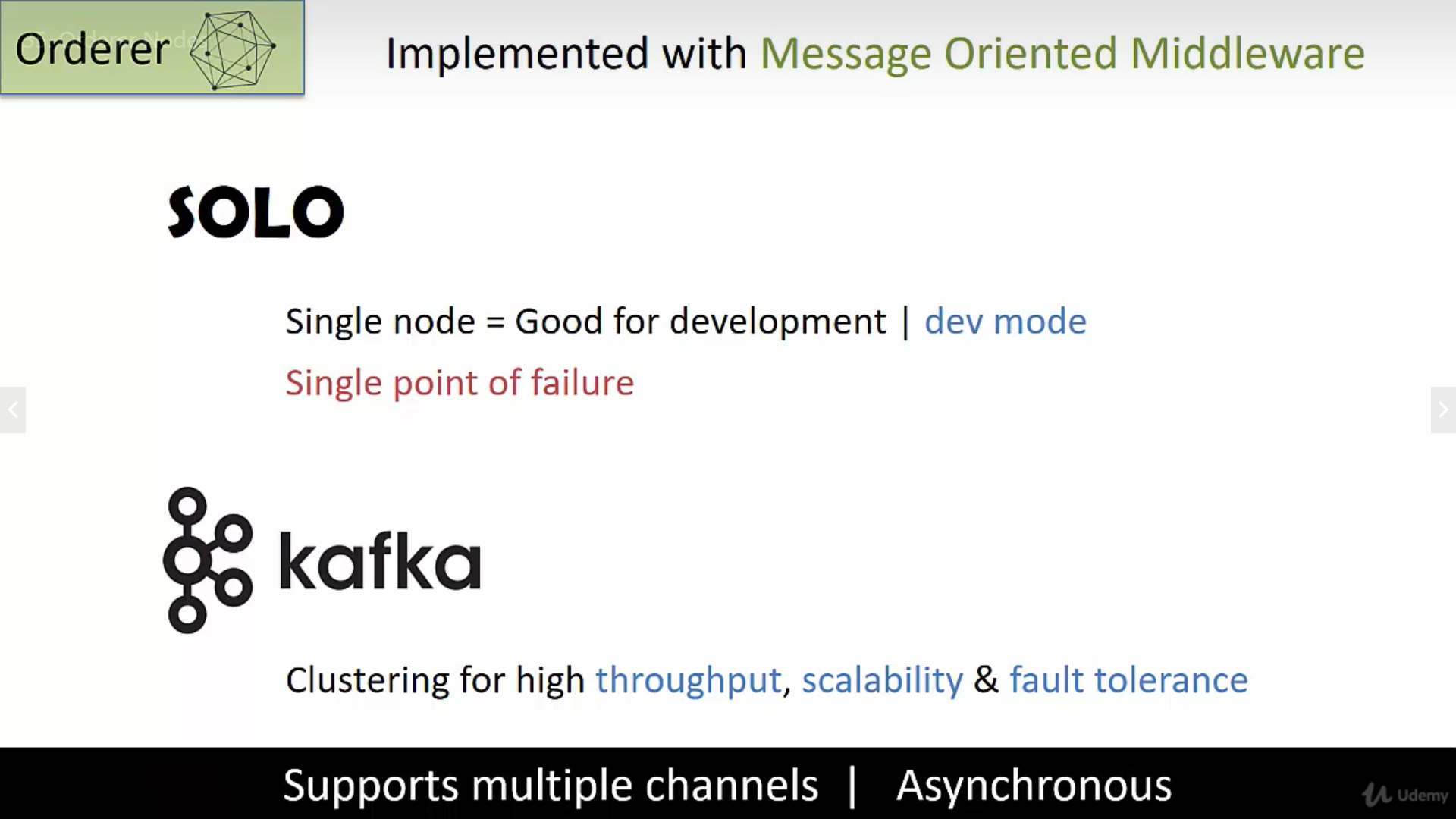












SOLO là một phần mềm trung gian nhắn tin chỉ được thiết lập với một node duy nhất. Không có tính năng phân cụm trong đó. Vì vậy, nó tốt cho sự phát triển nhưng sau đó nó giới thiệu một điểm thất bại duy nhất trong mạng của bạn. Rõ ràng có một trường hợp duy nhất trong môi trường sản xuất sẽ không được chấp nhận vì thất bại trong trường hợp duy nhất của Orderer sẽ đưa xuống toàn bộ mạng.

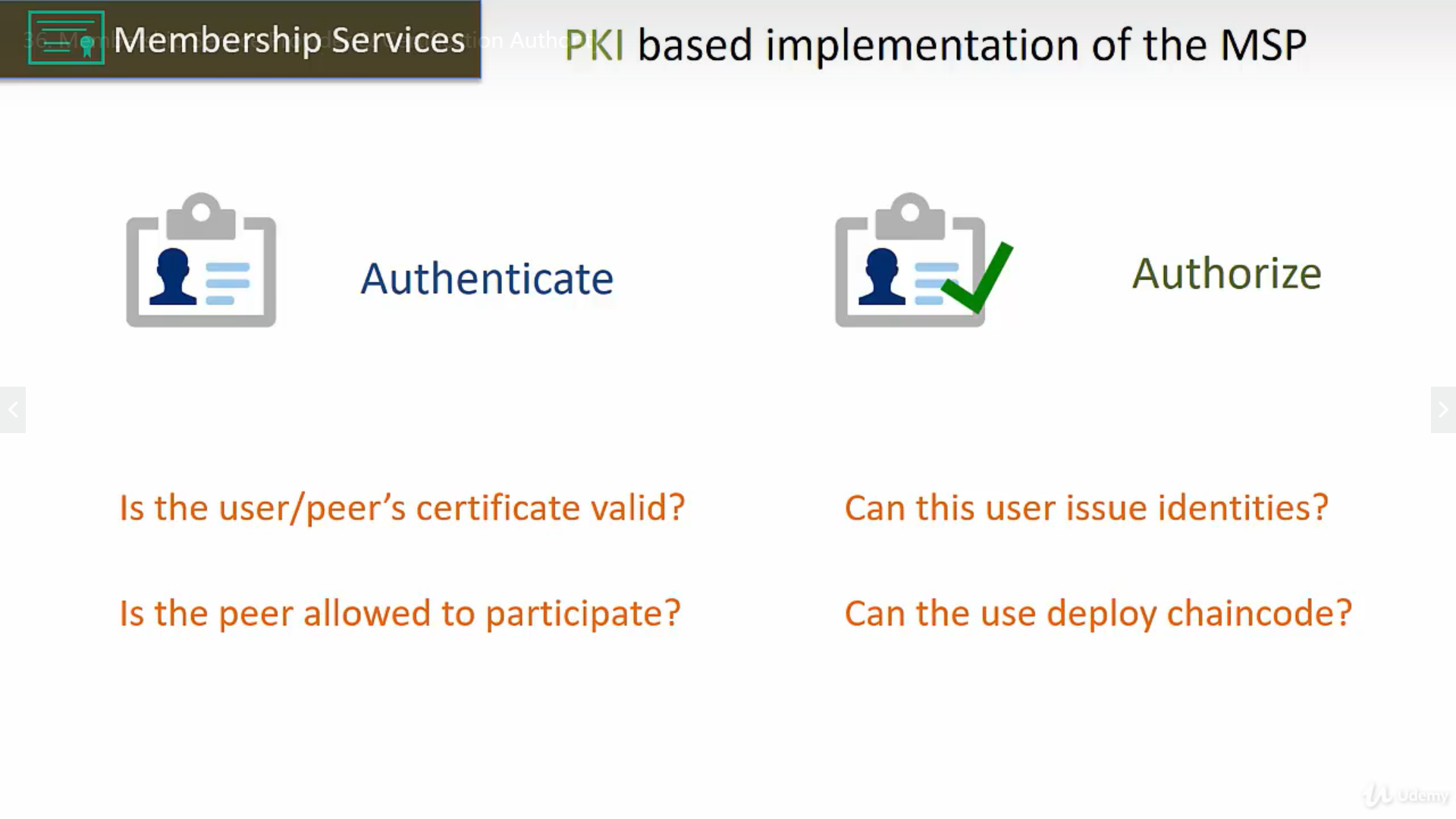
Trong sản xuất bạn sẽ không sử dụng SOLO. Phiên bản sản xuất của HF sẽ sử dụng Kafka. Kafka là một phần mềm nhắn tin có tính năng thông lượng cao, khả năng mở rộng cao và nó có khả năng chịu lỗi bằng cách phân cụm. Cả Kafka và SOLO đều hỗ trợ nhiều kênh vì vậy ngay cả khi bạn đang thử nghiệm với SOLO, bạn sẽ có thể kiểm tra các cấu hình khác nhau với nhiều kênh.

# **Membership service Provider & CA**

Ý tưởng trừu tượng là việc thực hiện xen kẽ MSP có thể được đặt vào mà không ảnh hưởng đến logic cốt lõi hoặc các thành phần nền tảng của mạng HF

Việc triển khai MSP mặc định dựa trên cơ sở hạ tầng khóa công khai (PKI).

Có hai dịch vụ chính mà MSP cung cấp: dịch vụ xác thực và dịch vụ ủy quyền.



# **Composer-Playground**

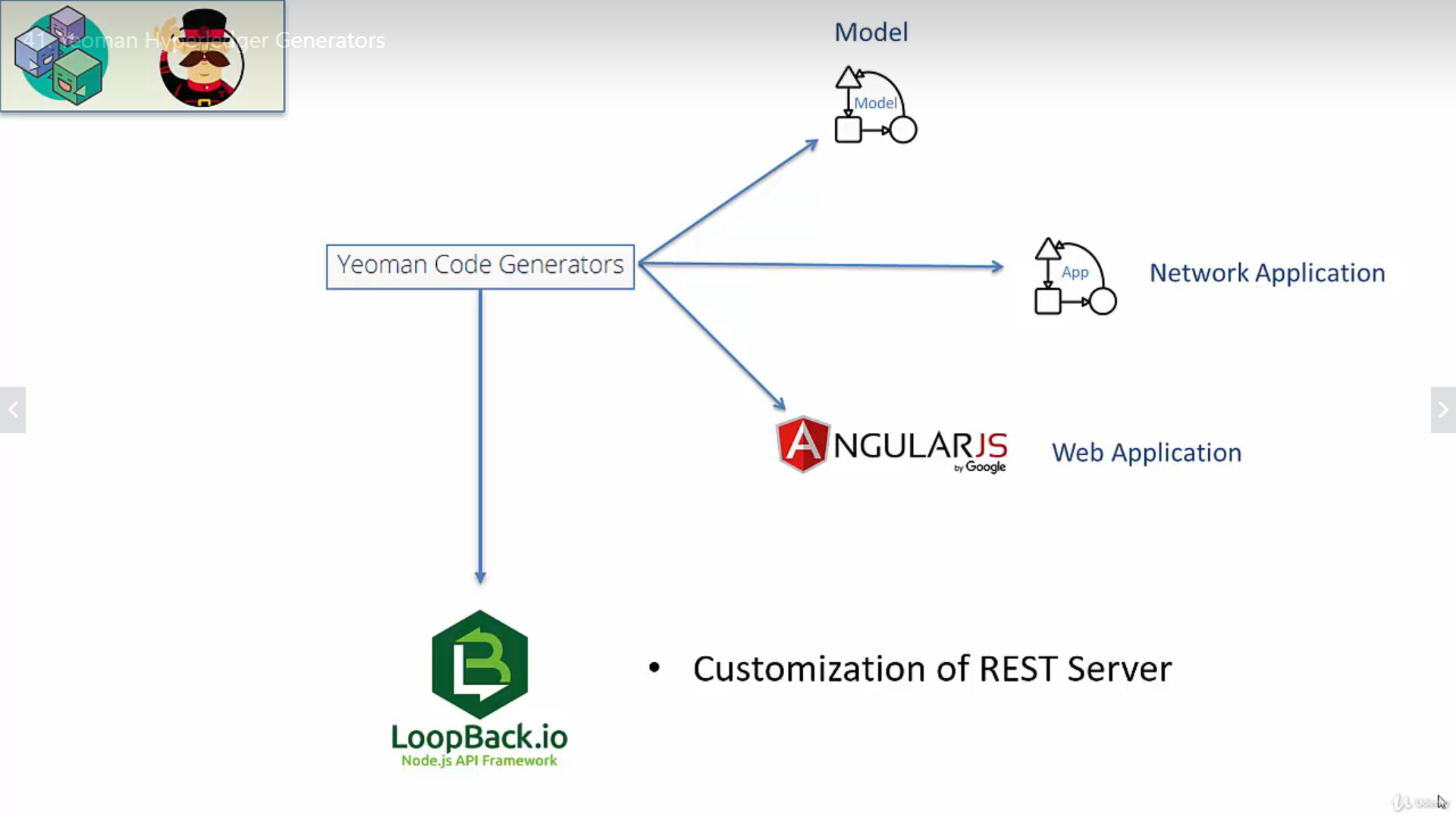
Là 1 sandbox web để bạn có thể deloy, edit và test các định nghĩa business network

# **Yeoman**

-Là một hệ thống giàn giáo chung cho phép tạo bất kỳ loại ứng dụng

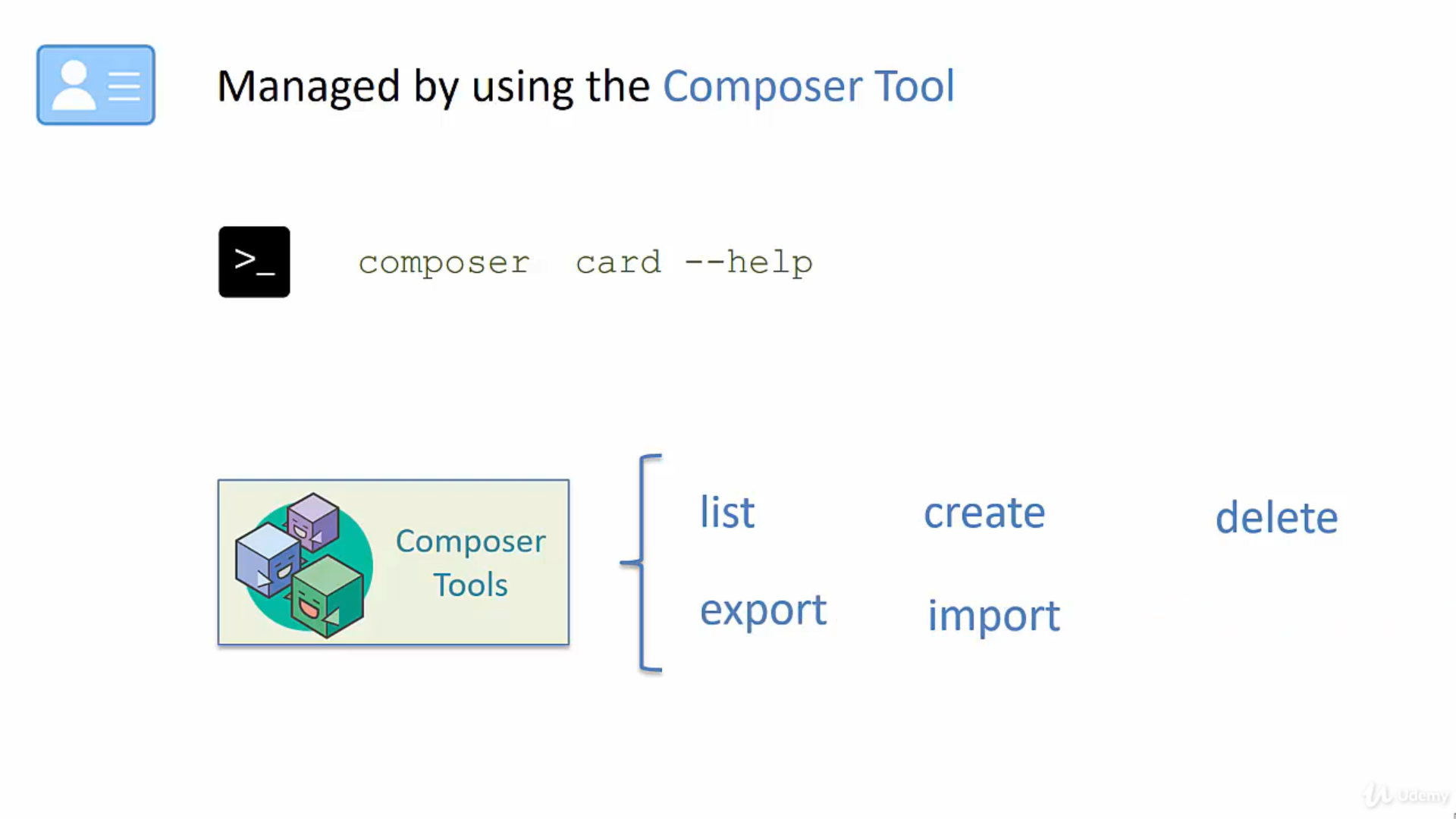
*npm install –g yo*

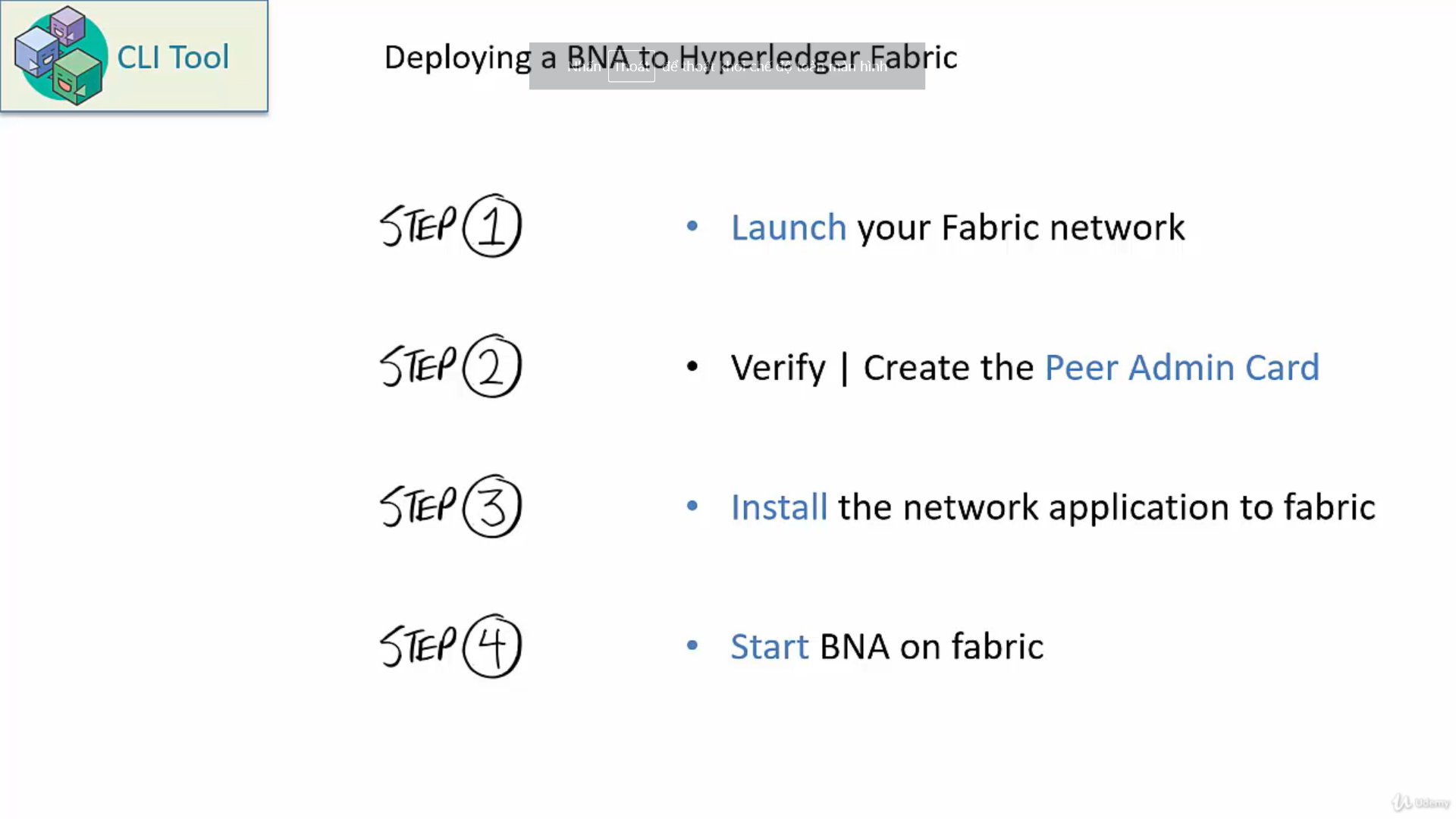
*Yo hyperledger-composer*



*~/.composer*







# **DataTypes**

