



Samsung Innovation Campus

| Internet kết nối vạn vật – IoT

Together for Tomorrow!
Enabling People
Education for Future Generations

Chương 8.

Nền tảng IoT và Trực quan hóa dữ liệu

Internet kết nối vạn vật – IoT

Mô tả chương



Mục tiêu

- ✓ Giới thiệu nền tảng IoT qua một ví dụ tiêu biểu về chuẩn oneM2M.
- ✓ Tìm hiểu về các tính năng và chức năng.
- ✓ Sử dụng các nền tảng mã nguồn mở và ứng dụng.



Nội dung

- ✓ Bài 1: Sử dụng nền tảng IoT
- ✓ Bài 2: Cài đặt và vận hành nền tảng mở IoT Mobius
- ✓ Bài 3: OpenHAB (Open Automation Bus)
- ✓ Bài 4: Trực quan hóa dữ liệu

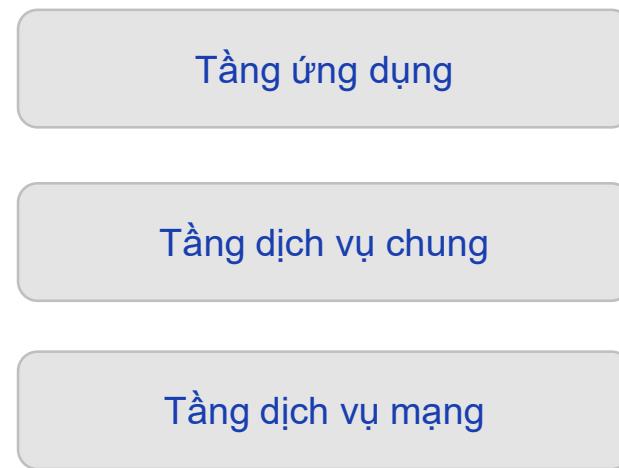
Chương 1.

Sử dụng nền tảng IoT

- | 1.1. Kiến trúc nền tảng oneM2M
- | 1.2. Nền tảng mã nguồn mở IoT: OCEAN

Nền tảng oneM2M là tầng dịch vụ chung

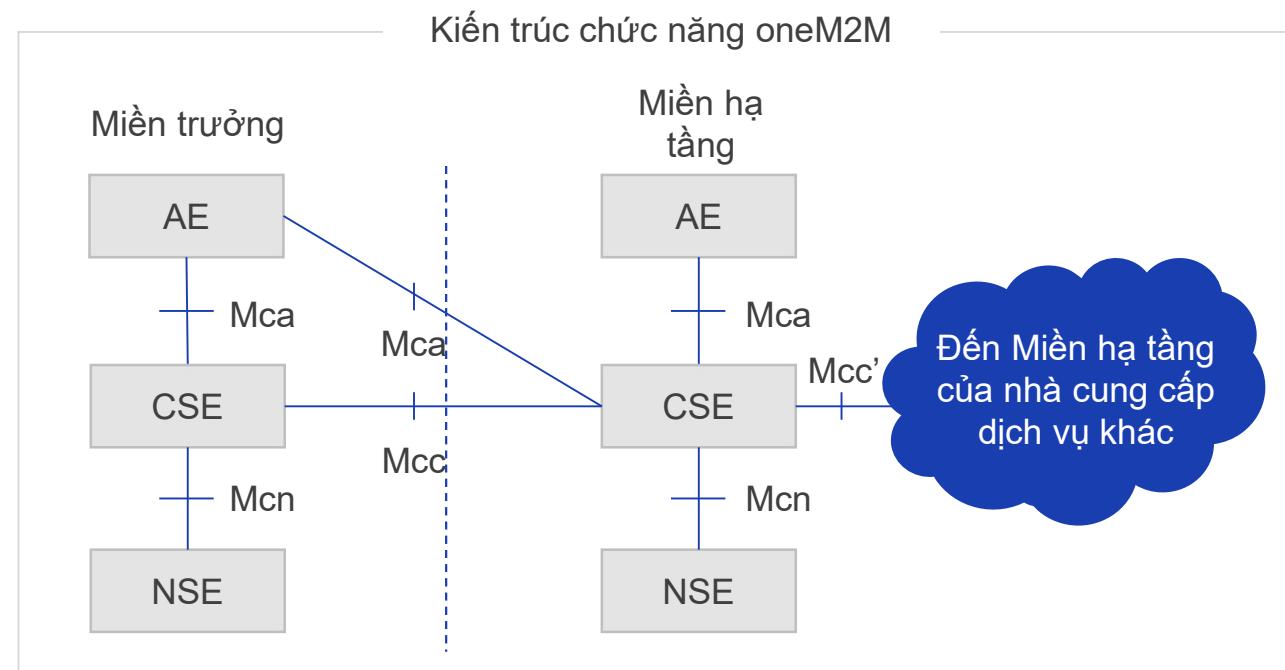
- | Nền tảng cung cấp chức năng phổ biến cần thiết cho các ứng dụng IoT
- | Nhà phát triển ứng dụng không cần tự triển khai các API dịch vụ mạng



Kiến trúc chức năng

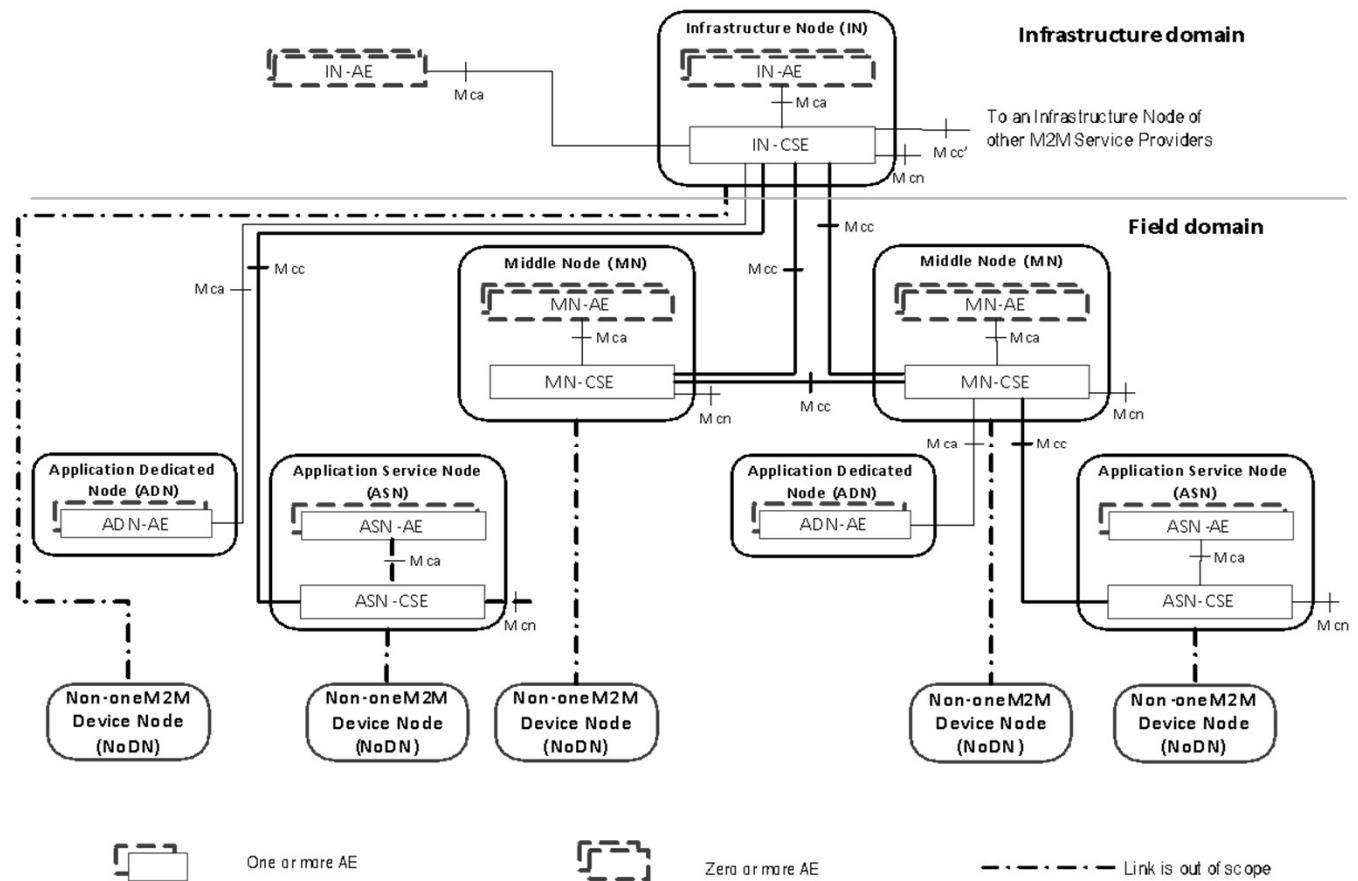
Hai miền trong hệ thống oneM2M

- Miền hạ tầng: máy chủ và ứng dụng oneM2M
- Miền trường: thiết bị oneM2M, cổng gateway và ứng dụng



Cấu hình hệ thống

- Nút hạ tầng
- Nút mạng giữa
- Nút dịch vụ ứng dụng
- Nút ứng dụng dùng riêng



Cấu hình được Kiến trúc oneM2M hỗ trợ

Mã định danh oneM2M

| Thực thể liên quan

- Mã định danh nhà cung cấp dịch vụ M2M
- Mã định danh thực thể ứng dụng
- Mã định danh ứng dụng
- Mã định danh CSE
- Mã định danh nút M2M
- Mã định danh bên ngoài
- Mã định danh mạng cơ sở
- Mã định danh người nhận kích hoạt

| Dịch vụ liên quan

- Mã định danh đăng ký M2M
- Mã định danh cấu hình dịch vụ M2M

| V.v.

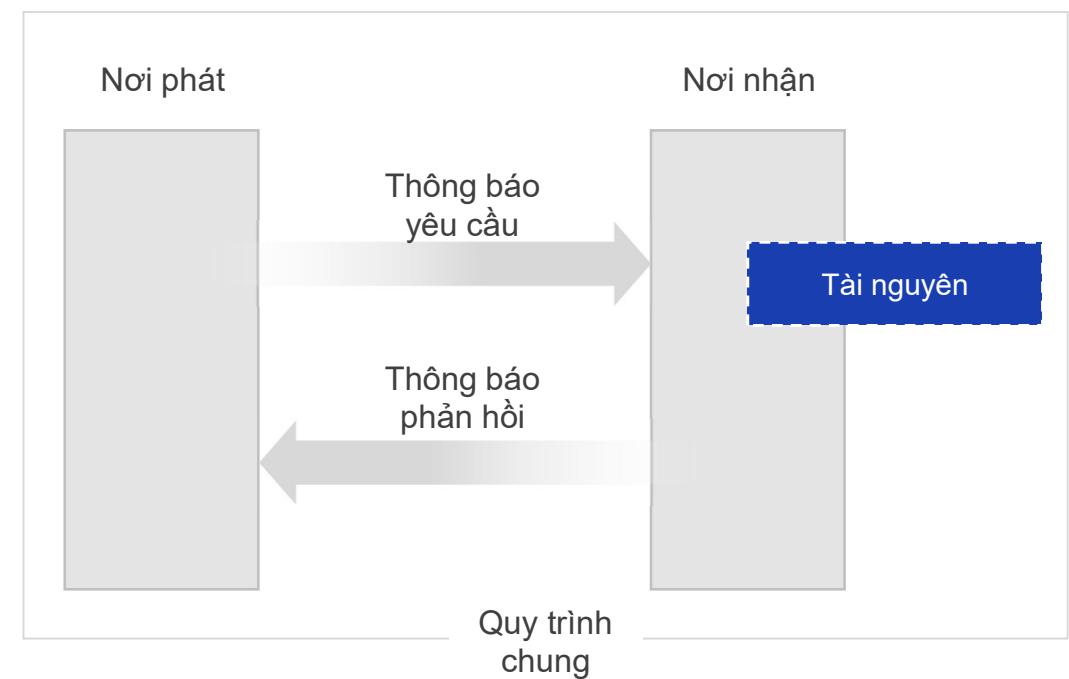
Quy trình chung

Cập Yêu cầu/Phản hồi

- Nơi phát gửi yêu cầu nhắm đến tài nguyên (thông thường)
- Nơi nhận luôn gửi lại phản hồi

Nơi phát tạo yêu cầu

- Tạo, Truy xuất, Cập nhật hoặc Xóa tài nguyên
- Thông báo thông tin



Mã định danh thực tế

| Thông tin về nơi phát yêu cầu

- có trong tham số From
- là AE-ID hoặc CSE-ID
 - AE hoặc CSE có thể là Nơi phát
- mã định danh có thể ở định dạng tuyệt đối, liên quan đến SP hoặc liên quan đến CSE
 - thông tin giống nhau có thể có định dạng khác nhau
 - định dạng tuyệt đối chứa SP-ID (ví dụ: //telecomABC/CSE12/CAE34)
 - định dạng liên quan đến SP chứa CSE-ID (ví dụ: /CSE12/CAE34)

Mã định danh tài nguyên

Thông tin về Mục tiêu của yêu cầu

- có trong tham số To
- mã định danh có thể ở định dạng tuyệt đối, liên quan đến SP hoặc liên quan đến CSE
- Về bản chất, mục tiêu là tài nguyên của REST API
- Tài nguyên được lưu trữ trên CSE gọi là CSE lưu trữ



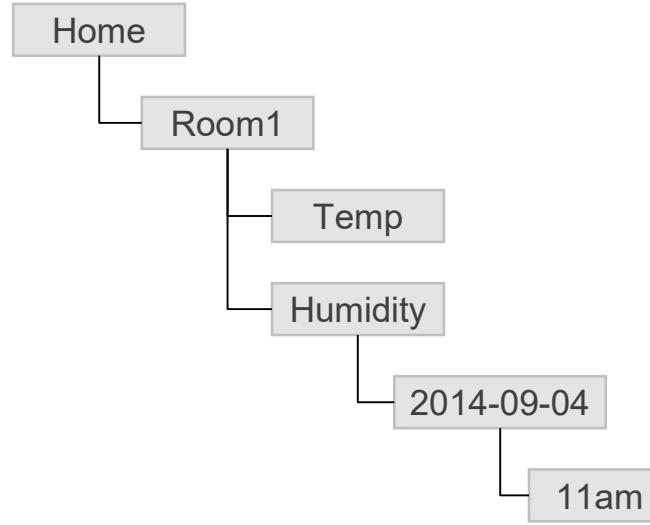
Sơ đồ định địa chỉ

| Phương pháp định địa chỉ phân cấp

- Mã định danh tài nguyên có cấu trúc (địa chỉ)

| Phương pháp định địa chỉ không phân cấp

- Mã định danh tài nguyên phi cấu trúc (địa chỉ)

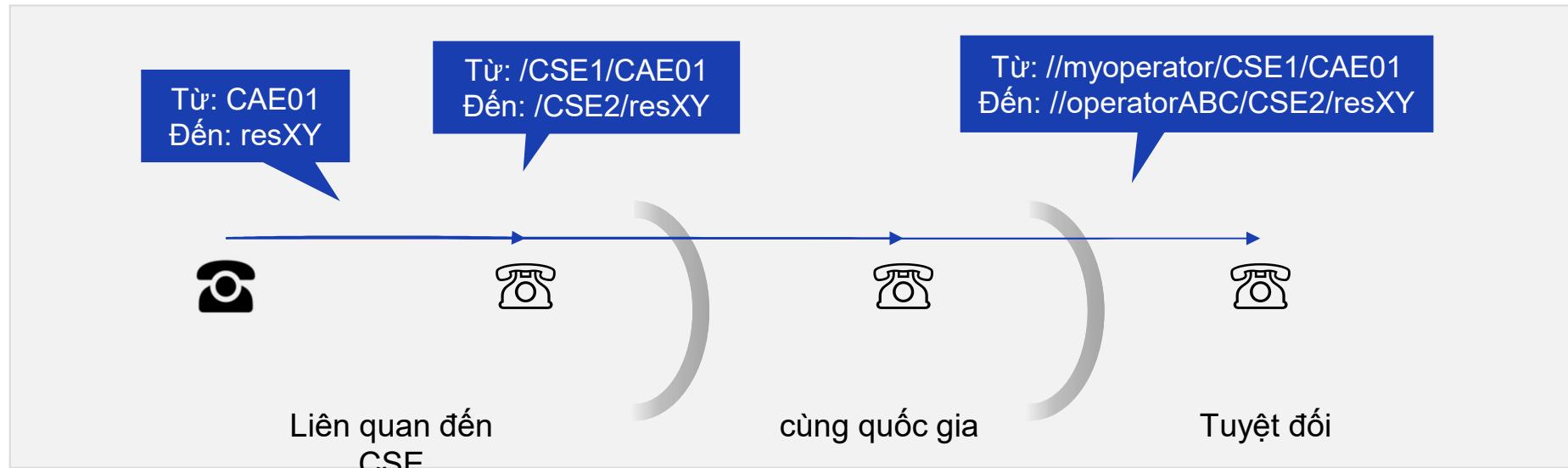


Có cấu trúc → Home/Room1/Humidity/2014-09-04/11am
Phi cấu trúc → inst1bc3d

Chọn định dạng mã định danh

Định dạng mã định danh phụ thuộc vào vị trí của Nơi phát và Nơi nhận

- Nếu là Tổ chức đăng ký tên miền và Cơ quan đăng ký → định dạng liên quan đến CSE
- Nếu cùng miền SP → định dạng liên quan đến SP
- Nếu không cùng miền SP → định dạng tuyệt đối

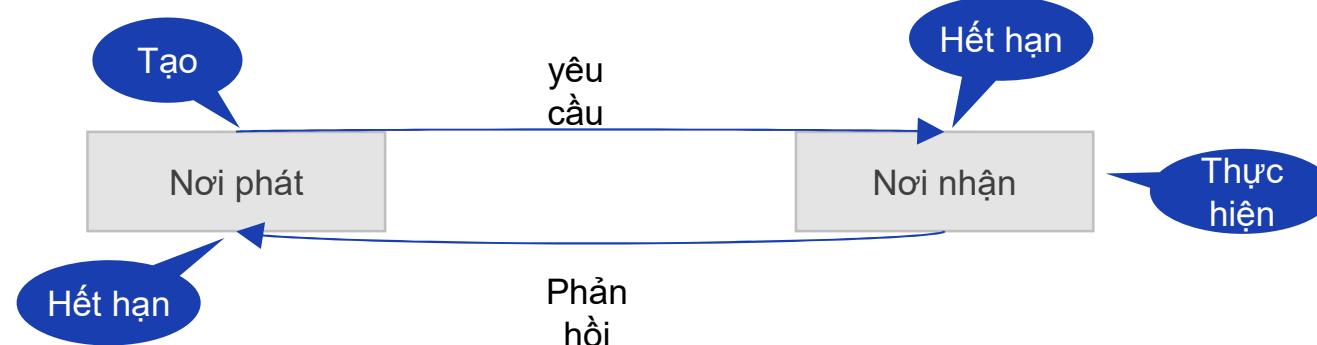


Tham số yêu cầu (1/3)

- | Từ
- | Đến
- | Tác vụ
- | Mã định danh yêu cầu
- | Nội dung
- | Kiểu tài nguyên
- | Kiểu phản hồi

Tham số yêu cầu (2/3)

- █ Nhãn thời gian gốc
- █ Nhãn thời gian hết hạn yêu cầu
- █ Nhãn thời gian hết hạn kết quả
- █ Thời gian thực hiện tác vụ



Tham số yêu cầu (3/3)

Tiêu chí lọc

- dùng làm bộ lọc tìm kiếm trong quá trình Tìm kiếm
 - chỉ trả lại kết quả những tài nguyên phù hợp
- Truy xuất, Cập nhật, Xóa theo điều kiện
 - chỉ khi đáp ứng tiêu chí lọc, nơi nhận mới thực hiện yêu cầu
- có điều kiện lọc
 - createdBefore, createdAfter, modifiedSince, unmodifiedSince, stateTagSmaller, stateTagBigger, expireBefore, expireAfter, sizeAbove, sizeBelow, labels, resourceType, contentType, attribute, filterUsage, filterOperation, limit, level, offset, semanticsFilter, contentFilterSyntax, contentFilterQuery

Nội dung kết quả

- cho biết kiểu kết quả yêu cầu có trong tham số Nội dung của thông báo phản hồi

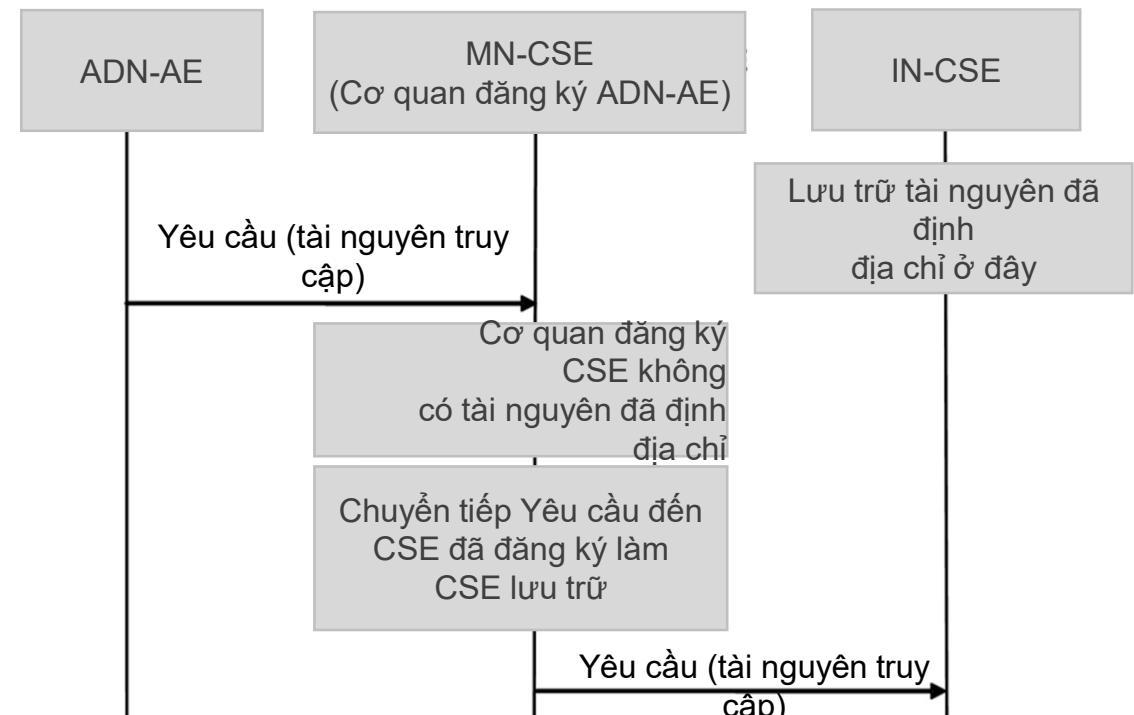
Xử lý tham số yêu cầu

Xử lý bằng CSE chuyển tiếp

- chuyển tiếp yêu cầu, xử lý yêu cầu hết hạn

Xử lý bằng CSE lưu trữ

- xử lý tác vụ theo yêu cầu (CRUDN)



AE/CSE truy cập tài nguyên tại CSE lưu trữ (Một bước nhảy)

Tham số phản hồi

Mã trạng thái phản hồi

- ▶ cho biết quá trình xử lý yêu cầu thành công, không thành công hay đã xác nhận
 - khi xử lý không thành công sẽ chỉ ra mã lỗi cụ thể
 - mã xác nhận được dùng cho chế độ đồng bộ không tuần tự hoặc không đồng bộ

Mã định danh yêu cầu

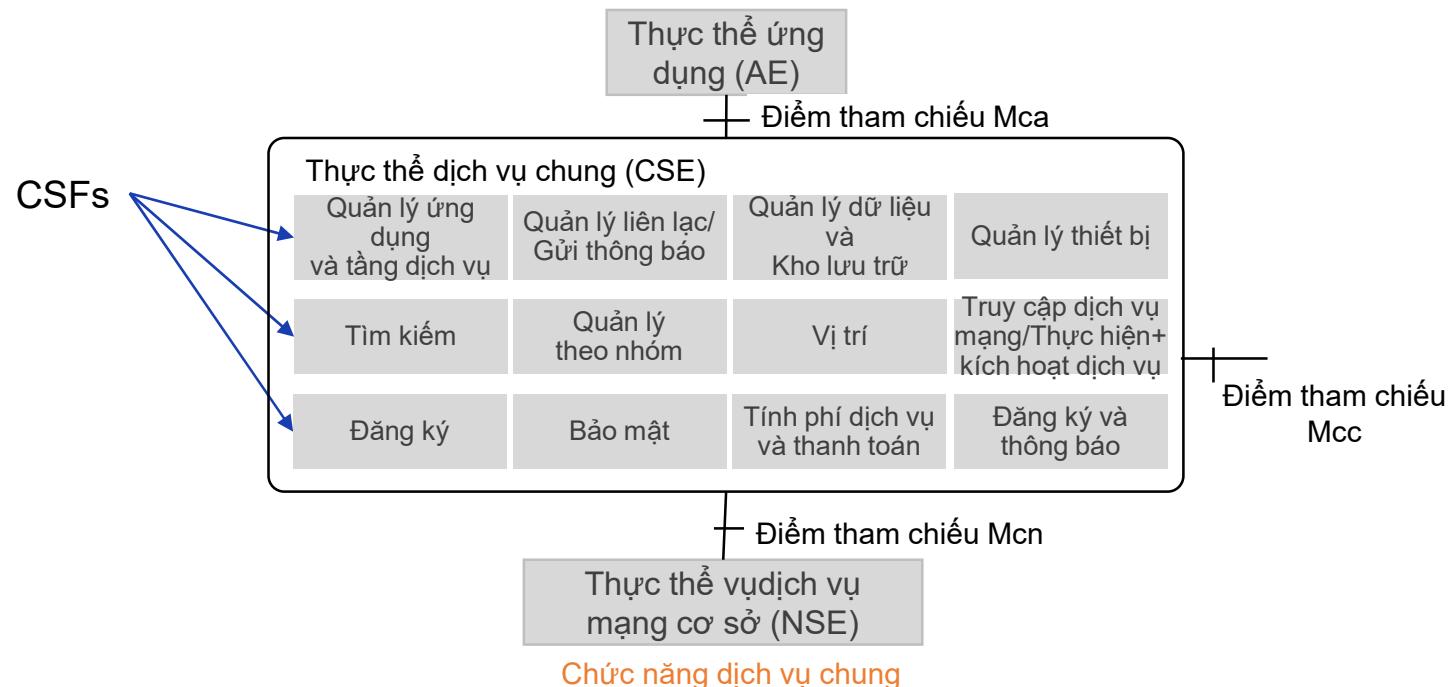
- ▶ cùng mã định danh sao chép từ thông báo yêu cầu

Nội dung

- ▶ phần đầu/phần thân thông báo phản hồi

Nền tảng oneM2M có chức năng như CSF

- | Các chức năng ứng dụng thường dùng cho dịch vụ IoT
- | nhà phát triển ứng dụng không phải triển khai và thay vào đó có thể sử dụng các chức năng phổ biến của oneM2M (API)
- | Ứng dụng có thể tương thích khi sử dụng cùng các chức năng phổ biến



Mô tả CSF

Đăng ký

- Quy trình bắt buộc/ban đầu trước khi sử dụng nền tảng

Quản lý dữ liệu và lưu trữ

- Lưu trữ và chia sẻ dữ liệu

Quản lý thiết bị (DM)

- Quản lý thiết bị bằng các giao thức quản lý thiết bị hiện có (ví dụ: OMA DM)

Quản lý ứng dụng và tầng dịch vụ

- Quản lý và định cấu hình phần mềm

Quản lý liên lạc và gửi thông báo

- Quản lý mạng lưới và gửi thông báo giữa các thực thể oneM2M

Danh sách CSF (tiếp)

Tìm kiếm

- Tìm kiếm tài nguyên trong CSE

Vị trí

- Tạo và chia sẻ thông tin vị trí

Quản lý theo nhóm

- Quản lý tài nguyên (dữ liệu, dịch vụ) theo nhóm, phân tách và tổng hợp tài nguyên

Đăng ký/Thông báo

- Thiết lập cấu hình đăng ký tài nguyên và thông báo sự kiện (dạng đầy)

Bảo mật

- Cấp phép, xác thực và ủy quyền

Truy cập dịch vụ mạng

- Sử dụng dịch vụ mạng (ví dụ: cấu hình mẫu lưu lượng mạng 3GPP)

Tính phí dịch vụ và thanh toán

- Ghi lại nhật ký sử dụng và tính phí dịch vụ

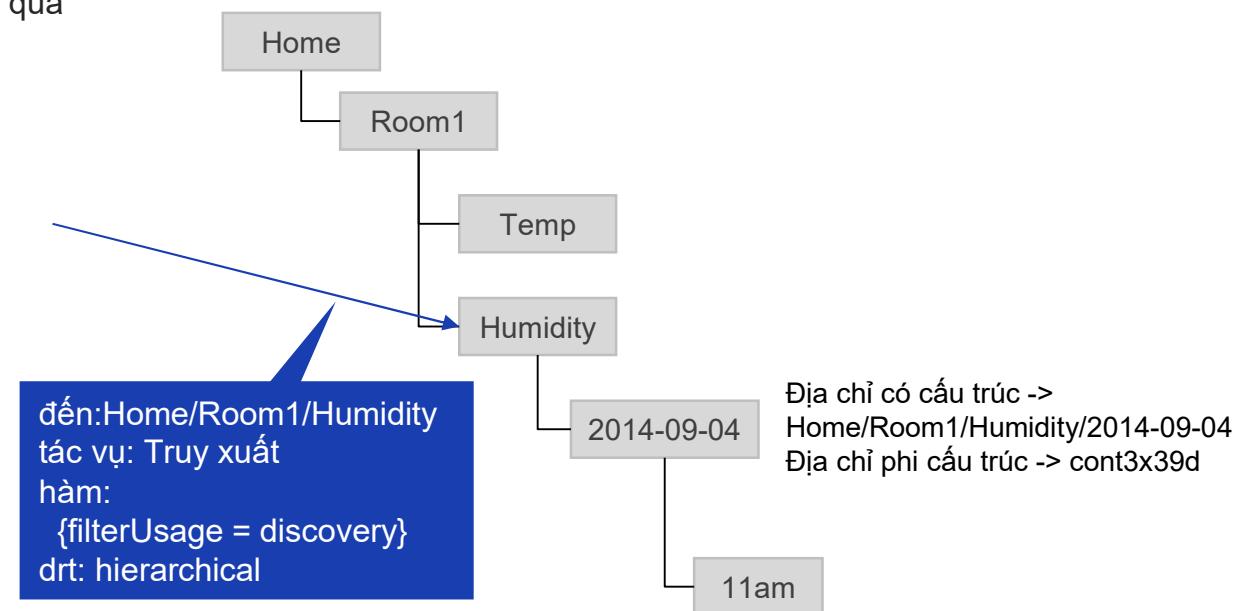
Khám phá

Tham số kiểu kết quả khám phá

- Cho biết kết quả chứa địa chỉ có cấu trúc hay phi cấu trúc

Phạm vi khám phá

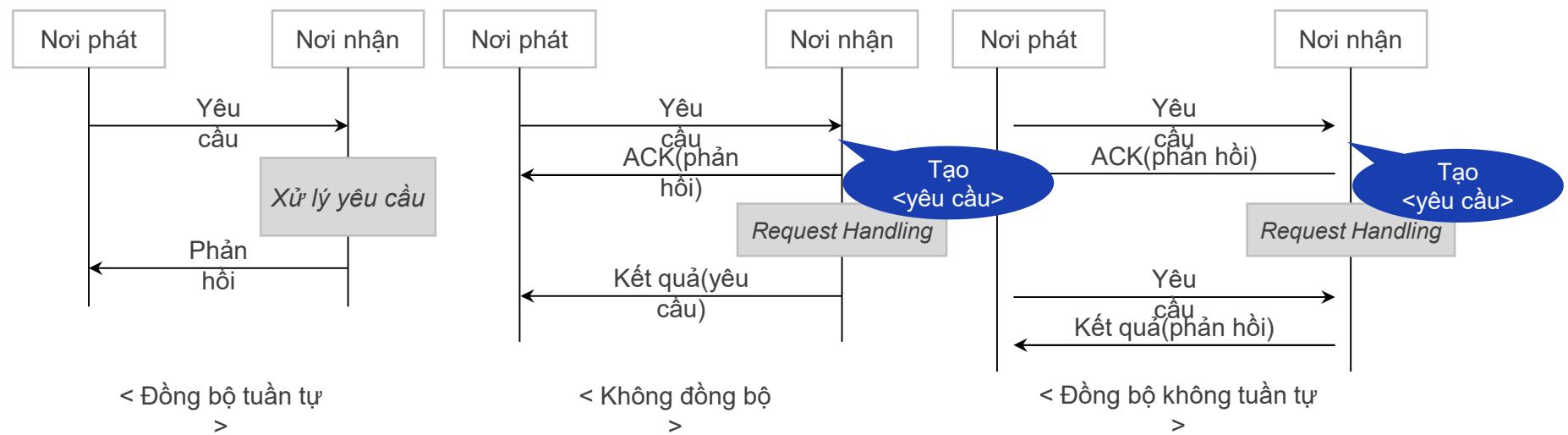
- Tài nguyên phụ (con) của tài nguyên đích
 - Tài nguyên đích không có trong kết quả



Chế độ truyền thông

Tham số kiểu phản hồi

- Nơi phát gọi ý chế độ **truyền thông**
 - Chế độ đồng bộ tuần tự, đồng bộ không tuần tự, không đồng bộ, đồng bộ tuần tự linh hoạt
 - Trong trường hợp không tuần tự, tạo tài nguyên <request>
 - Trong trường hợp tuần tự linh hoạt, Nơi nhận chọn chế độ thông tin



Tham số nội dung kết quả

- █ Không có kết quả → CUDN
- █ Thuộc tính → CRUD
- █ địa chỉ phân cấp
- █ địa chỉ phân cấp+thuộc tính
- █ thuộc tính+tài nguyên con
- █ tài nguyên con
- █ Thuộc tính+tham chiếu tài nguyên con
- █ tham chiếu tài nguyên con
- █ tài nguyên gốc

} Chỉ
tạo

} Chỉ
xuất

Tổng hợp giá trị của nội dung kết quả

Giá trị	Tạo	Truy xuất	Cập nhật	Xóa	Thông báo
thuộc tính	mặc định	mặc định	mặc định	Có giá trị	Không xác định
thuộc tính đã chỉnh sửa	Có giá trị	Không xác định	Có giá trị	Không xác định	Không xác định
địa chỉ phân cấp	Có giá trị	Không xác định	Không xác định	Không xác định	Không xác định
địa chỉ phân cấp+thuộc tính	Có giá trị	Không xác định	Không xác định	Không xác định	Không xác định
Thuộc tính+tài nguyên con	Không xác định	Có giá trị	Không xác định	Không xác định	Không xác định
Tài nguyên con	Không xác định	Có giá trị	Không xác định	Không xác định	Không xác định
Thuộc tính+tham chiếu tài nguyên con	Không xác định	Có giá trị	Không xác định	Không xác định	Không xác định
Tham chiếu tài nguyên con	Không xác định	Có giá trị	Không xác định	Không xác định	Không xác định
không có kết quả	Có giá trị	Không xác định	Có giá trị	default	Có giá trị
Tài nguyên gốc	Không xác định	Có giá trị	Không xác định	Không xác định	Không xác định

Nội dung phản hồi

- | Trong trường hợp yêu cầu Tạo, nếu Nội dung kết quả là “thuộc tính” thì phản hồi chứa địa chỉ không phân cấp
 - Để nhận được địa chỉ phân cấp, sử dụng một trong các giá trị sau
 - địa chỉ phân cấp
 - địa chỉ phân cấp+thuộc tính
- | Trong trường hợp yêu cầu Truy xuất, phản hồi có thể trả về các biểu diễn và/hoặc tham chiếu tài nguyên
 - Để chỉ ra kiểu thông tin của tài nguyên đích và/hoặc tài nguyên con, hãy sử dụng một trong các giá trị sau
 - thuộc tính+tài nguyên con
 - tài nguyên con
 - thuộc tính+tham chiếu tài nguyên con
 - tham chiếu tài nguyên con
 - “thuộc tính” là các thuộc tính (biểu diễn) của tài nguyên đích
 - “tài nguyên con” là các thuộc tính của tài nguyên con
 - “tham chiếu” là thông tin đặc tả chứa địa chỉ tài nguyên

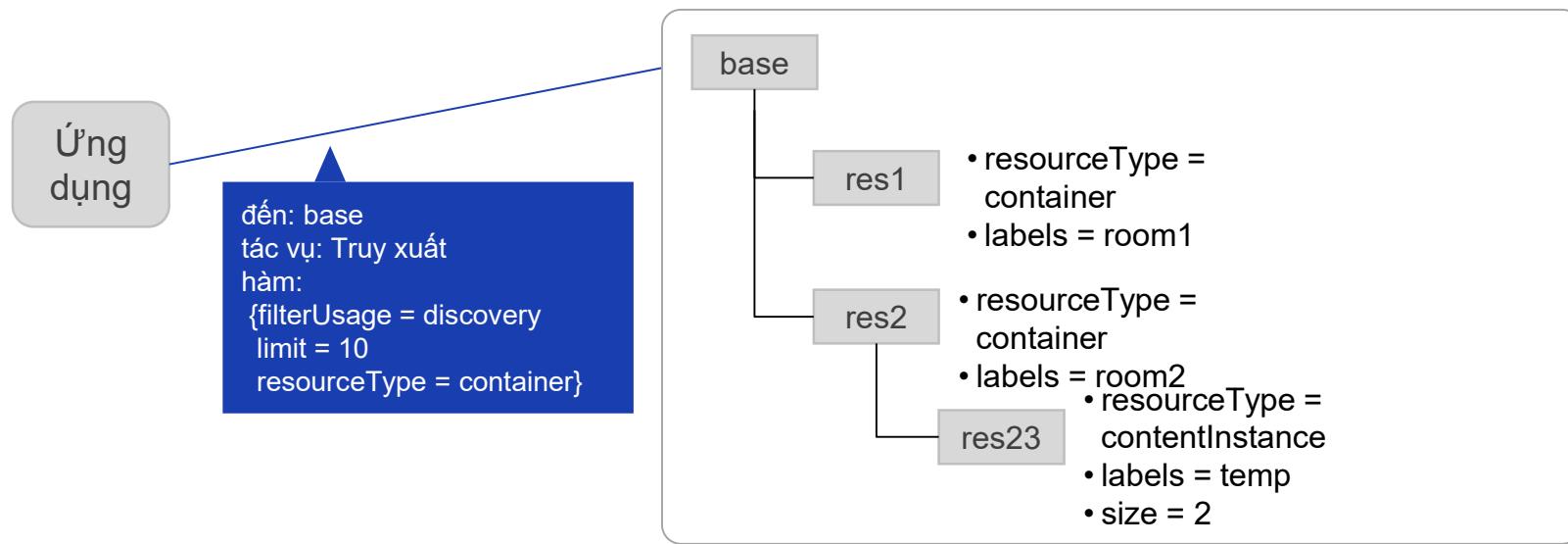
Lọc tài nguyên

Tiêu chí lọc

- áp dụng để khớp với giá trị thuộc tính của tài nguyên cụ thể
 - creationTime, lastModifiedTime, expirationTime, stateTag, label, attribute, resourceType, size, contentType

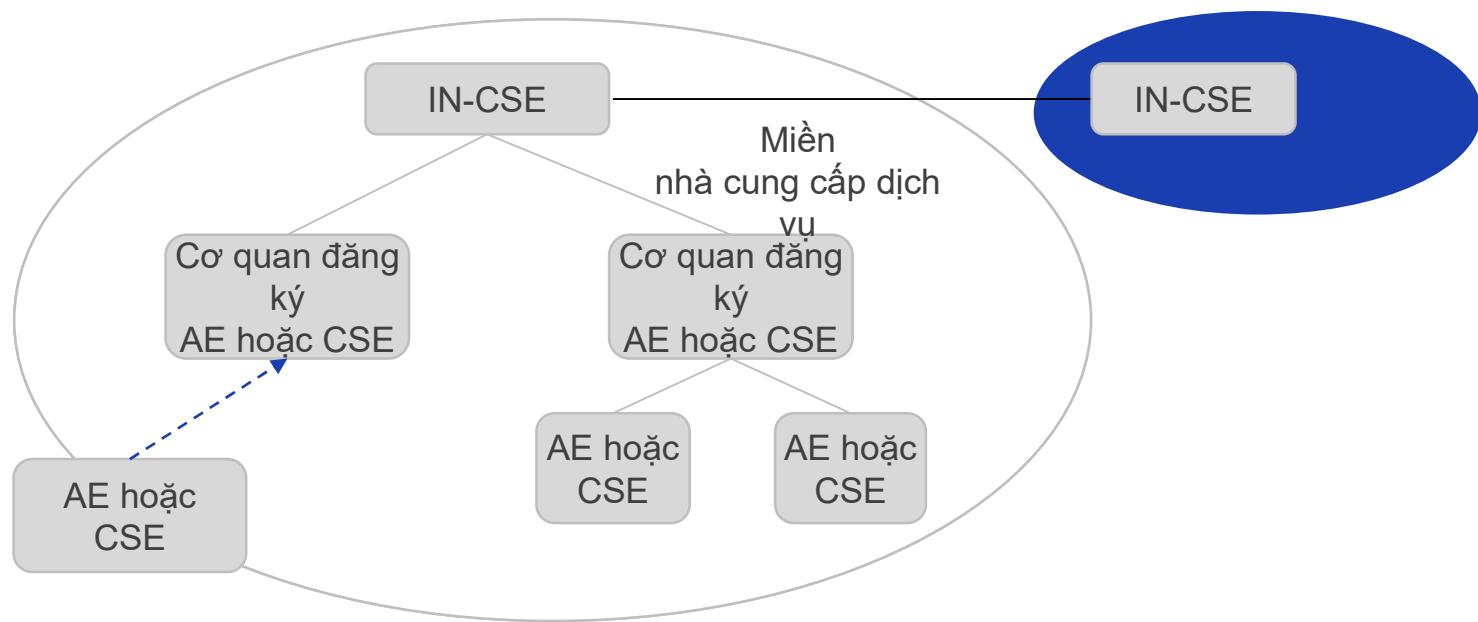
áp dụng cho quy trình lọc, không áp dụng cho giá trị thuộc tính

- filterUsage, filterOperation, limit, level, offset, semanticsFilter, contentFilterSyntax, contentFilterQuery



Đăng ký

- điều kiện tiên quyết để sử dụng hệ thống oneM2M xác nhận mã định danh thực thể bằng cách kiểm tra xác thực
- Tổ chức đăng ký tên miền gửi yêu cầu đăng ký đến Cơ quan đăng ký (CSE) được chỉ định trước



Đăng ký

Đăng ký AE

- AE sẽ được xác thực bằng thông tin đăng nhập bảo mật
- AE-ID thường do Cơ quan đăng ký CSE hoặc IN-CSE chỉ định
- API liên quan
 - Tài nguyên <AE>: Tạo/Truy xuất/Cập nhật/Xóa

Đăng ký CSE

- CSE được hệ thống tin tưởng hơn AE
- CSE-ID luôn được cấp trước
- API liên quan
 - Tài nguyên <remoteCSE>: Tạo/Truy xuất/Cập nhật/Xóa
 - Tài nguyên <CSEBase>: Truy xuất

Quản lý dữ liệu: kiểu tài nguyên container và contentInstance

- | vùng chứa có các vùng chứa phụ để biểu thị tính phân cấp của dữ liệu
- | vùng chứa chứa dữ liệu là phiên bản nội dung
 - có dữ liệu đặc tả cho tất cả các phiên bản trong vùng chứa
 - tối đa # phiên bản, hiện tại có # phiên bản, kích thước byte tối đa, kích thước byte hiện tại, thời gian tối đa của phiên bản
- | contentInstance
 - có dữ liệu đặc tả của từng phiên bản cũng như chính dữ liệu
 - kiểu phương tiện, bộ mã, kích thước, bộ tạo lập
- | Mới nhất và cũ nhất
 - là các con trỏ lần lượt đến tài nguyên <contentInstance> mới nhất và cũ nhất
- | API liên quan
 - Tài nguyên <container>: CRUD
 - Tài nguyên <contentInstance>: CRD
 - Tài nguyên <latest>, <oldest>: RD

Quản lý dữ liệu: kiểu tài nguyên flexContainer

- là biến thể của tài nguyên <container>, <contentInstance>
- là vùng chứa tùy chỉnh có thể có thuộc tính tùy chỉnh
 - do đó, có một số thuộc tính đặc tả của flexContainer
- đồng thời là vùng chứa và phiên bản dữ liệu
- API liên quan
 - Tài nguyên <flexContainer>: CRUD

Kiểu tài nguyên timeSeries và timeSeriesInstance

- là biến thể của tài nguyên <container>, <contentInstance>
- là thời gian cần tạo để ghi/xử lý dữ liệu, không phải thời gian lưu trữ dữ liệu
 - dataGenerationTime với creationTime
- API liên quan
 - Tài nguyên <timeSeries>: CRUD
 - Tài nguyên <timeSeriesInstance>: CRD

Kiểm soát truy cập: kiểm soát đặc quyền hoặc ủy quyền

- | Quyết định xem CSE với tư cách là nơi nhận có cho phép Nơi phát (CRUDN) yêu cầu truy cập tài nguyên trên CSE hay không
- | Kiểm tra ba phần thông tin bắt buộc:
 - Ai: Thông tin định danh Nơi phát (tham số From hoặc các tham số khác)
 - Cái gì: Đặc quyền truy cập tài nguyên đích (tham số To)
 - Làm gì: Thao tác (CRUDN) truy cập tài nguyên (tham số Thao tác)
- | Tập hợp các đặc quyền truy cập được định cấu hình trước trong hệ thống và sau đó được dùng để đưa ra quyết định sau này khi CSE nhận được yêu cầu

Điều khiển truy cập: Các cơ chế điều khiển truy cập khác nhau

Chính sách điều khiển truy cập

- Cơ chế cơ bản được hỗ trợ trong Rel-1 cũng như các bản phát hành khác
- Tài nguyên `<accessControlPolicy>` và thuộc tính `accessControlPolicyIDs`

Tư vấn ủy quyền động

- Tài nguyên `<dynamicAuthorizationConsultation>` và thuộc tính `dynamicAuthorizationConsultationIDs`

Vai trò

- Tài nguyên `<role>` và tham số Mã vai trò

Token

- Tài nguyên `<token>` và các tham số Chỉ báo yêu cầu token/TOKEN/Mã token

Kiểu tài nguyên Group và fanOutPoint

chức năng nhóm bao gồm

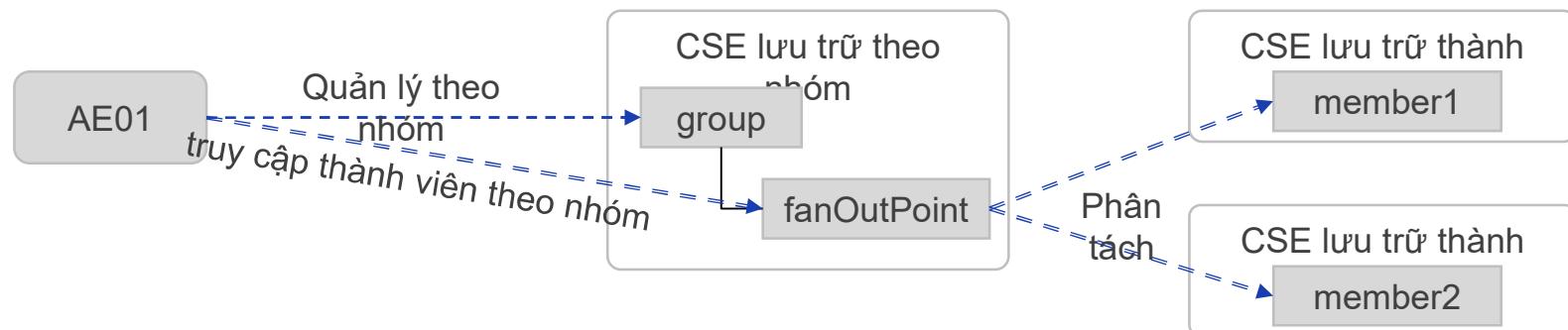
- quản lý thành viên theo nhóm
- phân tách yêu cầu và tổng hợp phản hồi

Tài nguyên <group> chứa

- danh sách thành viên và liên kết chính sách điều khiển truy cập để truy cập các thành viên, không phải tài nguyên nhóm
- điểm phân tách để nhận yêu cầu cần gửi tới tất cả các thành viên

API liên quan

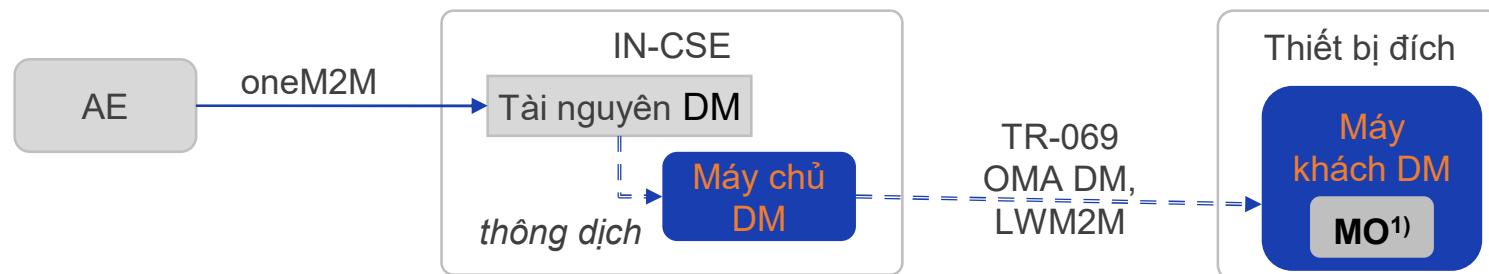
- Tài nguyên <group>, <fanOutPoint>: CRUD



Quản lý thiết bị (DM)

kiểu tài nguyên mgmtObj, mgmtCmd và execInstance

- Chức năng quản lý thiết bị của oneM2M sử dụng các giao thức quản lý thiết bị bên ngoài
 - BBF TR-069, OMA DM và Lightweight M2M
- và thông dịch giao thức oneM2M cho những thiết bị này
 - thao tác tài nguyên kích hoạt lệnh gọi API quản lý thiết bị
- API liên quan
 - Tài nguyên <mgmtObj>, <mgmtCmd>: CRUD
 - Tài nguyên <execInstance>: RUD



Kiểu tài nguyên LocationPolicy

| CSE cung cấp vị trí thực thể theo các cơ chế sau:

- Theo thiết bị (ví dụ: GPS)
- Theo mạng (ví dụ: máy chủ vị trí của người vận hành)
- Theo hoạt động chia sẻ

| chính sách vị trí lưu trữ cấu hình để nhận thông tin vị trí

- Cập nhật thời hạn, ID đích, máy chủ vị trí, khu vực đích, v.v.

| khi thiết lập chính sách sẽ tạo vùng chứa dữ liệu (tài nguyên <container>) đồng thời liên kết chính sách và vùng chứa với nhau

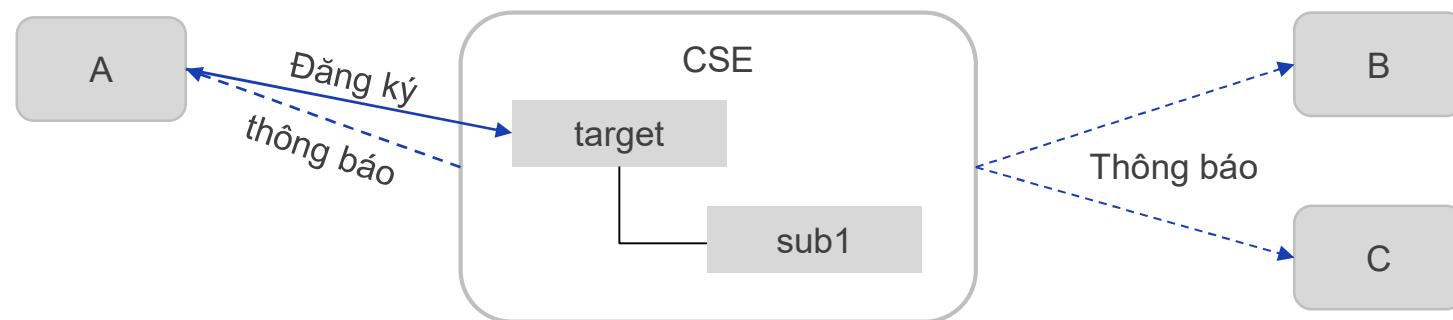
| sau đó CSE sẽ đặt thông tin vị trí làm tài nguyên <contentInstance> và các ứng dụng sẽ nhận được thông tin đó

| API liên quan

- Tài nguyên <locationPolicy>: CRUD

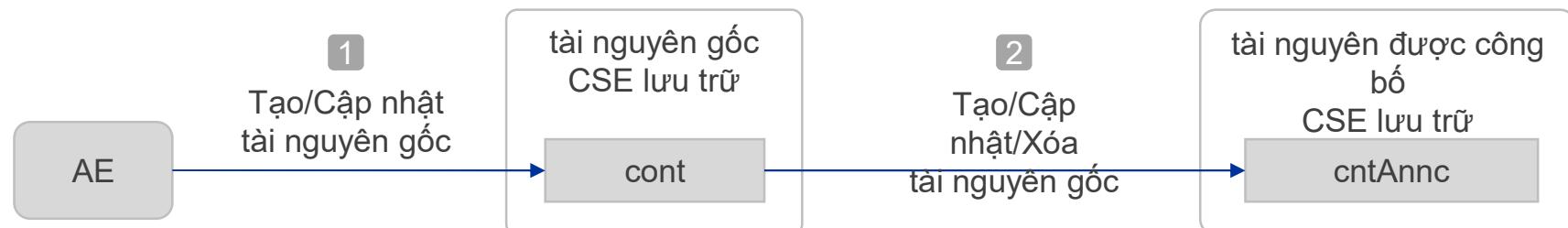
Kiểu tài nguyên Subscription

- Tài nguyên <subscription> biểu diễn các cấu hình để xác định sự kiện và gửi/truyền thông báo
- Không phải kiểu tài nguyên nào cũng có thể đăng ký
- Mục đích theo dõi sự kiện được đăng ký với tài nguyên và tài nguyên con; có thể lọc sự kiện
- Có thể có nhiều mục đích thông báo và bên đăng ký tài nguyên khác nhau có thể có mục đích thông báo khác nhau
- API liên quan
 - Tài nguyên <subscription>: CRUD



Kiểu tài nguyên xxAnnnc¹⁾

- █ Chức năng công bố trong oneM2M là tạo và sao phép một phần tài nguyên gốc sang CSE từ xa
 - Ứng dụng có thể chọn các thuộc tính cần công bố
 - Tài nguyên đã công bố sẽ có liên kết đến tài nguyên gốc
- █ Chức năng công bố giúp tìm kiếm tài nguyên dễ dàng hơn
- █ Ứng dụng không thể trực tiếp tạo tài nguyên đã công bố, ứng dụng sẽ phải yêu cầu CSE tạo/Cập nhật/Xóa tài nguyên đó
- █ API liên quan
 - Tài nguyên <announcableResourceType>: CU



1) Ví dụ: containerAnnnc

Xử lý yêu cầu

kiểu tài nguyên request

- Lưu trữ thông tin về trạng thái xử lý yêu cầu khi liên lạc không tuân tự
- Không có API riêng nhưng sử dụng tham số Kiểu phản hồi

kiểu tài nguyên delivery

- Gửi nhiều yêu cầu đến một CSE cùng lúc
- Không có API riêng nhưng sử dụng tham số Tổng hợp phân phối

Kiểu tài nguyên pollingChannel và pollingChannelURI

- Yêu cầu kiểm tra vòng thông báo khi thực thể không nhận được thông báo đầy đến
- API liên quan
 - Tài nguyên <pollingChannel>: CRUD
 - Tài nguyên <pollingChannelURI>: R

Bài 1.

Sử dụng nền tảng

- | 1.1. Kiến trúc nền tảng oneM2M
- | 1.2. Nền tảng mã nguồn mở IoT: OCEAN

OCEAN là gì

- | OCEAN (Open allianCE for iot stANDard) là liên minh được KETI và chính phủ Hàn Quốc thành lập năm 2015.
- | Mục tiêu của OCEAN là chia sẻ mã nguồn mở dựa trên tiêu chuẩn IoT và khuyến khích các thành viên trong liên minh hợp tác với nhau
- | OCEAN hỗ trợ hệ sinh thái thương mại hóa và quan trọng hóa sớm cho lĩnh vực IoT
- | Mã nguồn mở được phân phối trên trang chủ Github

Chính sách cấp phép

- | Chính sách cấp phép BSD 3 điều khoản
- | OCEAN điều chỉnh chính sách IPR gồm các tiêu chuẩn tham khảo từ nguồn mở

Thành viên hiện tại

- | Hơn 700 thành viên

Mã nguồn mở

| Càng nhiều nền tảng → càng phức tạp

OCEAN DEVELOPERS

MAKE THINGS CONNECTED AND INTELLIGENT

OCEAN provides open sources that realizes IoT-enabled smart world more easily

Collaborative open source projects for Internet of Things, data processing, interoperability and intelligence

RESOURCES FOR DEVELOPERS
CREATE, COLLABORATE AND INNOVATE

- OPEN SOURCES**
Open source implementations of IoT platform standard with open data, analysis and interworking features
- LABS**
Experiment and adaptation of OCEAN open source projects
- TUTORIAL**
Slides and webinars that help understanding OCEAN open source and lab projects

MODULE UPDATES

IOT-NODE DEVICE	TOOLS	IOT-SERVER	ACADEMY
NCUBE THYME NODEJS 2.0.0 UPDATED Application Dedicated Node Dev... August 25, 2017	ONEMZBROWSER 1.0.4.1 UPDATED Resources Monitoring Tool for M... August 25, 2017	MOBIUS 2.0.0 UPDATED Mobius is an open source softw... August 25, 2017	IOT OCEAN SJU UPDATED IOT OCEAN OPEN Lab in Sejong... July 12, 2017

<http://developers.iotocean.org/>

OCEAN DEVELOPERS

OPEN SOURCE

- IoT**
Standard compliant IoT platform implementations
- APPLICATION**
Service applications that made of OCEAN open sources
- DATA & ANALYSIS**
Open data platform, semantic platform, big data analysis component
- INTERWORKING**
Interworking with other IoT platforms, connectivity technologies and legacy devices
- INTELLIGENCE**
Enabling IoT platform, enables to existing IoT platforms
- TOOLS**
Developer supporting tools like oneM2M resource browser, performance tester

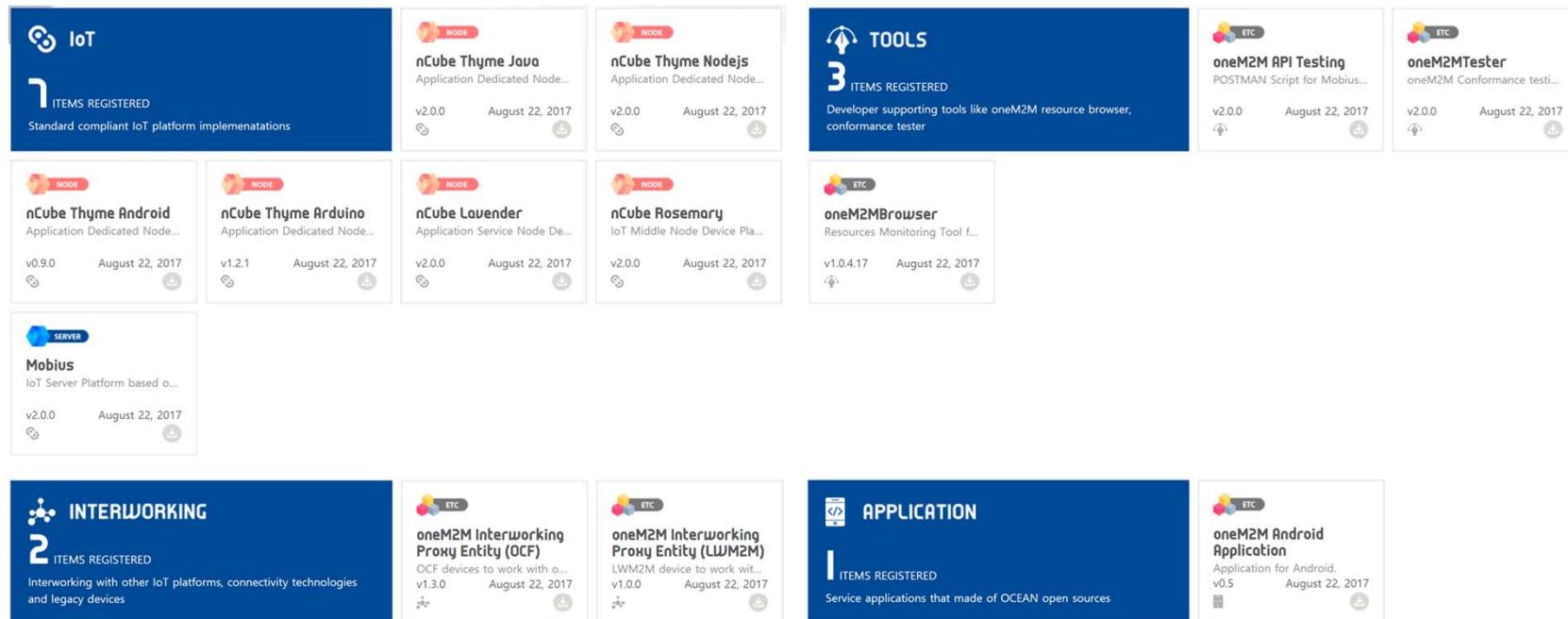
Mobius
IoT Server Platform based on Node.js.

Introduction
Mobius is the open source IoT server platform based on the oneM2M (<http://www.onem2m.org>) standard. As oneM2M specifies, Mobius provides common services functions (e.g. registration, data management, subscription/notification, security) as middleware to IoT applications of different service domains. Not just oneM2M devices, but also non-oneM2M devices (i.e. by oneM2M Interworking Specifications and OCEAN TAK) can connect to Mobius.

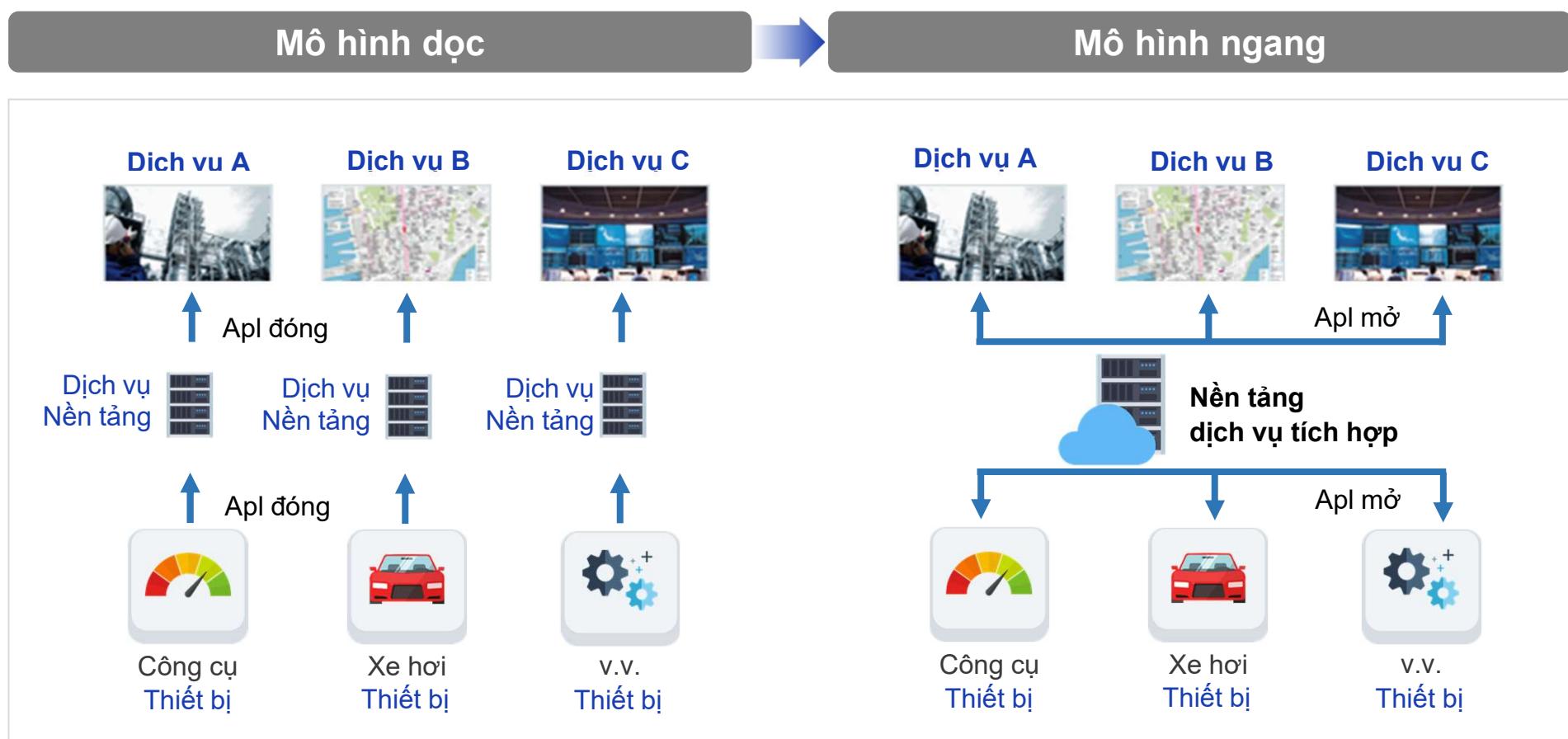
System Structure
In oneM2M architecture, Mobius implements the IN-CSE which is the cloud server in the infrastructure domain. IoT applications communicate with field domain IoT gateways/devices via Mobius.

[Http://developers.iotocean.org/opensource](http://developers.iotocean.org/opensource)

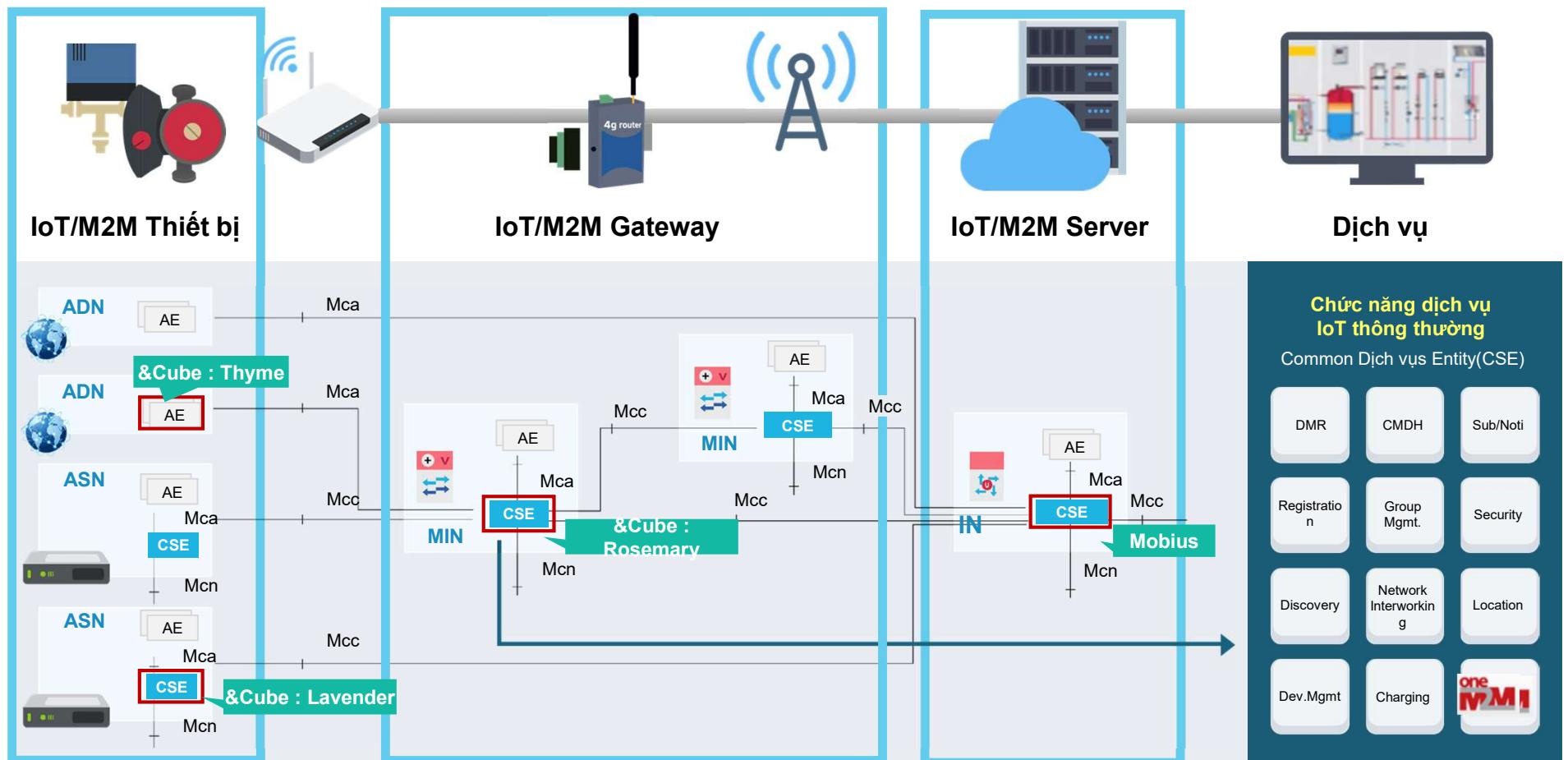
- █ Nền tảng IoT: 7 (Máy chủ 1, Thiết bị cổng 1, Thiết bị 5)
- █ Liên kết mạng: 2 (OCF, LWM2M)
- █ Công cụ hỗ trợ: 3 (Kiểm tra mức độ phù hợp, Kiểm tra chức năng, Tài liệu hướng dẫn về API)
- █ Ứng dụng: 1 (Mẫu Android)



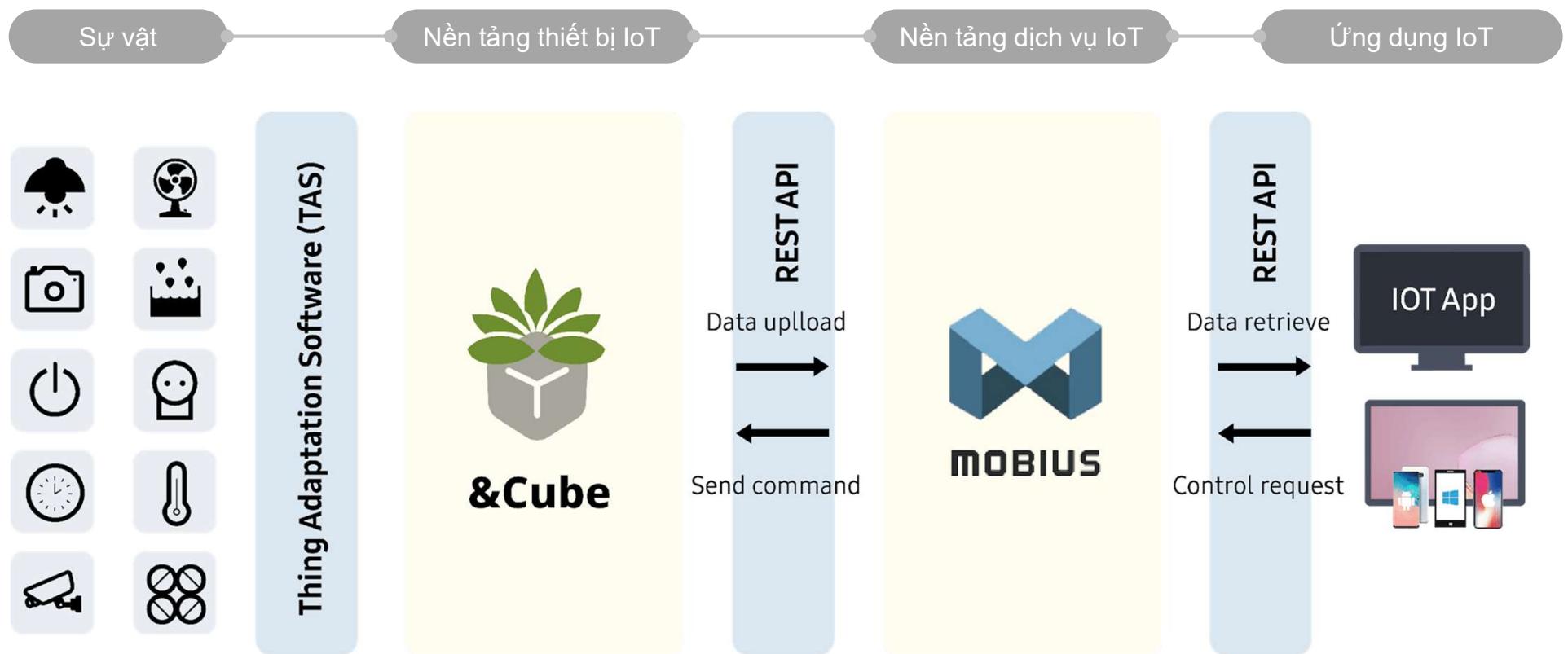
Nền tảng IoT ngang



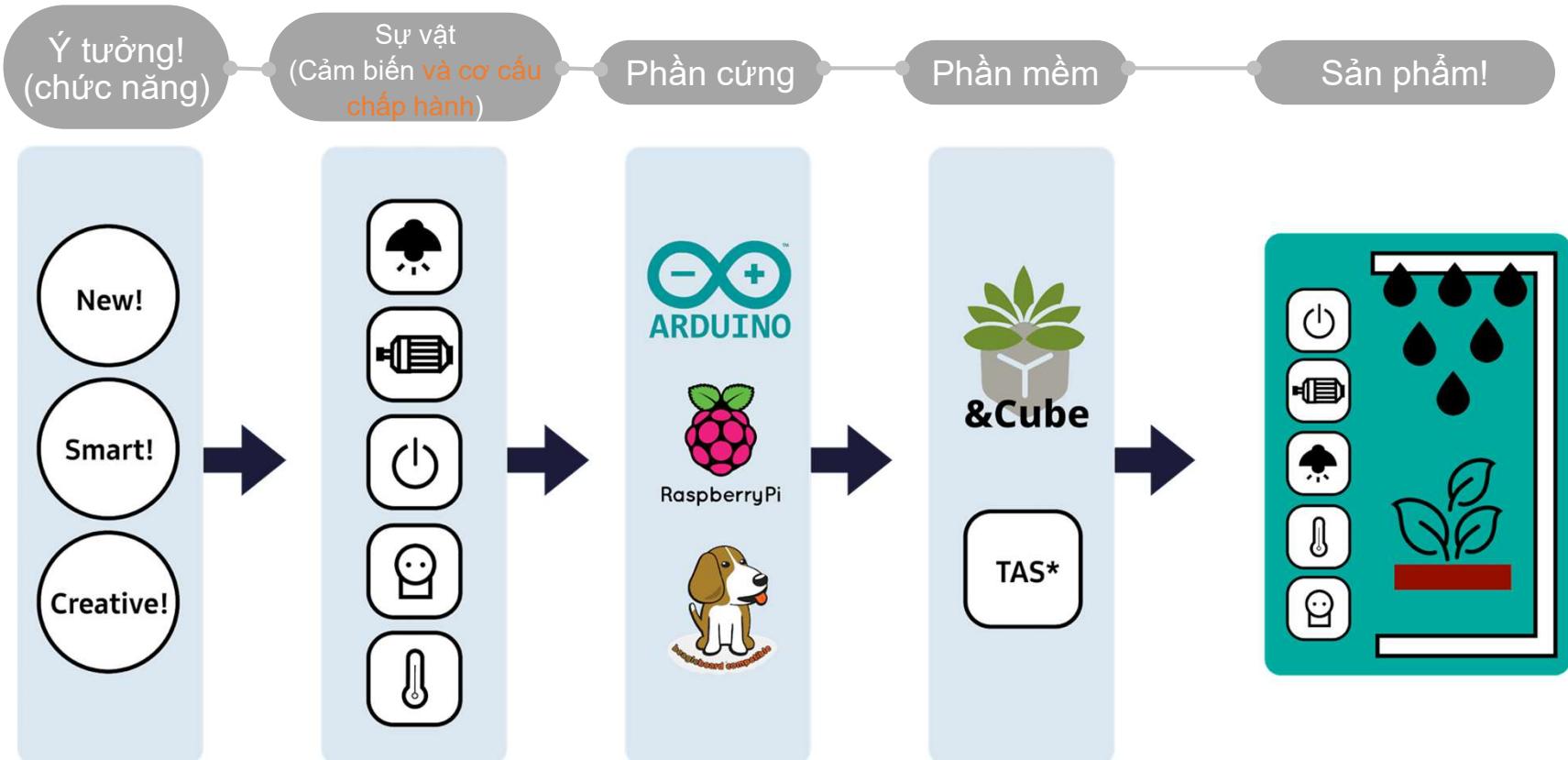
Cấu trúc kết nối



Cách thức hoạt động của thiết bị IoT



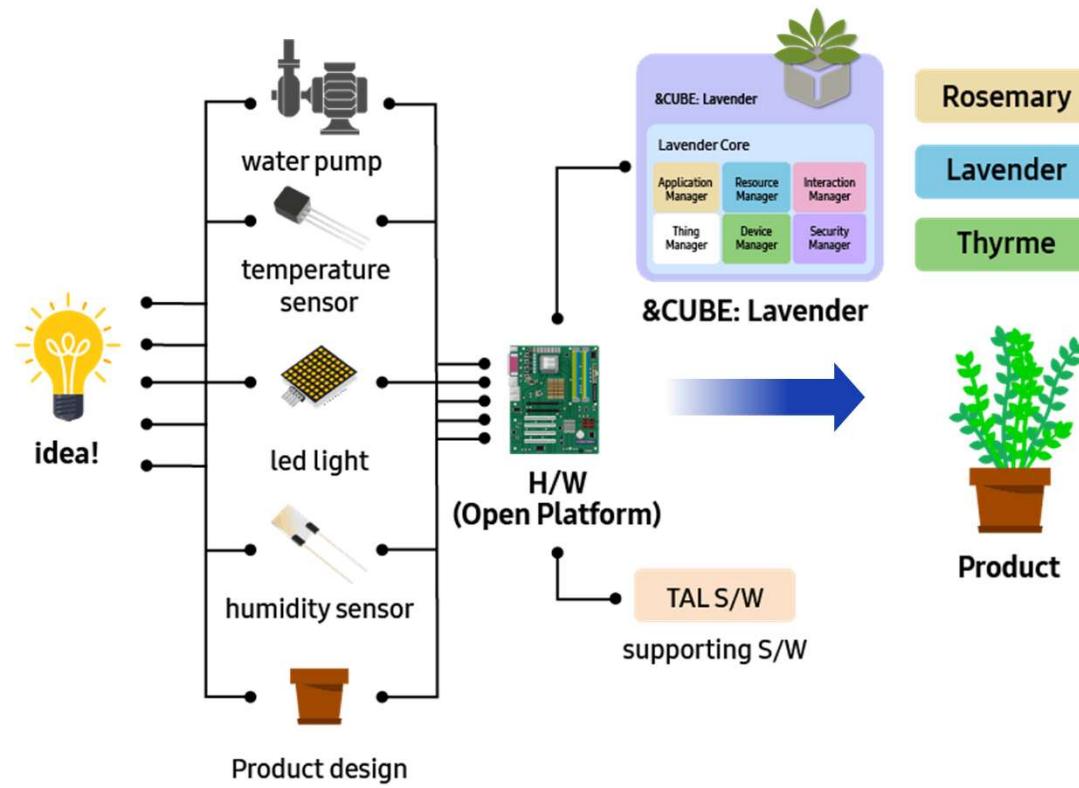
Quy trình phát triển



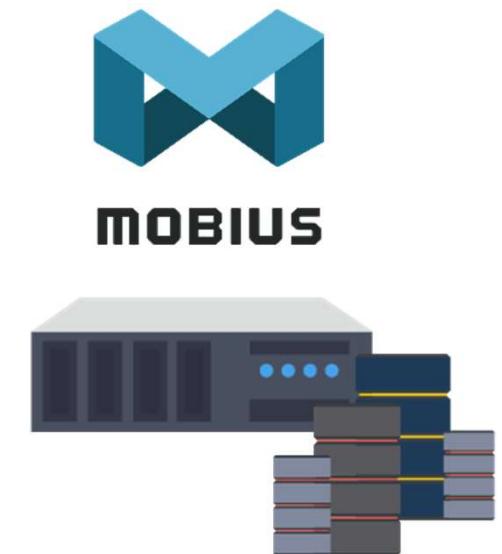
*TAS: Phần mềm thích ứng sự vật

Cách thức phát triển

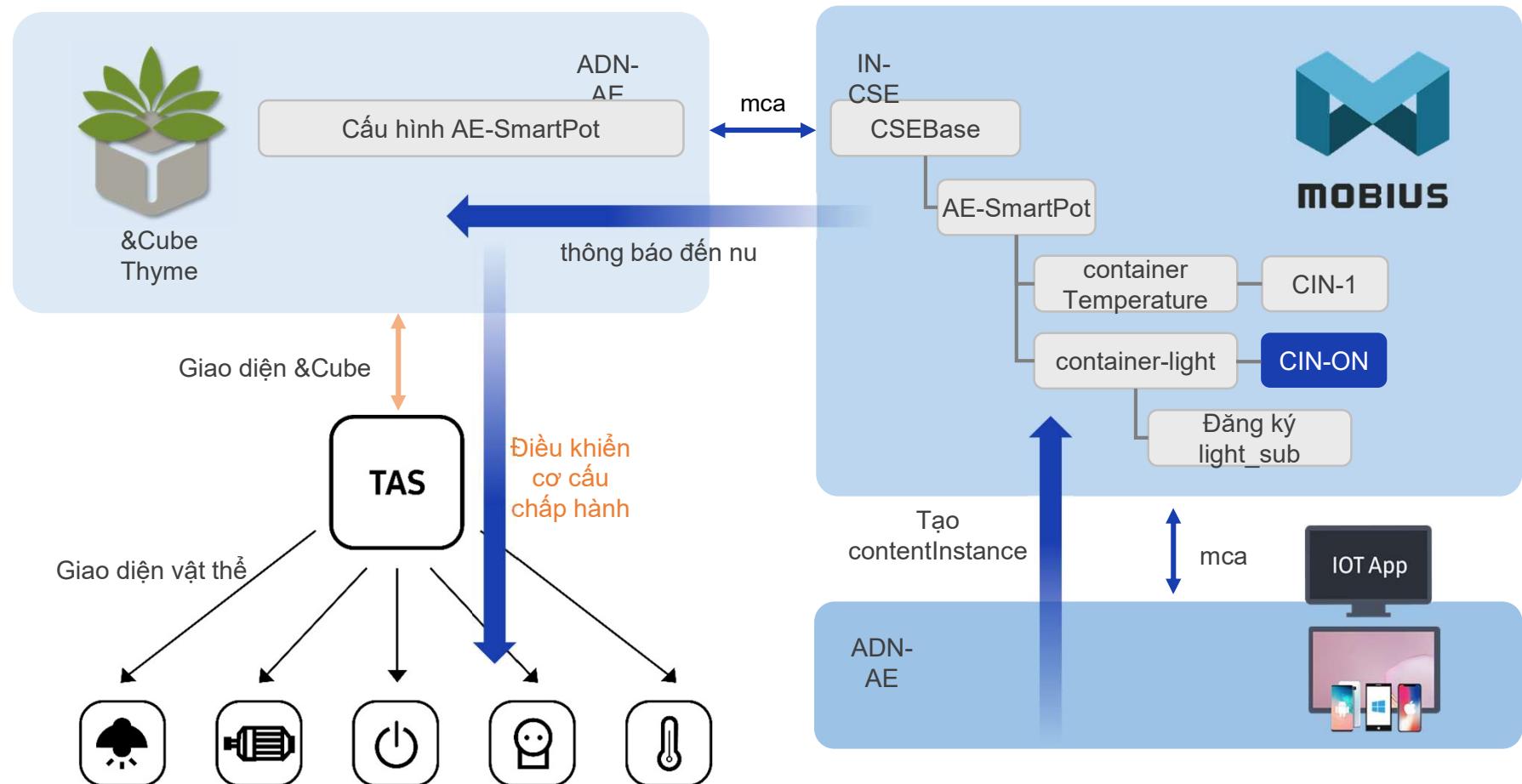
Sản phẩm=Ý tưởng + H/W &CUBE + TAS



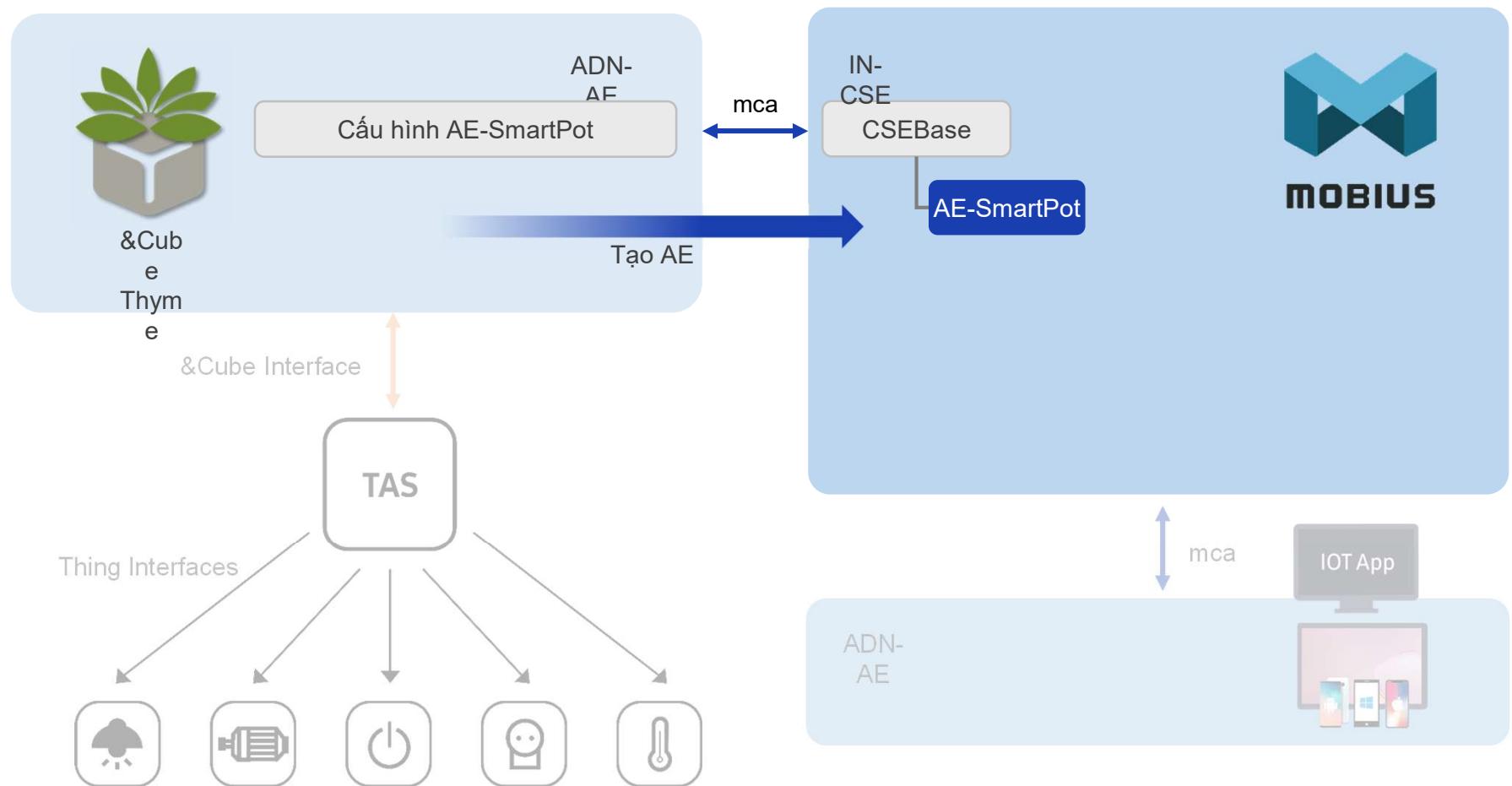
IoT Server(H/W) + Mobius



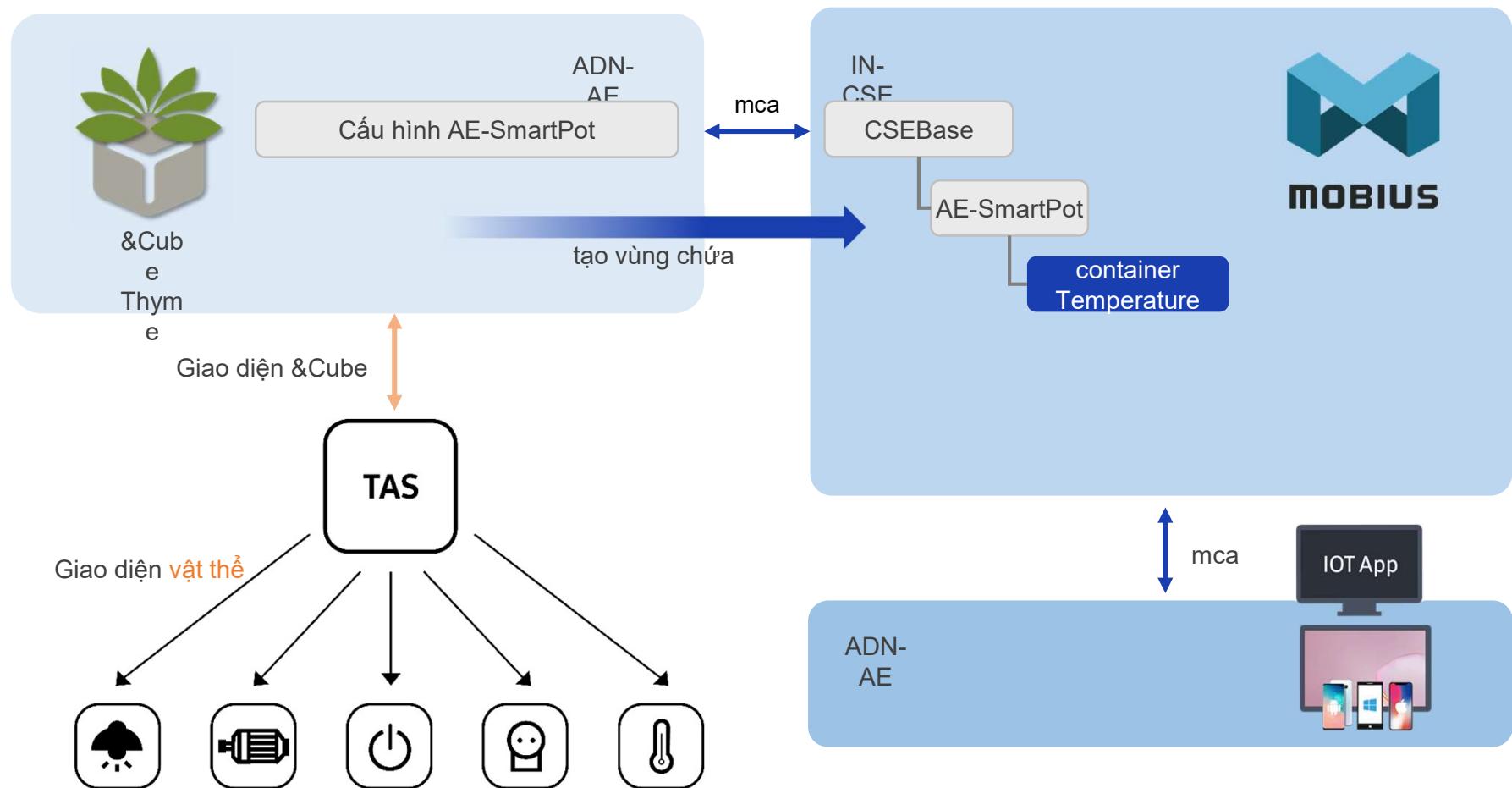
Điều khiển vật thể (cảm biến)



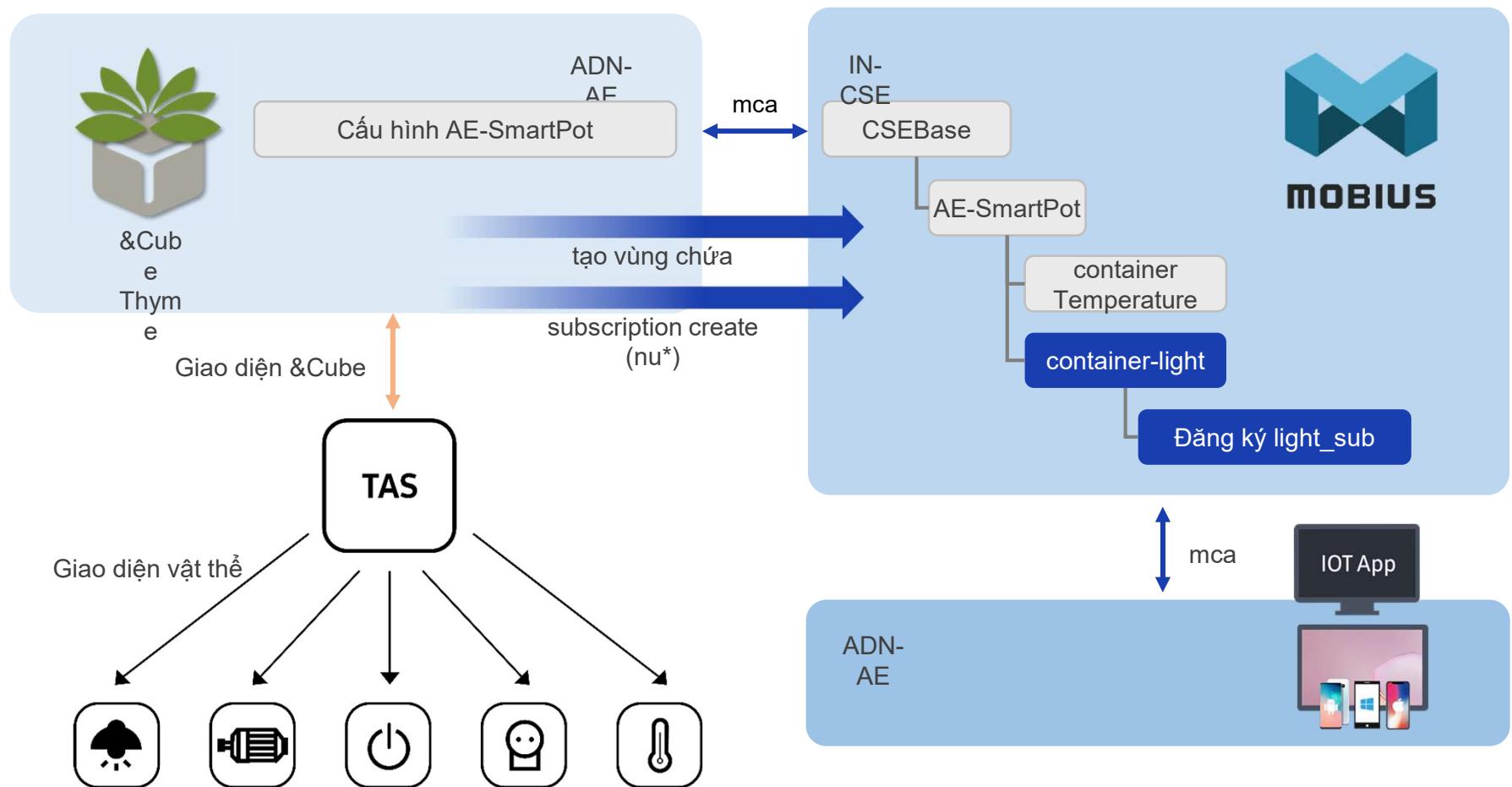
Đăng ký thiết bị



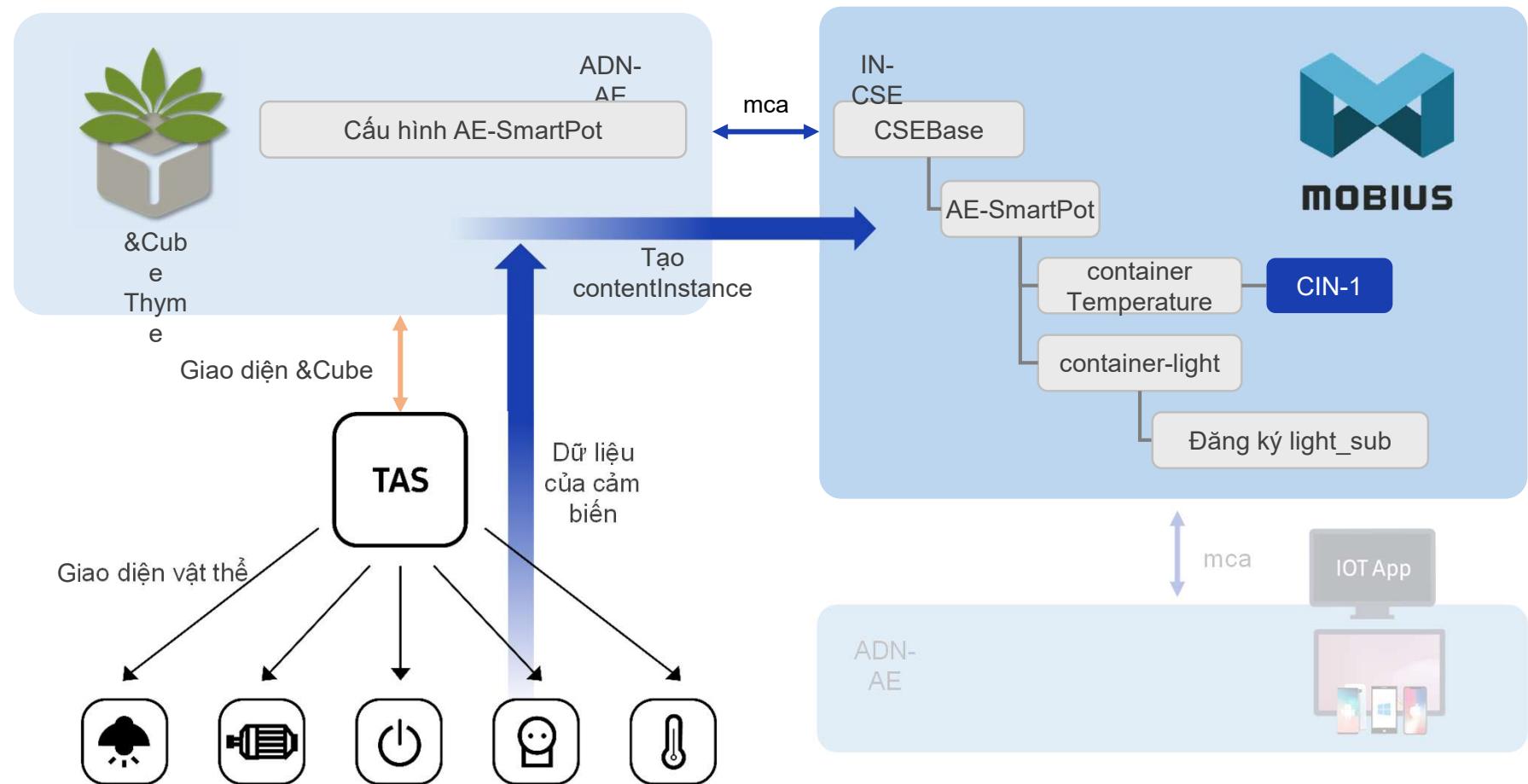
Đăng ký vật thể (cảm biến)



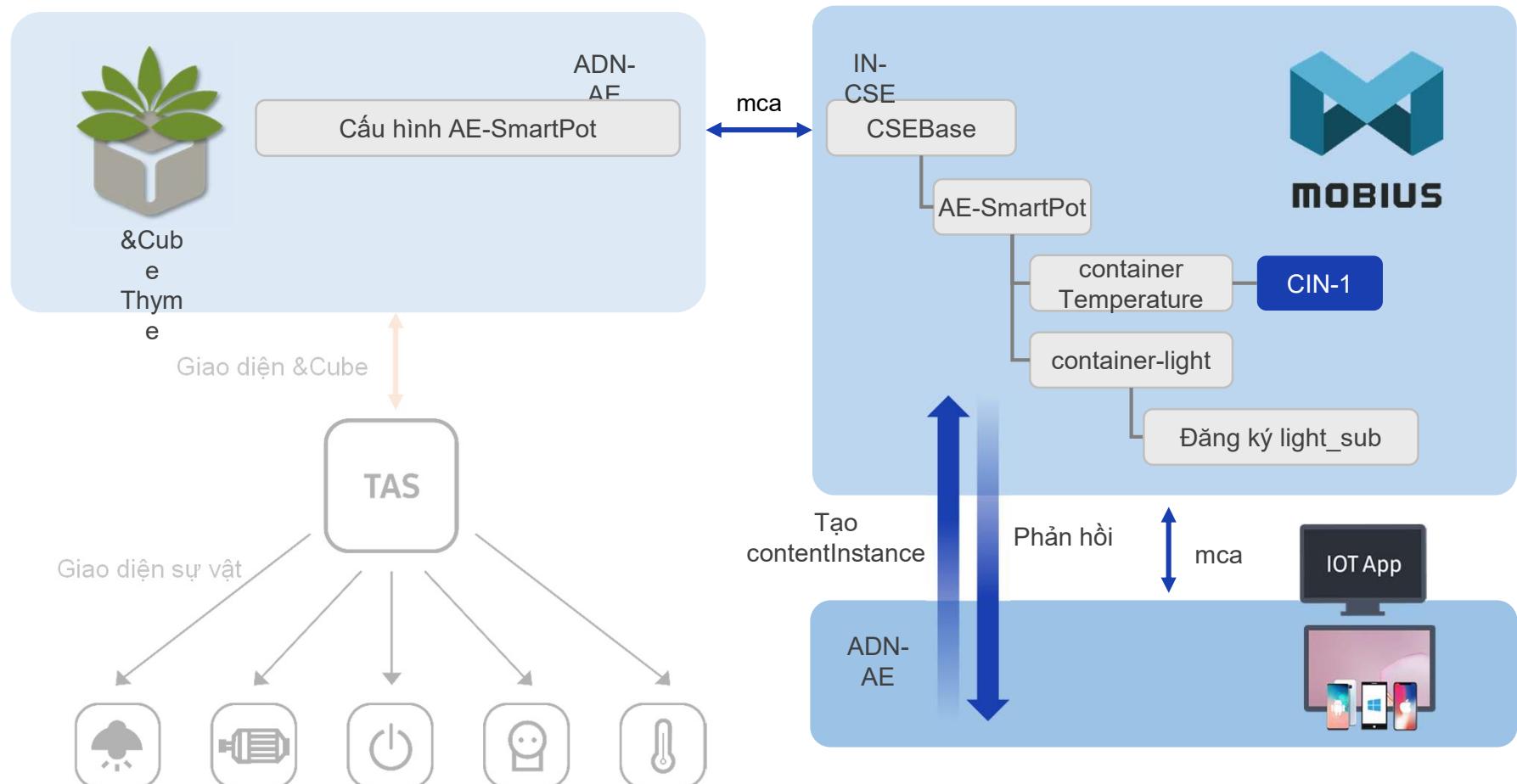
Đăng ký vật thể (cơ cấu chấp hành)



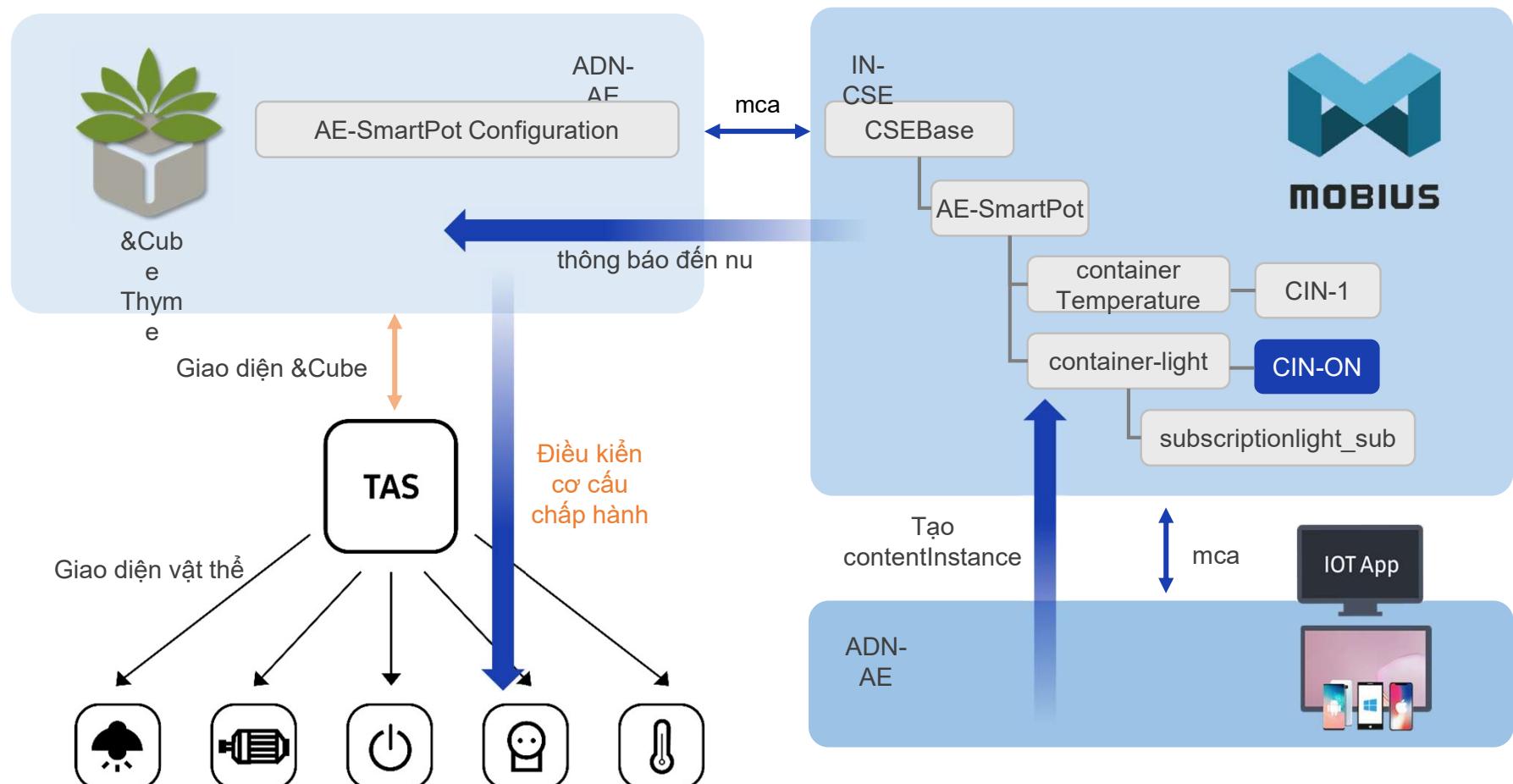
Tải lên dữ liệu sự vật (cảm biến)



Truy xuất dữ liệu sự vật (cảm biến)



Điều khiển vật thể (cảm biến)



Raspberry Pi 3 Model B (hoặc 4 Model B)

- Raspberry Pi là tập hợp các máy tính bo mạch đơn với kích thước bằng thẻ tín dụng, được Raspberry Pi Foundation phát triển tại Vương quốc Anh
- Sử dụng hệ điều hành (OS) Raspbian (Debian Linux)
- Hỗ trợ các hệ điều hành khác
- Tham khảo tại: <http://www.raspberrypi.org/>



Raspberry Pi 3 Model B

Cảm biến CO2 CM1106

| CM1106 là mô-đun cảm biến CO₂ hồng ngoại thu nhỏ để phát hiện nồng độ CO₂

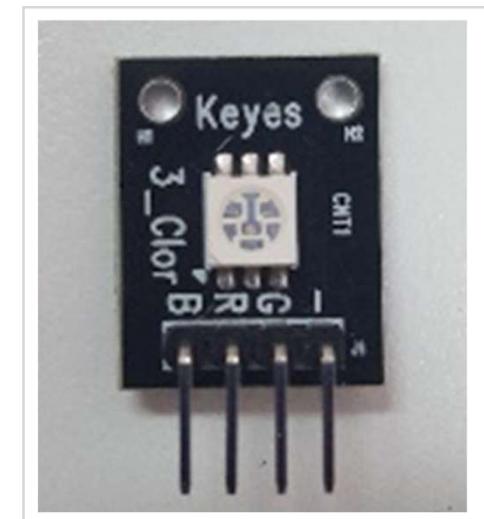
1. Phạm vi đo: 0-2000ppm, 0-5000ppm
2. Nguồn điện: DC5V±5%, Pmax0,6W
3. Kích thước: 33*19,7*8,9 mm
4. Độ phân giải: 1ppm
5. Độ chính xác: ± (50Ppm+5% khả năng đọc)
6. Cách lấy mẫu: khuếch tán
7. Thời gian phản hồi: 60s
8. Điều kiện hoạt động: 0-50°C, 0-95% RH
9. Ngõ ra tín hiệu: UART, PWM
10. Tuổi thọ: 10 năm



Mô-đun LED RGB SMD (SZH-EK045)

Mô-đun LED RGB bao gồm một đèn LED SMD hiển thị đầy đủ màu sắc được tạo bởi điện áp đầu vào PWM ba chân R, G, B có thể điều chỉnh cường độ màu của ba màu cơ bản (đỏ/xanh dương/xanh lá) để đạt được hiệu ứng trộn màu đầy đủ

1. sử dụng đèn LED 5050 hiển thị đầy đủ màu sắc
2. Điện trở giới hạn ba màu RGB để tránh cháy nổ
3. có thể tạo bộ điều chỉnh PWM bằng cách trộn ba màu cơ bản thành nhiều màu khác nhau
4. có thể là giao diện có nhiều loại vi điều khiển
5. Điện áp vận hành: 5 V
6. Chế độ ổ đĩa LED: ổ đĩa cực âm phổ biến



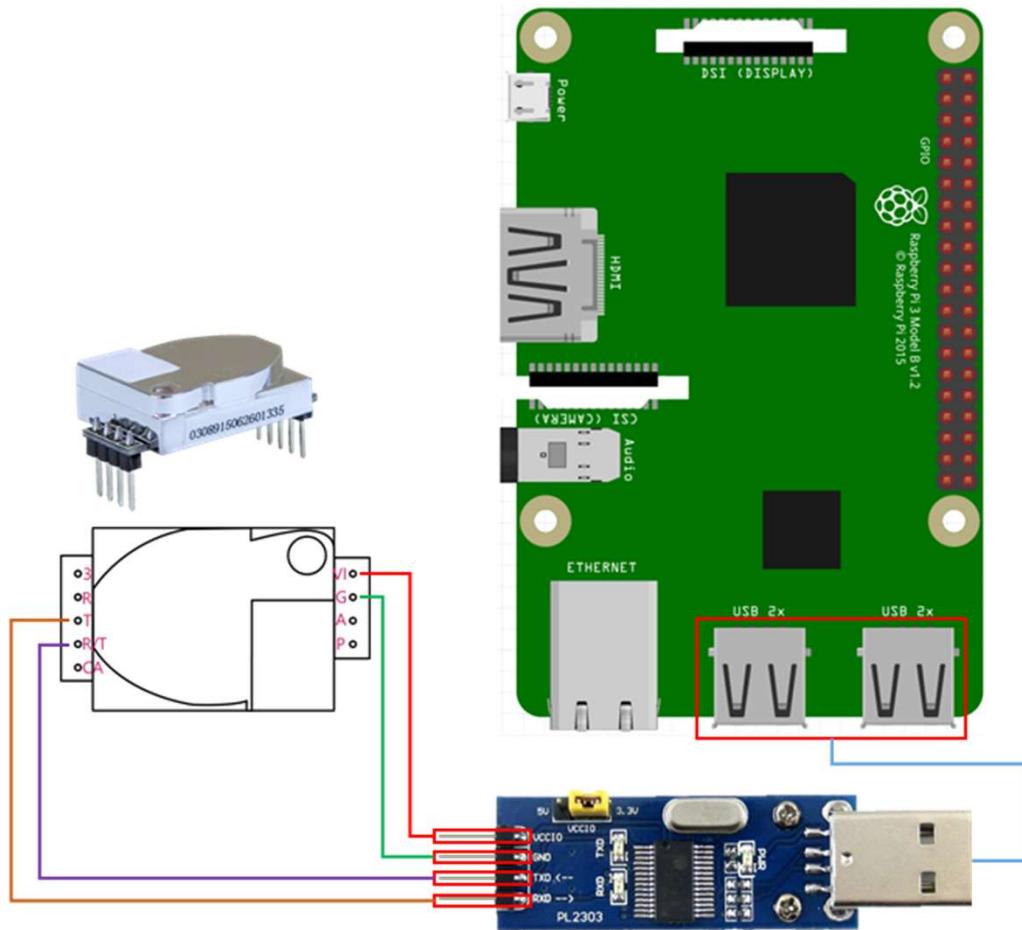
Bo mạch UART USB PL2303 (loại A)

| Giải pháp USB TO UART với đầu nối USB Type A

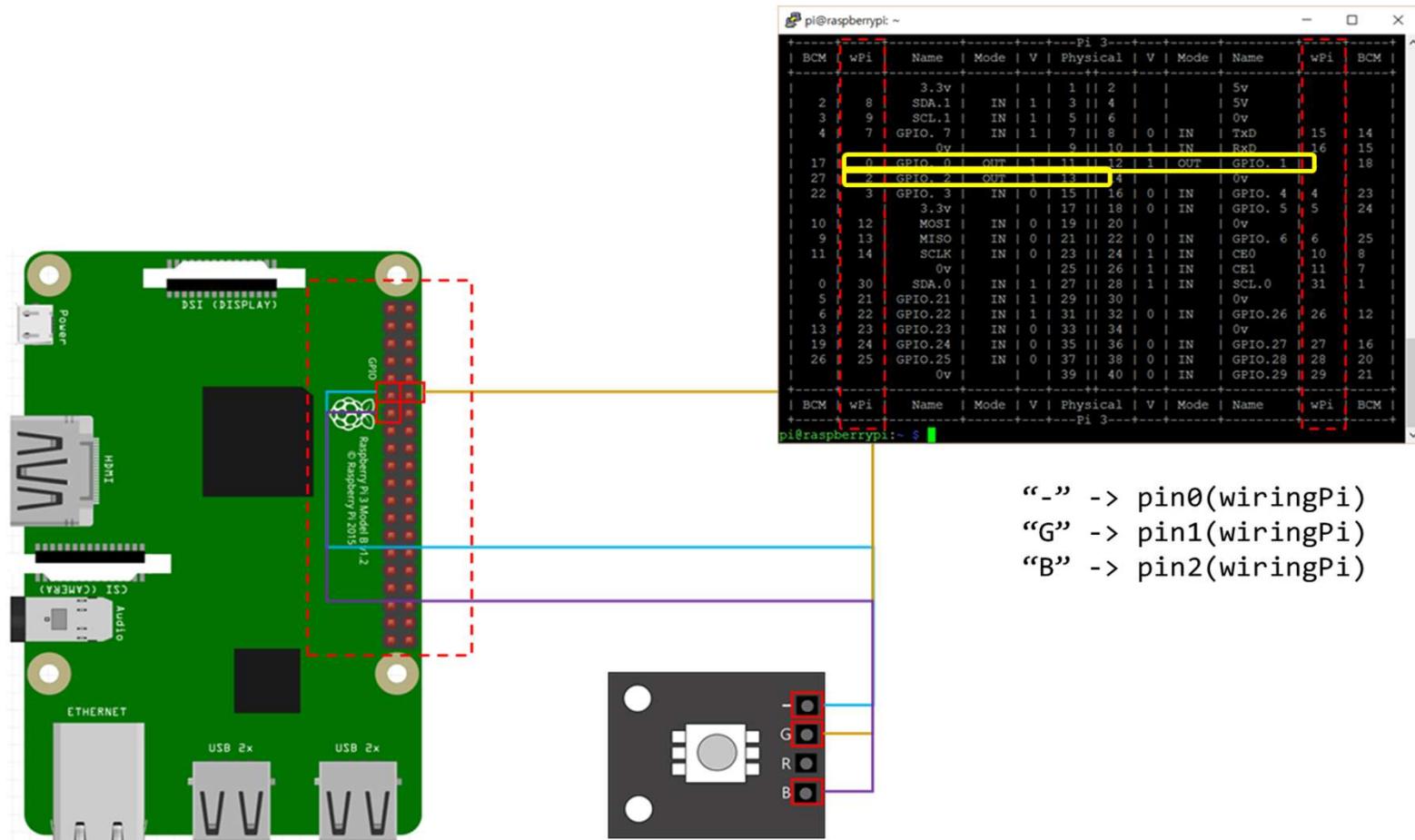
1. PL2303TA tích hợp (thiết bị mới nhất, tương thích với PL2303HXA)
2. Hỗ trợ windows XP/7/8/8.1/10/...
3. chế độ nguồn: Đầu ra 5V, đầu ra 3,3V hoặc cấp nguồn bằng bo mạch đích (3,3V-5V)
4. 3 đèn LED: TXD LED, RXD LED, POWER LED



Thiết lập cảm biến CO₂



Thiết lập đèn LED RGB



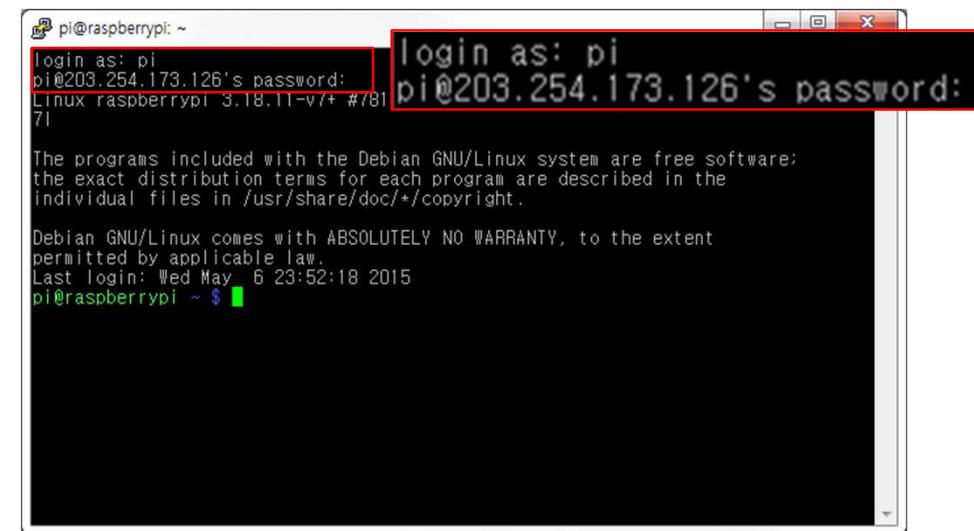
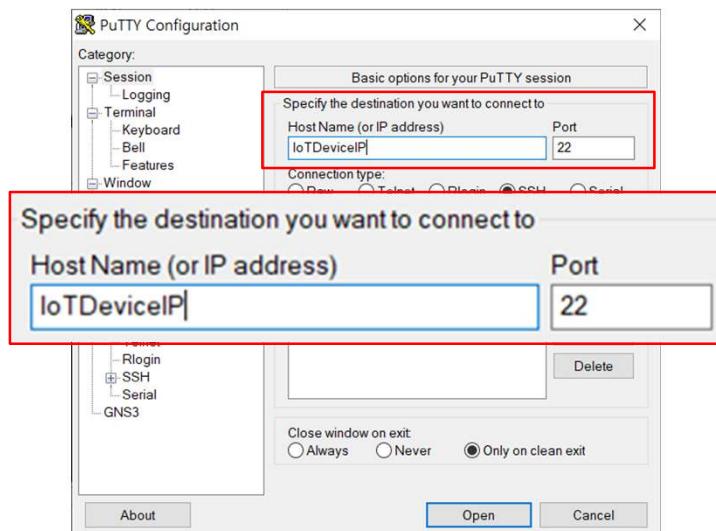
Thiết lập máy khách SSH trên Windows

Tải xuống và cài đặt máy khách SSH (PuTTY)

▶ <http://www.chiark.greenend.org.uk/~sgtatham/putty/>

Thiết lập cấu hình và kết nối PuTTY

1. Run (Chạy) → Host Name (Tên máy chủ) (địa chỉ IP của Raspberry-pi) → Open (Mở)
2. Xác nhận Khóa máy chủ (Có)
3. Đã kết nối → ID: pi, PW: raspberry (Raspberry-pi mặc định)



Thiết lập máy chủ tệp Samba trong Raspberry Pi

| Cập nhật kho lưu trữ

```
pi@raspberrypi ~ $ sudo apt-get update
```

.....
Reading package lists... Done

| Cài đặt Samba

```
pi@raspberrypi ~ $ sudo apt-get install samba samba-common-bin
```

.....
Bạn có muốn tiếp tục [Y/n]? Y

| Đặt mật khẩu Samba

```
pi@raspberrypi ~ $ sudo smbpasswd -a pi
```

Mật khẩu SMB mới: (Input user password)

Xác minh mật khẩu SMB mới: (Nhập mật khẩu người dùng)

Đã thêm pi người dùng.

| Cấu hình người dùng
Samba

```
pi@raspberrypi ~ $ sudo vi /etc/samba/smb.conf
```

..... (Kết thúc tệp)

[pi]

comment = raspberry pi folder

path = /home/pi

valid user = pi

writable = yes

browsable = yes

Thiết lập máy chủ tệp Samba trong Raspberry - Pi (tiếp)

Khởi động lại Samba

```
pi@raspberrypi ~ $ sudo Dịch vụ samba restart  
[ ok ] Dừng chạy chương trình nền của Samba: nmbd smbd.  
[ ok ] Khởi chạy chương trình nền của Samba : nmbd smbd.
```

Kết nối với thư mục Samba trên Windows

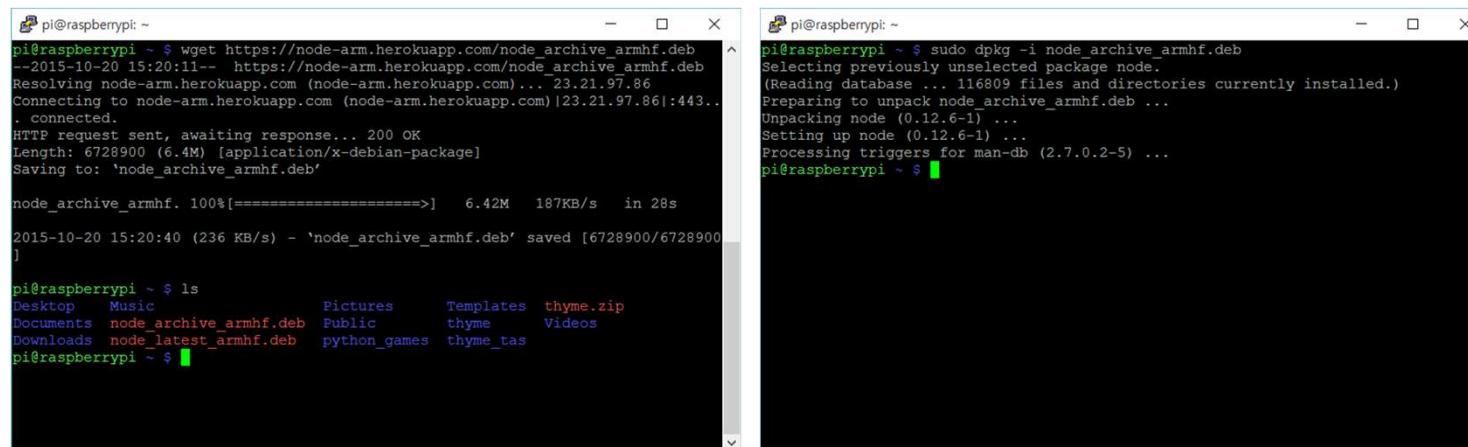
- Chạy File Explorer
- Nhập (địa chỉ IP raspberry-pi)
- **Ex** \\203.254.173.126
- nhấp đúp chuột vào thư mục pi
- Nhập (ID và mật khẩu)

Thiết lập Node.js trong Raspberry-Pi

Cấu hình phần cứng

- Node.js installation

```
pi@raspberrypi ~ $ mkdir node
pi@raspberrypi ~ $ cd node
pi@raspberrypi ~/node $ sudo apt-get update
pi@raspberrypi ~/node $ sudo apt-get upgrade
pi@raspberrypi ~/node $ sudo apt-get remove nodejs
pi@raspberrypi ~/node $ sudo wget https://node-arm.herokuapp.com/node_archive_armhf.deb
pi@raspberrypi ~/node $ sudo dpkg -i node_archive_armhf.deb (Lệnh cài đặt gói)
pi@raspberrypi ~/node $ node -v (Lệnh xem phiên bản Node.js)
pi@raspberrypi ~/node $ npm -v (Lệnh xem phiên bản npm của công cụ cài đặt thư viện bên ngoài)
```



```
pi@raspberrypi: ~
pi@raspberrypi ~ $ wget https://node-arm.herokuapp.com/node_archive_armhf.deb
--2015-10-20 15:20:11-- https://node-arm.herokuapp.com/node_archive_armhf.deb
Resolving node-arm.herokuapp.com (node-arm.herokuapp.com) ... 23.21.97.86
Connecting to node-arm.herokuapp.com (node-arm.herokuapp.com)|23.21.97.86|:443...
HTTP request sent, awaiting response... 200 OK
Length: 6728900 (6.4M) [application/x-debian-package]
Saving to: 'node_archive_armhf.deb'

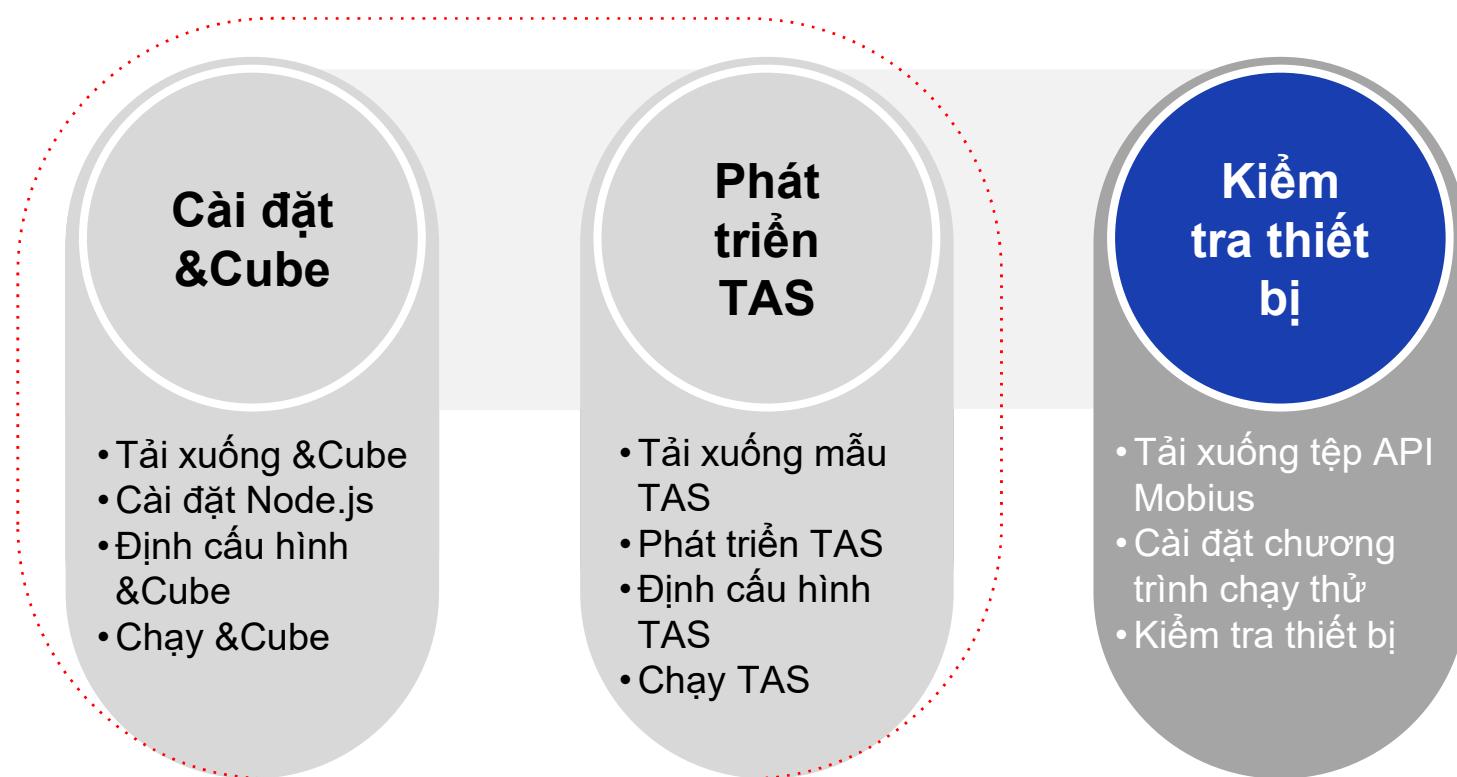
node_archive_armhf. 100%[=====] 6.42M 187KB/s in 28s

2015-10-20 15:20:40 (236 KB/s) - 'node_archive_armhf.deb' saved [6728900/6728900]

pi@raspberrypi ~ $ ls
Desktop   Music      Pictures    Templates thyme.zip
Documents node_archive_armhf.deb Public     thyme   Videos
Downloads node_latest_armhf.deb python_games thyme_tas
pi@raspberrypi ~ $
```

```
pi@raspberrypi: ~
pi@raspberrypi ~ $ sudo dpkg -i node_archive_armhf.deb
Selecting previously unselected package node.
(Reading database ... 116809 files and directories currently installed.)
Preparing to unpack node_archive_armhf.deb ...
Unpacking node (0.12.6-1) ...
Setting up node (0.12.6-1) ...
Processing triggers for man-db (2.7.0.2-5) ...
pi@raspberrypi ~ $
```

Cài đặt &Cube và TAS (Pi)



&Cube: Tải xuống và sao chép Thyme và TAS sang Raspberry Pi

- Tải xuống &Cube: Sử dụng lệnh git, sao chép Thyme và TAS từ địa chỉ <https://github.com/IoTKETI/nCube-Thyme-Nodejs.git>

```
[pi@raspberrypi:~ $ git clone https://github.com/IoTKETI/nCube-Thyme-Nodejs.git
Cloning into 'nCube-Thyme-Nodejs'...
remote: Counting objects: 123, done.
remote: Total 123 (delta 0), reused 0 (delta 0), pack-reused 123
Receiving objects: 100% (123/123), 8.60 MiB | 1.39 MiB/s, done.
Resolving deltas: 100% (64/64), done.
Checking connectivity... done.
pi@raspberrypi:~ $ ]
```

&Cube: Thiết lập Thyme bằng Raspberry Pi

- █ Đăng nhập Raspberry Pi bằng chương trình putty SSH
- █ Vào thư mục Thyme và cài đặt Thyme
 - pi@raspberrypi : ~ \$ cd nCube-Thyme-Nodejs
 - pi@raspberrypi : ~/nCube-Thyme-Nodejs \$ sudo npm install

```
mqtt@1.14.0 node_modules/mqtt
└── inherits@2.0.1
└── reinterval@1.1.0
└── xtend@4.0.1
└── help-me@0.1.0
└── minimist@1.2.0
└── readable-stream@1.0.34 (string_decoder@0.10.31, isarray@0.0.1, core-util-is@1.0.2)
    ├── commist@1.0.0 (leven@1.0.2)
    ├── mqtt-connection@2.1.1 (through2@0.6.5, reduplexer@1.1.0)
    ├── mqtt-packet@3.4.7 (bl@0.9.5)
    ├── end-of-stream@1.1.0 (once@1.3.3)
    ├── pump@1.0.1 (once@1.3.3)
    ├── concat-stream@1.5.1 (typedarray@0.0.6, readable-stream@2.0.6)
    └── split2@2.1.0 (through2@2.0.1)
    └── websocket-stream@3.2.1 (ws@1.1.1, through2@2.0.1, duplexify@3.4.5)

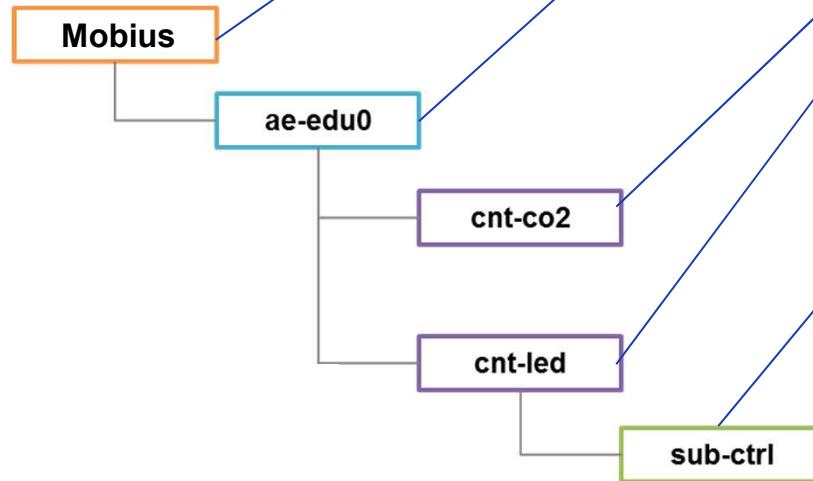
xmlbuilder@2.6.5 node_modules/xmlbuilder
└── lodash@3.10.1

xml2js@0.4.17 node_modules/xml2js
└── sax@1.2.1
    └── xmlbuilder@4.2.1 (lodash@4.15.0)
pi@raspberrypi:~/Thyme $
```

Định cấu hình & Cube: Thyme

Mở tệp cấu hình và chỉnh sửa

```
pi@raspberrypi : ~/nCube-Thyme-Nodejs $ sudo nano conf.js
..... (Chỉnh sửa tệp cấu hình)
<Ctrl>+<X> → Y → <Enter>
```



```

conf.useprotocol = 'http';

// build cse
cse.host      = '203.253.128.161';
cse.port      = '7579';
cse.name      = 'Mobius';
cse.id        = '/Mobius';
cse.mqttport  = '1883';

// build ae
ae.parent    = '/' + cse.name;
ae.name      = 'ae-edu0';
ae.id        = ae.name;
ae.appid    = '0.2.481.1.1';
ae.port      = '9727';
ae.bodytype  = 'json';
ae.tasport   = '3105';

// build cnt
var count = 0;
cnt_arr[count] = {};
cnt_arr[count].parent = '/' + cse.name + '/' + ae.name;
cnt_arr[count++].name = 'cnt-co2';
cnt_arr[count] = {};
cnt_arr[count].parent = '/' + cse.name + '/' + ae.name;
cnt_arr[count++].name = 'cnt-led';
//cnt_arr[count] = {};
//cnt_arr[count].parent = '/' + cse.name + '/' + ae.name;
//cnt_arr[count++].name = 'cnt-cam';

// build sub
count = 0;
sub_arr[count] = {};
sub_arr[count].parent = '/' + cse.name + '/' + ae.name + '/' + cnt_arr[1].name;
sub_arr[count].name = 'sub-ctrl';
sub_arr[count++].nu = 'mqtt://'+cse.host+'/' + ae.id;
//sub_arr[count++].nu = 'http://203.254.173.104:' + ae.port + '/noti';
//sub_arr[count++].nu = 'coap://203.254.173.104:' + ae.port + '/noti';

// build acp: not complete
acp.parent = '/' + cse.name + '/' + ae.name;
acp.name = 'acp-' + ae.name;
acp.id = ae.id;

conf.usesscure = 'disable';
  
```

Định cấu hình & Cube: Thyme

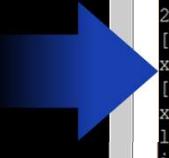
Các tham số

Tham số		Mô tả	Nội dung
cse	useprotocol	Định cấu hình giao thức liên lạc giữa thyme và mobius	Chuỗi (opt: "http", "mqtt", "coap")
	cse	Khu vực cấu hình của Thực thể dịch vụ chung	Đổi tượng JSON
	máy chủ	Định cấu hình IP máy chủ CSEBase	Chuỗi (ví dụ: "203.253.128.161")
	cổng kết nối	Định cấu hình cổng kết nối máy chủ CSEBase	Chuỗi (ví dụ: "7579")
	tên	Định cấu hình tên tài nguyên CSEBase	Chuỗi (ví dụ: "Mobius")
	id	Định cấu hình ID tài nguyên CSEBase	Chuỗi (ví dụ: "/Mobius")
ae	cổng mqtt	Định cấu hình cổng kết nối máy chủ cài MQTT	Chuỗi (ví dụ: "1883")
	đường dẫn cha mẹ	Định cấu hình đường dẫn cha mẹ tạo AE	Chuỗi (ví dụ: "ae-edu0")
	tên	Định cấu hình tên AE	Chuỗi (ví dụ: "example")
	id	Định cấu hình AE-ID	Chuỗi (ví dụ: "S12345") lưu ý: phải bắt đầu bằng "S" hoặc "C"
	id ứng dụng	Định cấu hình ID ứng dụng	Chuỗi (ví dụ: "0.13123.0012")
	cổng kết nối	Định cổng kết nối thông báo qua http của Mobius	Chuỗi (ví dụ: "7000")
cnt[]	bodytype	Định cấu hình kiểu thông báo yêu cầu/phản hồi	Chuỗi (opt: "json", "xml")
	cổng kết nối tas	Định cấu hình cổng TCP để liên lạc với TAS	Chuỗi (ví dụ: "3105") lưu ý: phải giống thông tin cấu hình của TAS
	đường dẫn cha mẹ	Định cấu hình đường dẫn cha mẹ tạo vùng chứa	Chuỗi (ví dụ: "/example") lưu ý: phải bắt đầu bằng "/"
	tên	Định cấu hình tên tạo vùng chứa	Chuỗi (ví dụ: "temp")
sub[]	đường dẫn cha mẹ	Định cấu hình đường dẫn cha mẹ tạo đăng ký	Chuỗi (ví dụ: "/example/ctrl") lưu ý: phải bắt đầu bằng "/"
	tên	Định cấu hình tên tạo đăng ký	Chuỗi (ví dụ: "sub")
	nu	Định cấu hình URL thông báo đăng ký	Chuỗi (ví dụ: "mqtt://203.253.128.161/S123145") lưu ý: Trường hợp của MQTT phải phù hợp với định dạng "mqtt://[ip]/[AE-ID]" Trường hợp của HTTP phải phù hợp với định dạng "http://[ip]/[path]"

Chạy &Cube: Thyme

Nhập &Cube: Lệnh chạy Thyme

```
pi@raspberrypi : ~/nCube-Thyme-Nodejs $ node thyme.js
```



```
pi@raspberrypi:~/Thyme
pi@raspberrypi:~/Thyme $ node thyme.js
```

```
CT0000000000000000000000000000000039</ri><pi>AE0000000000000000000000000000000096</pi><ct>20160822T144317</ct><lt>20160822T182327</lt><lbl>cnt-co2</lbl><et>99991231T000000</et><st>267</st><cr>SHJ-tF-Og000000000004931720896</cr><cni>267</cni><cbs>1068</cbs><la>/CB00061/18644122219/ae-edu0/cnt-co2/la</la><ol>/CB00061/18644122219/ae-edu0/cnt-co2/ol</ol></m2m:cnt>
[crtct response] : 409
x-m2m-rsc : 4105 <----<m2m:cnt rn="cnt-led" xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><ty>3</ty><ri>CT0000000000000000000000000000000038</ri><pi>AE0000000000000000000000000000000096</pi><ct>20160822T144317</ct><lt>20160822T182327</lt><lbl>cnt-led</lbl><et>99991231T000000</et><st>6</st><cr>SHJ-tF-Og000000000004931720896</cr><cni>6</cni><cbs>6</cbs><la>/CB00061/18644122219/ae-edu0/cnt-led/la</la><ol>/CB00061/18644122219/ae-edu0/cnt-led/ol</ol></m2m:cnt>
[sh_state] : delsub
x-m2m-rsc : 2002 <---->
[sh_state] : crtsub
x-m2m-rsc : 2001 <----<m2m:sub rn="sub-ctrl" xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><ty>23</ty><ri>SS0000000000000000000000000000000029</ri><pi>CT00000000000000000000000000000038</pi><ct>20160822T205712</ct><lt>20160822T205712</lt><et>99991231T000000</et><enc><net>3</net></enc><nu>mqt://oneM2M/req/+/SHJ-tF-Og000000000004931720896</nu><nct>2</nct></m2m:sub>
[sh_state] : crtci
TCP Server (192.168.0.34) is listening on port 3105
```

Thiết lập TAS bằng Raspberry Pi

Tạo thư mục TAS và cài đặt TAS

```
pi@raspberrypi:~/node$ mkdir tas_co2
pi@raspberrypi:~/node$ cd tas_co2
pi@raspberrypi:~/node/tas_co2$ unzip keti_tas_thyme-v1.X.zip
pi@raspberrypi:~/node/tas_co2$ sudo npm install
```

```
mqtt@1.14.0 node_modules/mqtt
└── inherits@2.0.1
  ├── reinterval@1.1.0
  ├── xtend@4.0.1
  ├── help-me@0.1.0
  ├── minimist@1.2.0
  ├── readable-stream@1.0.34 (string_decoder@0.10.31, isarray@0.0.1, core-util-is@1.0.2)
  ├── commist@1.0.0 (leven@1.0.2)
  ├── mqtt-connection@2.1.1 (through2@0.6.5, reduplexer@1.1.0)
  ├── mqtt-packet@3.4.7 (bl@0.9.5)
  ├── end-of-stream@1.1.0 (once@1.3.3)
  ├── pump@1.0.1 (once@1.3.3)
  ├── concat-stream@1.5.1 (typedarray@0.0.6, readable-stream@2.0.6)
  ├── split2@2.1.0 (through2@2.0.1)
  └── websocket-stream@3.2.1 (ws@1.1.1, through2@2.0.1, duplexify@3.4.5)

xmlbuilder@2.6.5 node_modules/xmlbuilder
└── lodash@3.10.1

xml2js@0.4.17 node_modules/xml2js
└── sax@1.2.1
    └── xmlbuilder@4.2.1 (lodash@4.15.0)
```

Định cấu hình TAS

Mở tệp cấu hình và chỉnh sửa

```
pi@raspberrypi:~/node/tas_co2 $ sudo nano conf.json
..... (Chỉnh sửa tệp cấu hình)
<Ctrl>+<X> → Y → <Enter>

pi@raspberrypi:~/node/thyme_tas
GNU nano 2.2.6                               File: conf.xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:conf xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <tas>
    <comport>/dev/ttyUSB0</comport>
    <baudrate>115200</baudrate>
    <parenthostname>localhost</parenthostname>
    <parentport>3105</parentport>
  </tas>
  <upload>
    <ctname>ss_1</ctname>
    <id>fe80:0000:0000:0000:0212:4b00:0235:bc89</id>
  </upload>
  <download>
    <ctname>ss_1_ctrl</ctname>
    <id>fe80:0000:0000:0000:0212:4b00:0235:bc89</id>
  </download>
</m2m:conf>
```

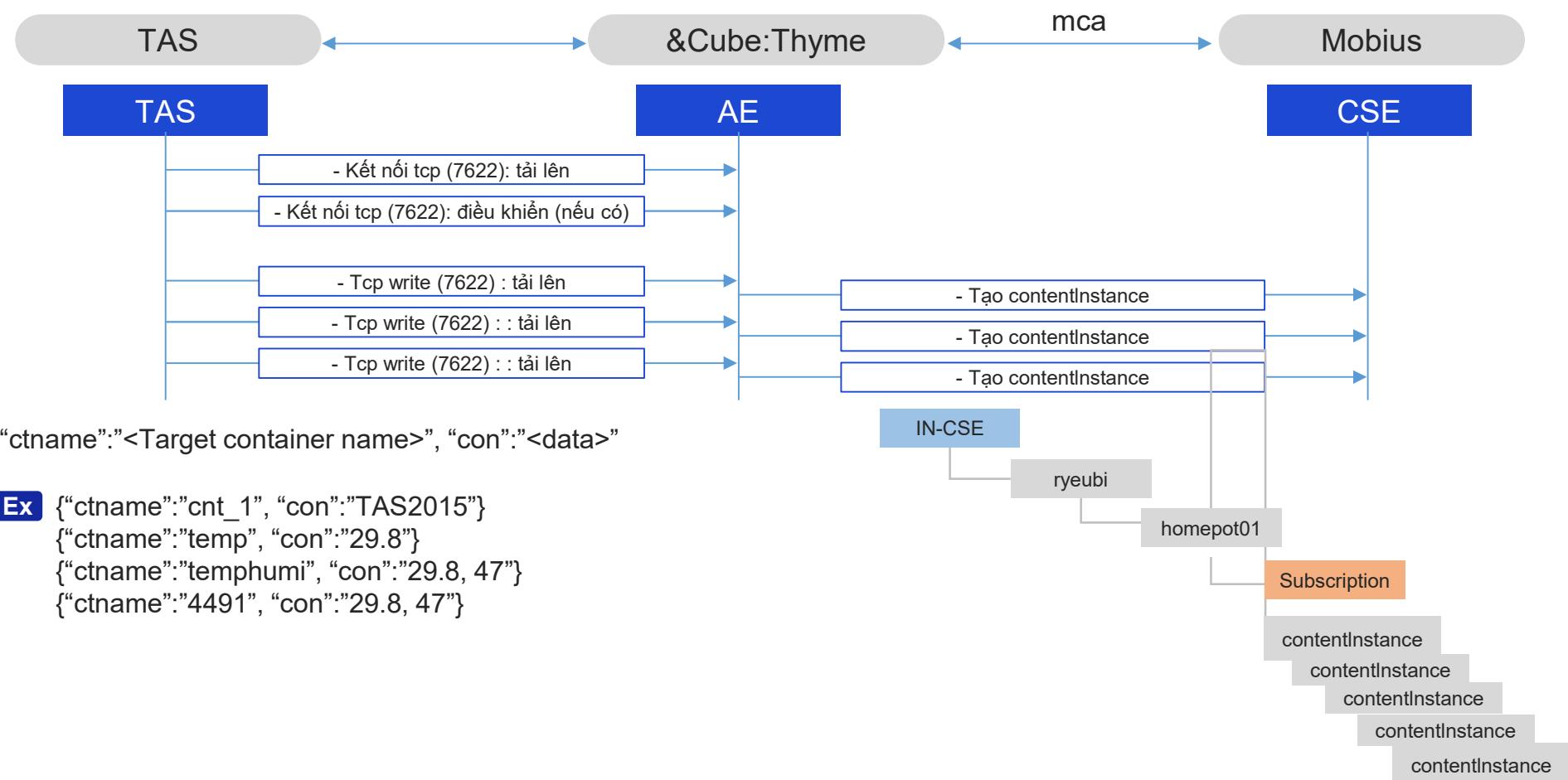
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:conf
  xmlns:m2m="http://www.onem2m.org/xml/protocols"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <tas>
    <comport>/dev/ttyUSB0</comport>
    <baudrate>9600</baudrate>
    <parenthostname>localhost</parenthostname>
    <parentport>7622</parentport>
  </tas>
  <upload>
    <ctname>cnt-co2</ctname>
    <id>fe80:0000:0000:0000:0212:4b00:0235:bce5</id>
  </upload>
</m2m:conf>
```

Định cấu hình &Cube: Thyme

Các tham số

Tham số	Mô tả	Nội dung
tas	Khu vực cấu hình của Phần mềm thích ứng sự vật	Thẻ XML cấp thấp
cổng thông tin	Định cấu hình tên cổng nối tiếp của thiết bị	Chuỗi (ví dụ về Linux: “/dev/ttyUSB0”, ví dụ về Win: “COM0”)
tốc độ truyền	Định cấu hình tốc độ truyền cổng nối tiếp của thiết bị	Chuỗi (ví dụ: “115200”)
tên máy chủ cha	Định cấu hình số ID máy chủ TCP của Thyme	Chuỗi (ví dụ: “127.0..0.1”)
cổng giao tiếp cha	Định cấu hình số cổng máy chủ TCP của Thyme	Chuỗi (ví dụ: “3105”)
tải lên	Khu vực cấu hình tải lên dữ liệu của cảm biến	Thẻ XML cấp thấp
tải xuống	Khu vực cấu hình tải xuống lệnh của bộ dẫn động	Thẻ XML cấp thấp
ctname	Định cấu hình tên vùng chứa để liên kết với thyme	Chuỗi (ví dụ: temp)
id	Định cấu hình ID thiết bị để liên kết với vùng chứa dữ liệu	Chuỗi

Sơ đồ TAS (Phần mềm thích ứng vật thể)



Chạy TAS

Nhập lệnh chạy TAS

```
pi@raspberrypi:~/node/tas_co2$ node app.js
```

```
pi@raspberrypi:~/node/thyme_tas$ node app.js
port open. Data rate: 115200
1
{"con":"0.00W,1,6245","id":"fe80:0000:0000:0000:0212:4b00:0235:bba0"}
tas_develop_test
2
{"con":"0.00W,1,6246","id":"fe80:0000:0000:0000:0212:4b00:0235:bba0"}

tas_develop_test
3
upload Connected
download Connected - control_test_container hello
reconnect
Received: {"ctname":"control test container","con":"hello"}
{"con":"0.00W,1,6247","id":"fe80:0000:0000:0000:0212:4b00:0235:bba0"}

tas_develop_test
ACK : {"ctname":"test_container","con":"2001"} <----
4
{"con":"0.00W,1,6248","id":"fe80:0000:0000:0000:0212:4b00:0235:bba0"}

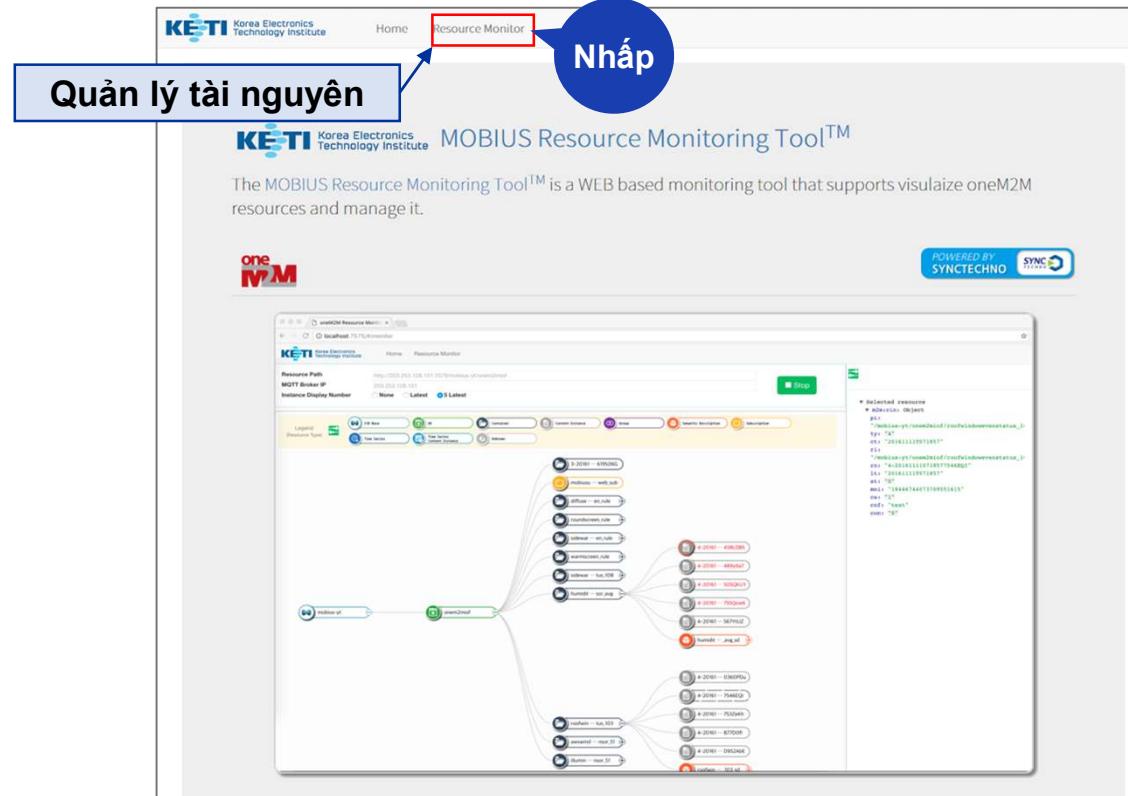
tas_develop_test
ACK : {"ctname":"test_container","con":"2001"} <----
5
{"con":"0.00W,1,6249","id":"fe80:0000:0000:0000:0212:4b00:0235:bba0"}

tas_develop_test
ACK : {"ctname":"test_container","con":"2001"} <----
```

Trình xem tài nguyên Mobius

http://203.253.128.161:7575

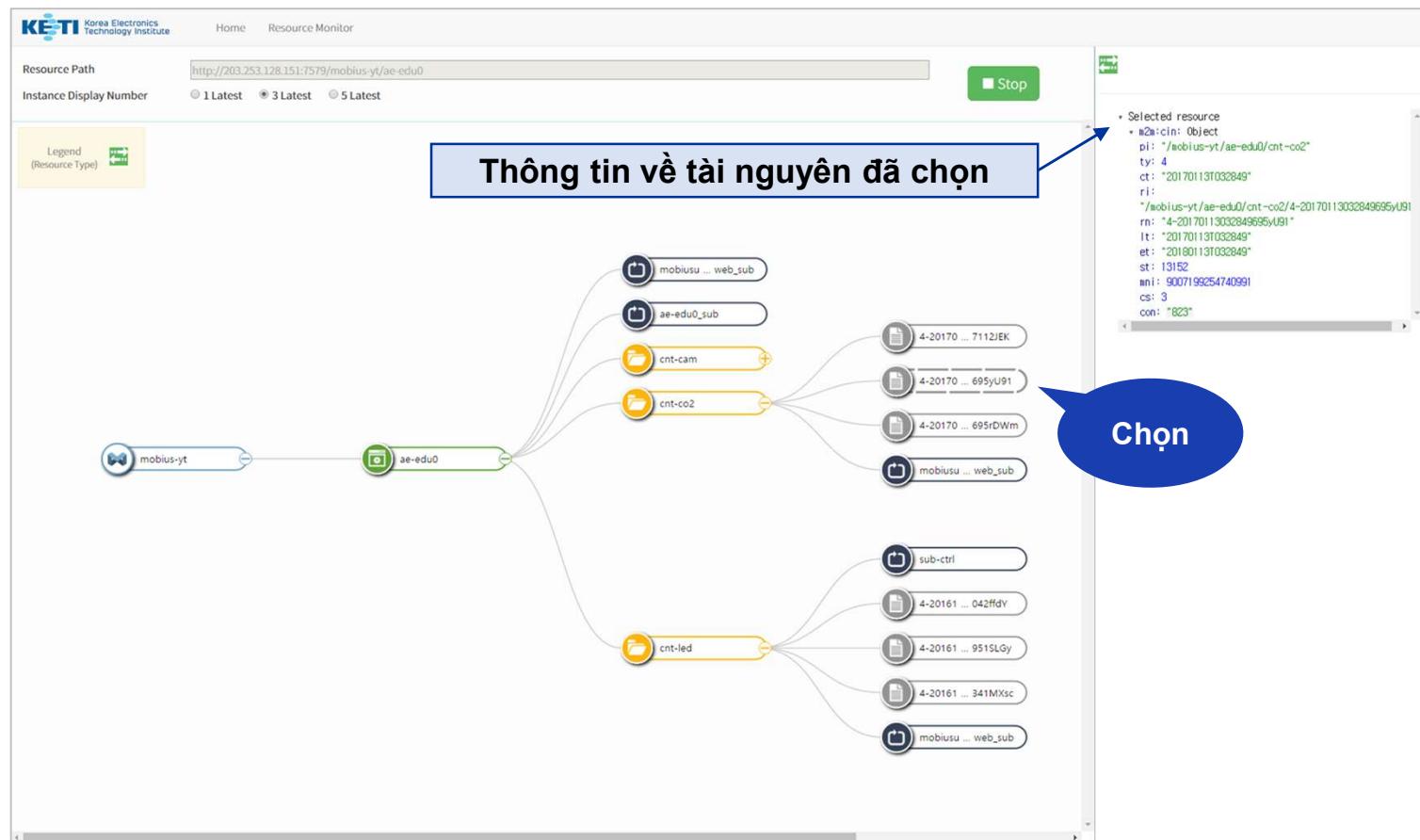
| Trình xem tài nguyên trang web để xem trạng thái tài nguyên của AE bạn tạo



Trình xem tài nguyên Mobius

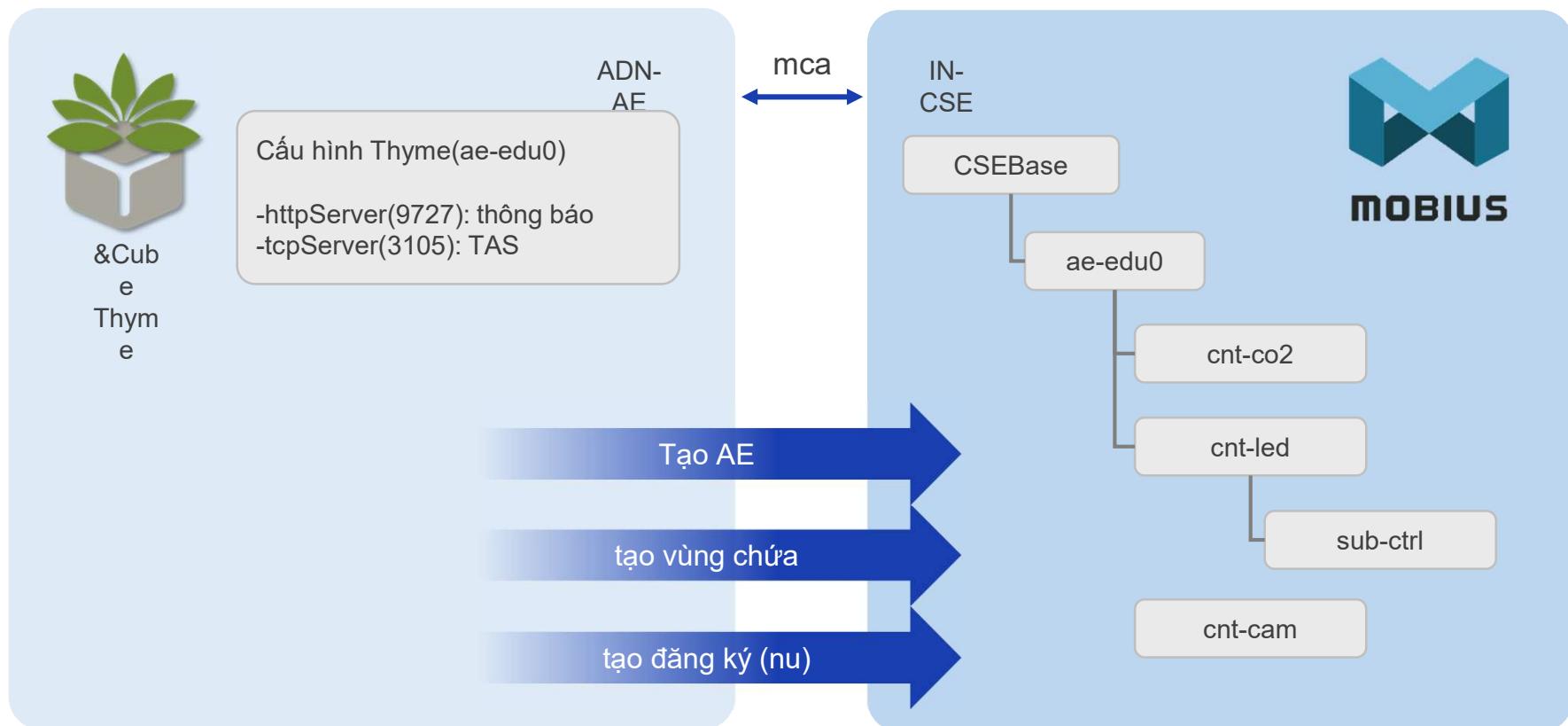


Trình xem tài nguyên Mobius



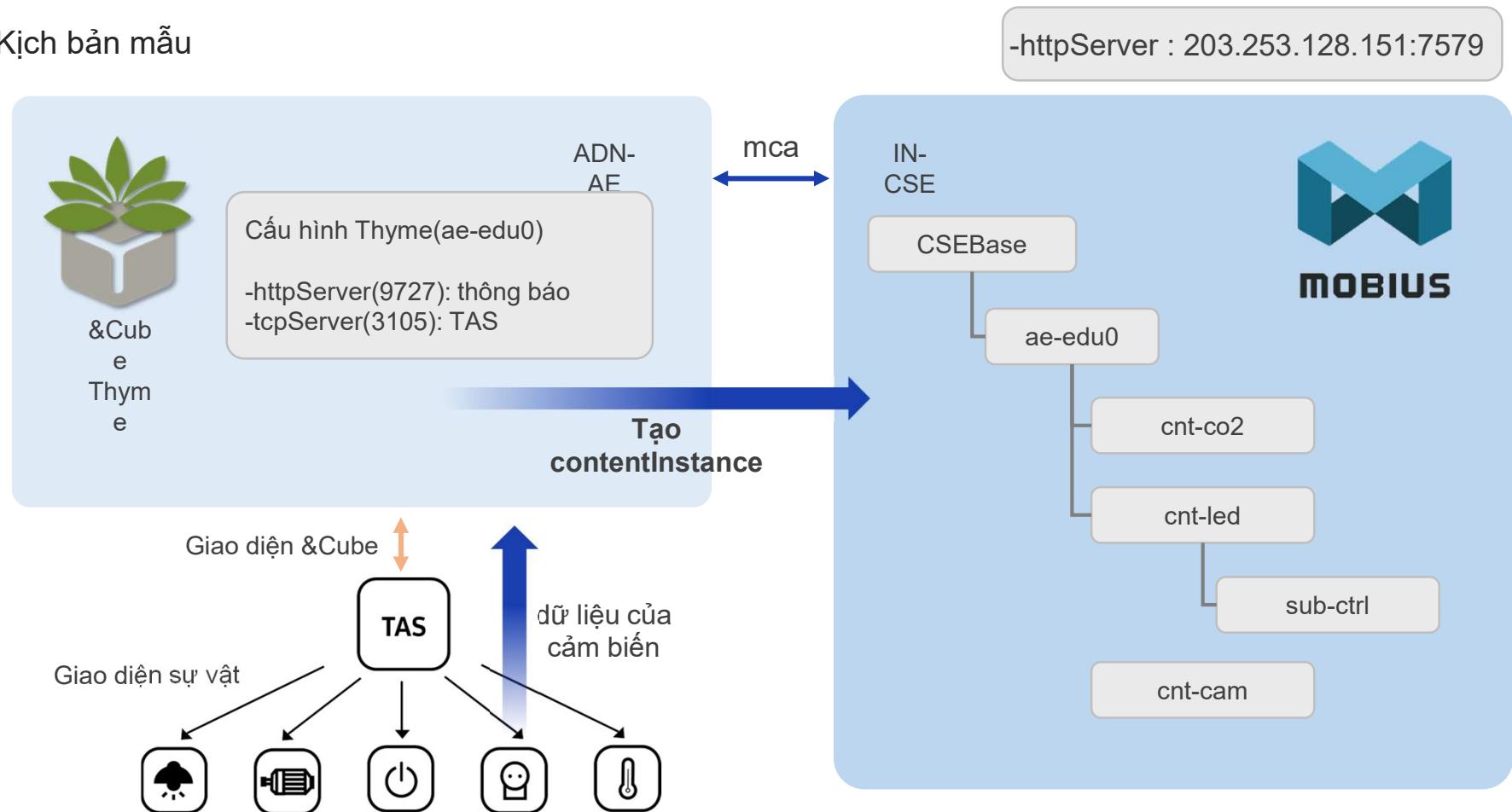
Chạy &Cube: Thyme

I Kịch bản mẫu

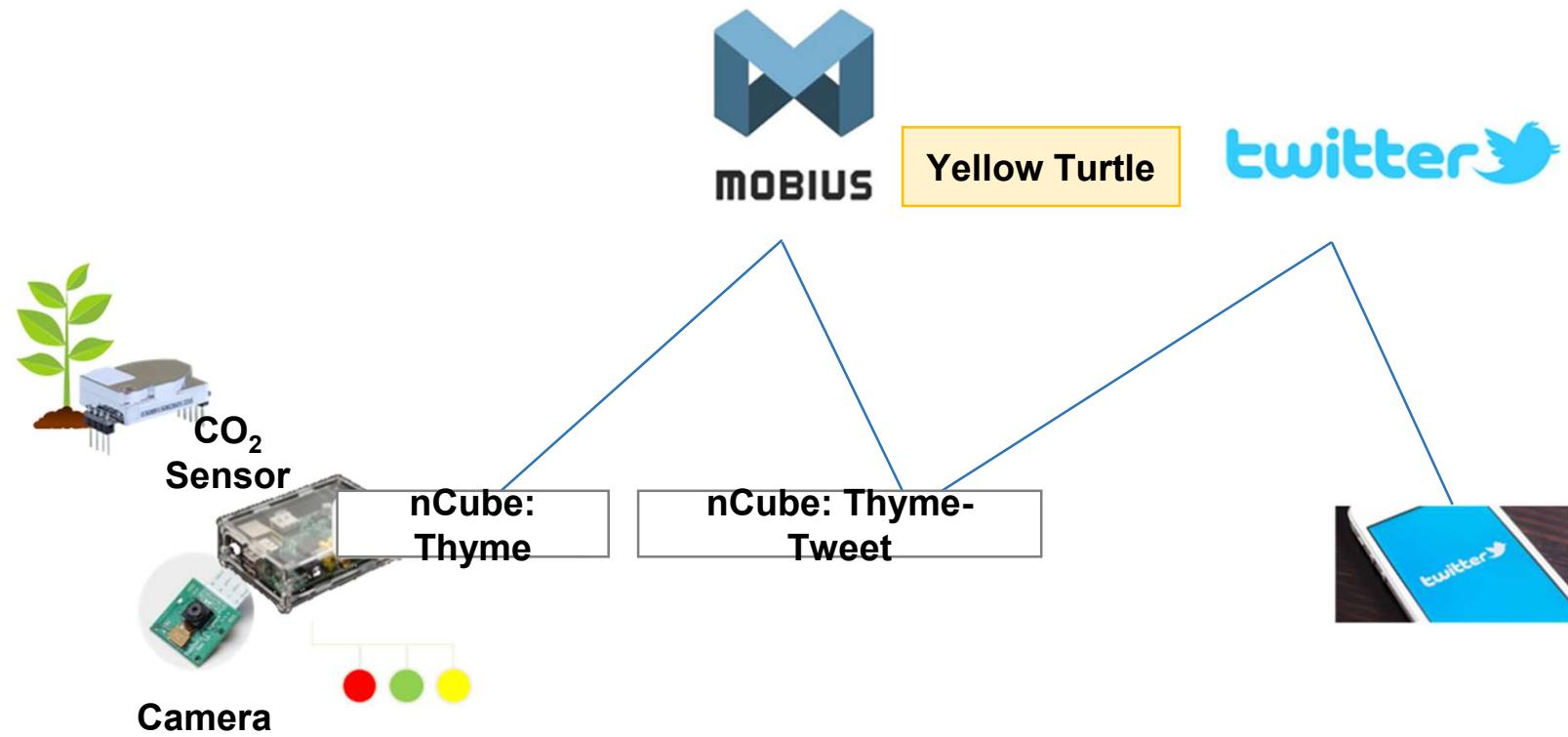


Chạy TAS

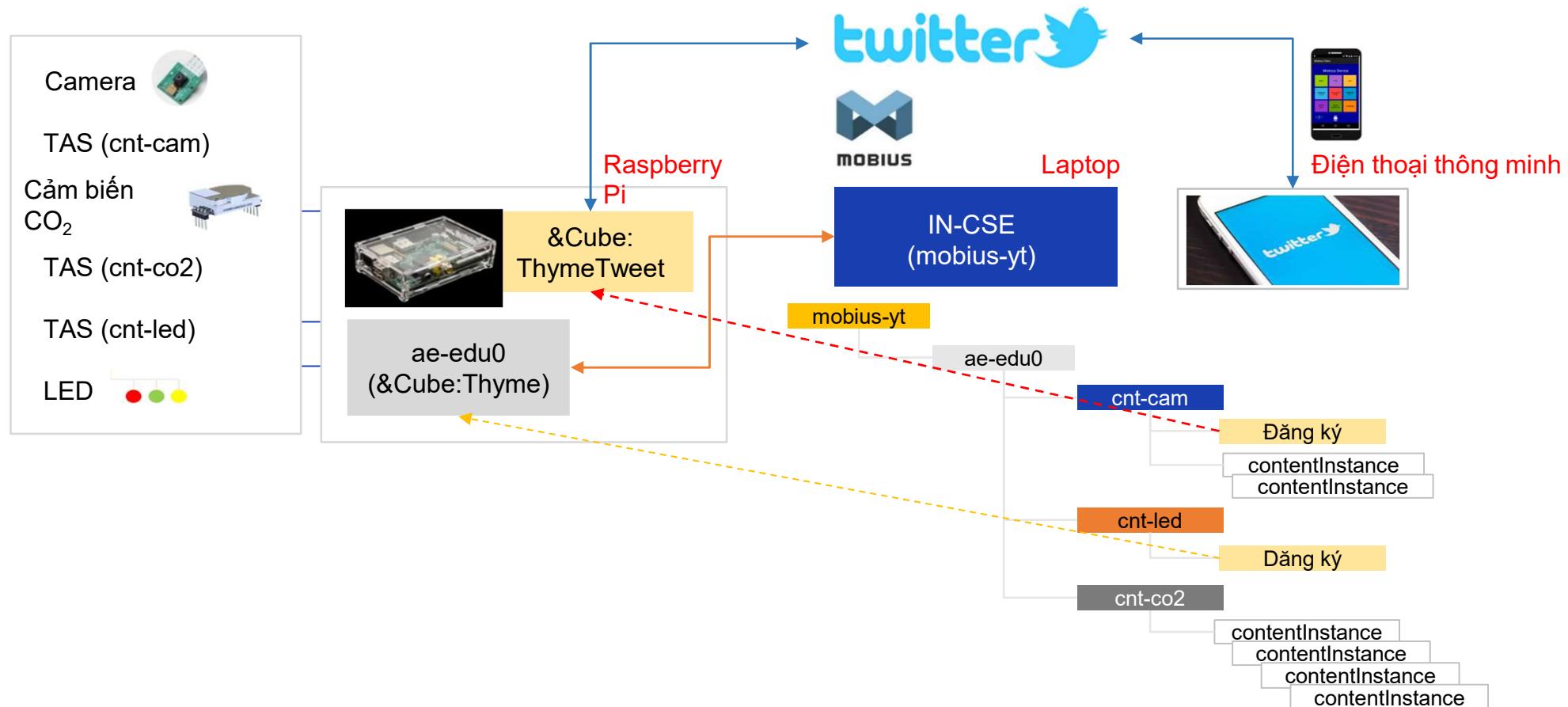
I Kịch bản mẫu



Nghiên cứu trường hợp của ToT (Tweet vạn vật)



Cấu hình của ToT



Bài 2.

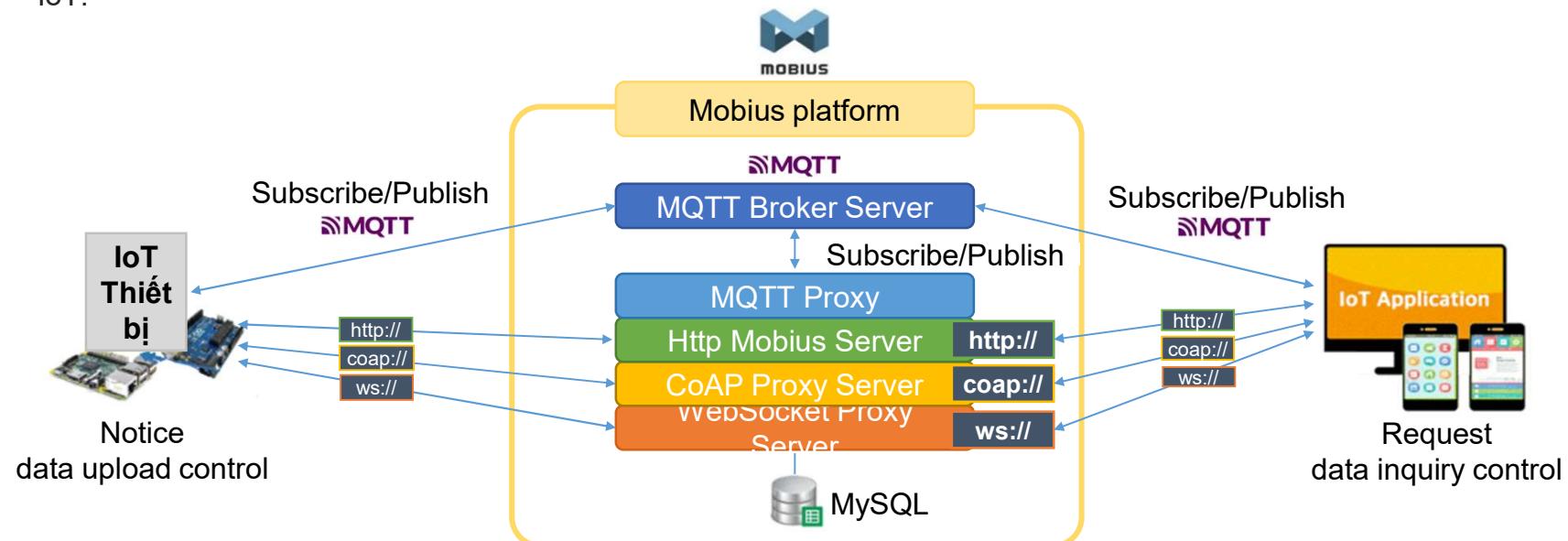
Nền tảng mở IoT Cài đặt và vận hành Mobius

- | 2.1. Tổng quan về nền tảng Mobius
- | 2.2. Thiết lập môi trường **dịch vụ**

Nền tảng mở IoT: tổng quan về Mobius

Cấu hình ứng dụng nền tảng Mobius

- Nền tảng máy chủ mở (middleware) cung cấp Ứng dụng IoT dựa trên chuẩn quốc tế oneM2M.
 - Cung cấp hỗ trợ CSF (Chức năng ứng dụng chung) và REST API để kiểm soát truy cập, xác thực và quản lý dữ liệu của Thiết bị IoT.
- Cấu hình nền tảng máy chủ mở
 - Tính năng chính: Hoạt động như một cầu nối trung gian cho phép giao tiếp và tương tác giữa các thiết bị và ứng dụng IoT.



Nền tảng mở IoT: tổng quan về Mobius

| Danh sách Ứng dụng



Tra cứu trạng thái
Mobius



Quản lý thiết bị



Quản lý quyền
truy cập



Trình khám phá
dữ liệu



Trình duyệt tài
nguyên



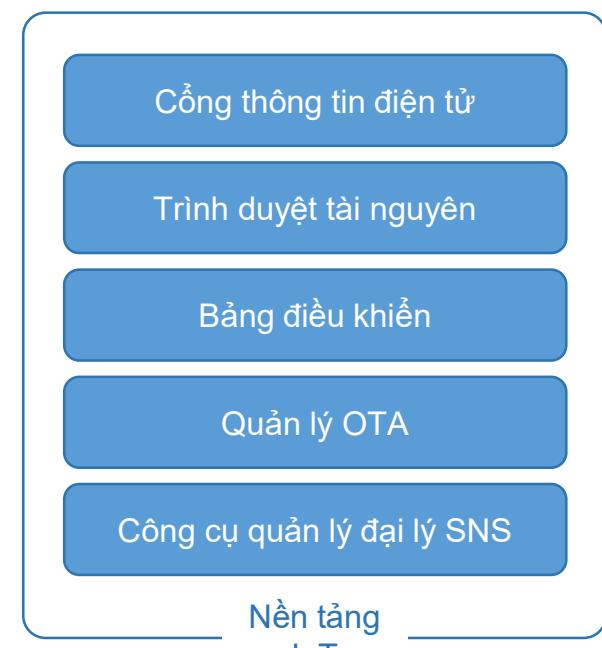
Bảng điều khiển



Quản lý OTA



Đại lý quản lý



Nền tảng mở IoT: tổng quan về Mobius

| Ứng dụng cung cấp bởi Mobius

Ứng dụng	Mô tả
Tra cứu trạng thái Mobius	Một ứng dụng giúp hiển thị trạng thái Mobius hiện tại
Quản lý thiết bị	Một ứng dụng thêm và quản lý Thiết bị đã kết nối với Mobius
Quản lý quyền truy cập	Một Ứng dụng kiểm soát quyền truy cập vào tài nguyên Mobius do người dùng đăng ký
Trình khám phá dữ liệu	Một ứng dụng giúp truy vấn dữ liệu của Thiết bị do người dùng đăng ký
Trình duyệt tài nguyên	Tính năng giúp trực quan hóa cấu trúc tài nguyên Mobius và hiển thị nó dưới dạng sơ đồ cây
Bảng điều khiển	Một ứng dụng hiển thị trạng thái tài nguyên của Mobius dưới dạng tiện ích con theo các điều kiện do người dùng tạo ra
Quản lý OTA	Cung cấp các bản cập nhật chương trình cơ sở từ xa cho nCubes có kết nối với Mobius
Đại lý quản lý	Một ứng dụng để đăng ký, tạo và quản lý lệnh cho Telegram Bots có thể tương tác với Mobius

Nền tảng mở IoT: tổng quan về Mobius

- | Cài đặt biến môi trường (nội dung sẽ được phản ánh trong quá trình cài đặt)
- | Tải xuống URL

mobius_open_iot : <https://github.com/IoTKETI/mobius-open-platform>

- | Cổng mặc định cho mỗi ứng dụng

Cấu hình ứng dụng	Cổng
Cổng thông tin điện tử	8881
Bảng điều khiển	7580
Trình duyệt tài nguyên	7575
Công cụ quản lý đại lý SNS	7597
Công cụ quản lý OTA	8730

Nền tảng mở IoT: tổng quan về Mobius

- | Cài đặt biến môi trường (nội dung sẽ được phản ánh trong quá trình cài đặt)
- | Tải xuống URL

mobius_open_iot : <https://github.com/IoTKETI/mobius-open-platform>

- | Hệ điều hành và phiên bản chương trình (nó là mã nguồn mở nên có thể thay đổi phiên bản mới nhất)

OS	Linux ubuntu 18.04 LTS	Vagrant	2.2.14
Node.js	v14.15.3	Virtual box	6.1
npm	v6.14.9	Nginx	1.19.6
mongoDB	4.0.22	PM2	4.5.1
Gulp-cli	2.3.0		

Nền tảng mở IoT: tổng quan về Mobius

| Thiết lập tên miền máy chủ ứng dụng

Tên miền máy chủ: **openmobius.org**

- Nếu bạn không thiết lập tên miền, bạn không thể đăng nhập vào Ứng dụng. **localhost** không khả dụng

Bài 2.

Nền tảng mở IoT Cài đặt và vận hành Mobius

- | 2.1. Tổng quan về nền tảng Mobius
- | 2.2. Thiết lập môi trường **dịch vụ**

Dịch vụ xây dựng môi trường theo yêu cầu

- | Cài đặt máy ảo (Virtualbox, Vmware,...)
- | Cài đặt Vagrant
- | Cài đặt Node.js
- | Cài đặt cơ sở dữ liệu (Maria DB, MySQL, Mongo, ...)
- | Cài đặt Gulp & PM2
- | Cài đặt máy chủ web Nginx

Tổ chức môi trường máy ảo bằng mã nguồn mở

| Cấu hình môi trường máy ảo

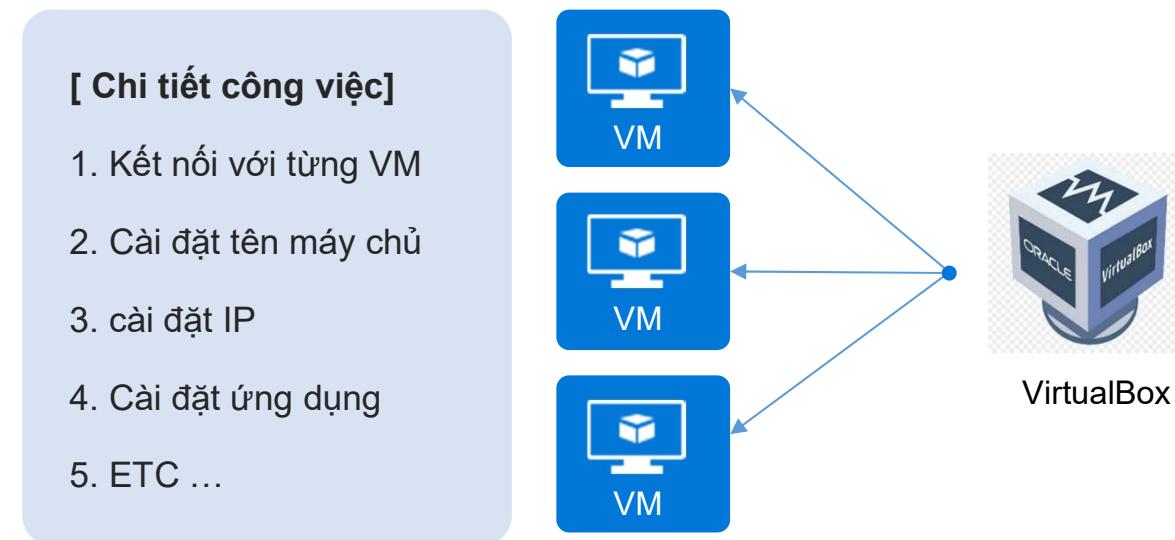
- Hãy định cấu hình CentOS, Ubuntu và các môi trường máy ảo khác nhau bằng các công cụ mã nguồn mở VirtualBox (Oracle) và Vagrant (HashCorp).



Tổ chức môi trường máy ảo bằng mã nguồn mở

| Cách tạo và quản lý máy ảo không dùng Vagrant

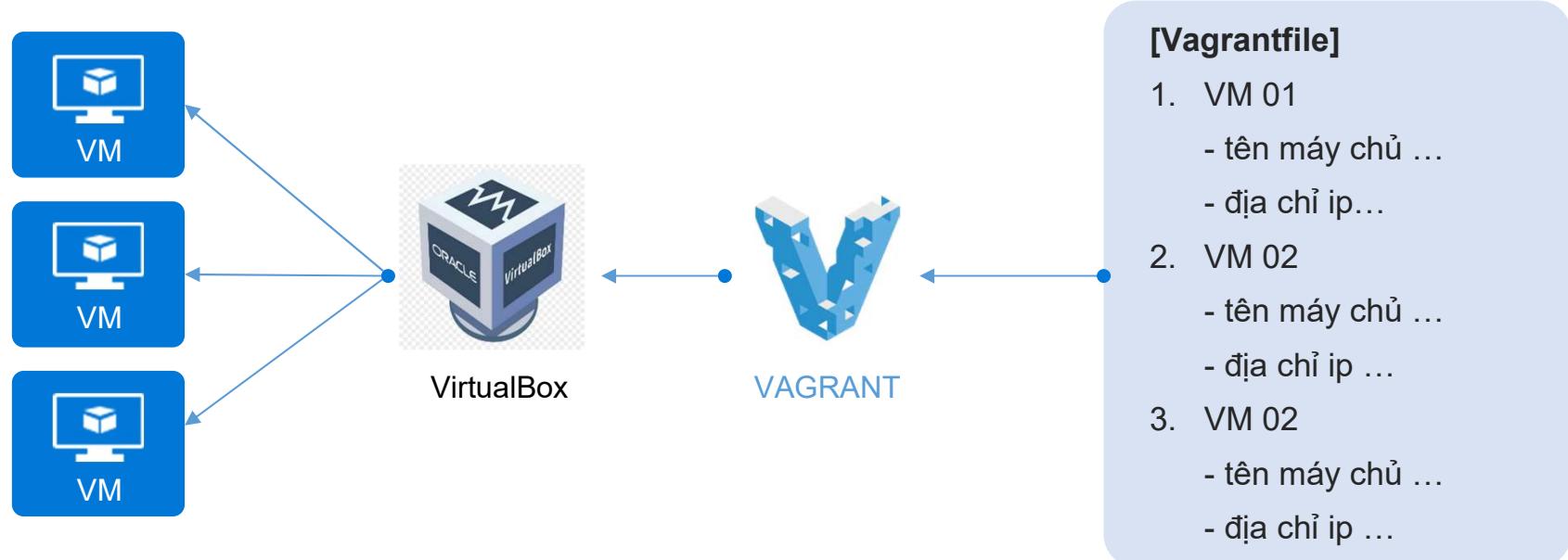
- Trong VirtualBox, từng máy ảo riêng lẻ phải được tạo, kết nối và thiết lập riêng.



Tổ chức môi trường máy ảo bằng mã nguồn mở

Cách tạo và quản lý máy ảo bằng Vagrant

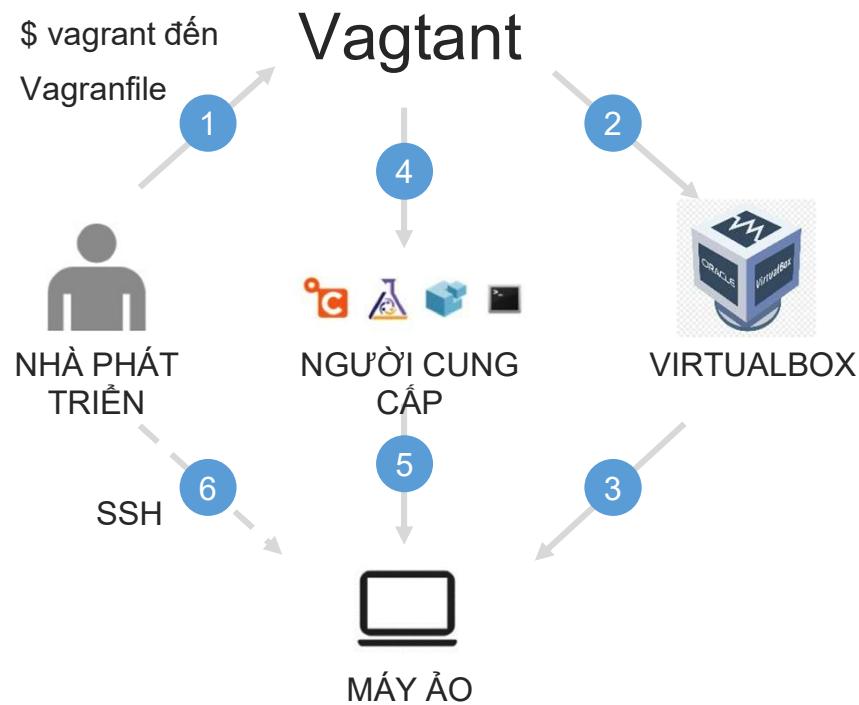
- Sau khi xác định cài đặt máy ảo và các tác vụ sẽ được thực hiện trong tệp tương ứng thông qua Vagrantfile, việc cung cấp có thể được thực hiện thông qua VirtualBox.
- Khi cung cấp máy ảo thông qua Vagrant, bạn có thể dễ dàng tạo, xóa, sửa đổi máy ảo.



Tổ chức môi trường máy ảo bằng mã nguồn mở

| Tại sao chúng ta lại cần môi trường ảo trong máy cục bộ?

- Trong VirtualBox, từng máy ảo riêng lẻ phải được tạo, kết nối và thiết lập riêng.



- Các môi trường do Vagrant quản lý có thể chạy trên nền tảng ảo hóa cục bộ như VirtualBox hoặc VMware, trên đám mây thông qua AWS hoặc OpenStack hoặc trong vùng chứa như Docker.
- Vagrant cung cấp một khuôn khổ và định dạng cấu hình để tạo và quản lý một môi trường phát triển hoàn toàn di động.
- Các môi trường phát triển này có thể nằm trên máy tính của bạn hoặc trên đám mây và nó có thể di động giữa Windows, Mac OS X và Linux.

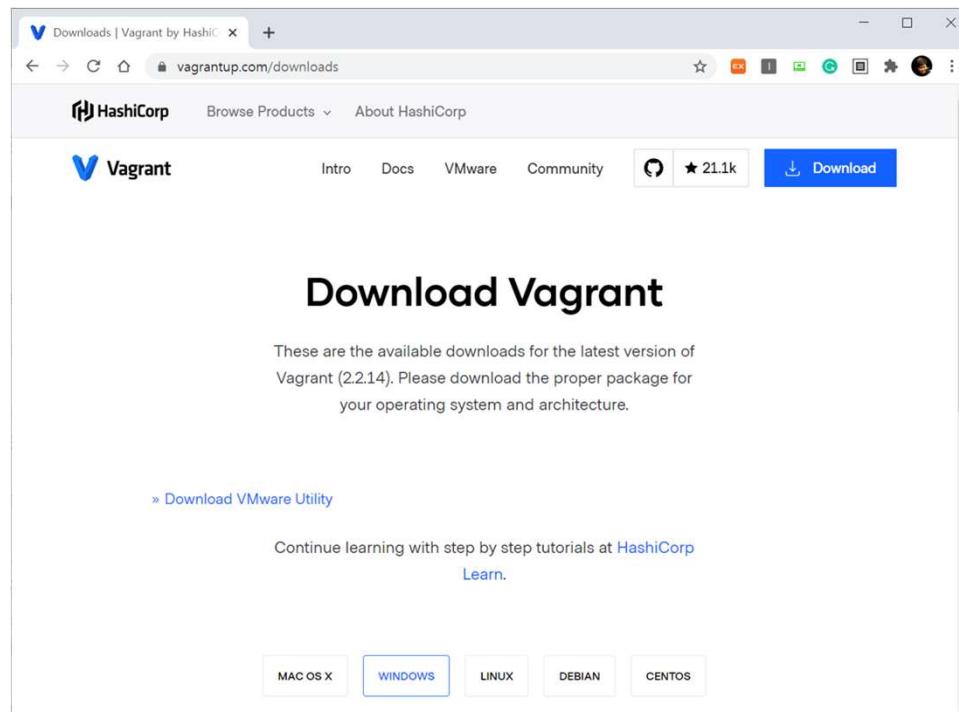
Các công cụ cung cấp là để sử dụng việc cung cấp (cài đặt phụ thuộc), tự động hóa, quản lý cấu hình và điều phối, đáp ứng tự duy và cách thức làm việc của DevOps.

Tổ chức môi trường máy ảo bằng mã nguồn mở

Cài đặt Vagrant

- ▶ Tải và cài đặt theo hệ điều hành sử dụng Vagrant

Vào trang web <https://www.vagrantup.com/downloads.html> để tải xuống.

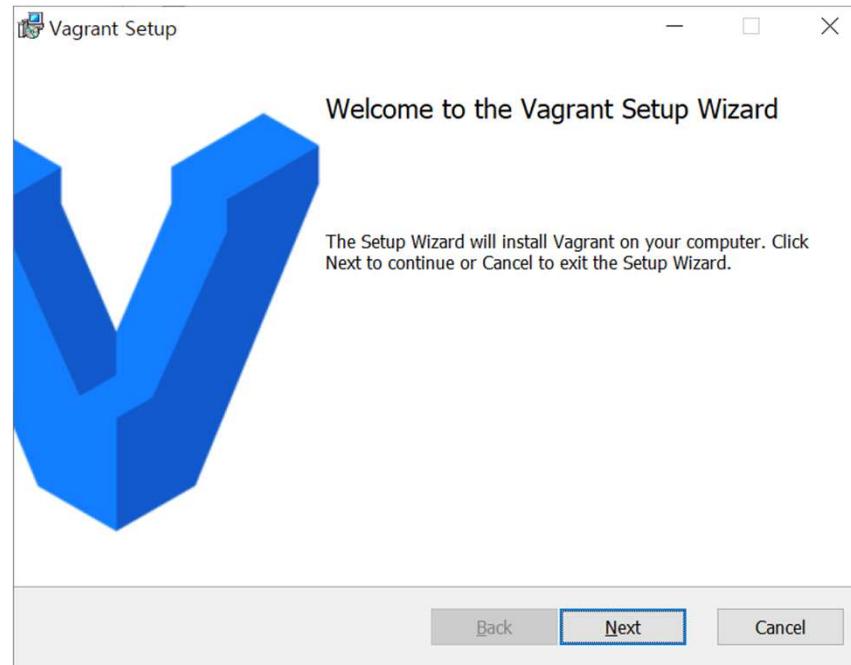


- ▶ Vagrant có thể dễ dàng cung cấp hệ điều hành và nó chủ yếu được sử dụng để tạo và quản lý các máy ảo.
- ▶ Nó là một công cụ cung cấp cài đặt cho các môi trường khác nhau như tên máy chủ, IP và Ứng dụng cho máy ảo theo nhu cầu của người dùng và cho phép người dùng sử dụng hệ thống ngay lập tức nếu họ muốn.

Tổ chức môi trường máy ảo bằng mã nguồn mở

| Cài đặt Vagrant

- ▶ Sau khi cài đặt VirtualBox và Vagrant, hãy chạy Windows Terminal hoặc Powershell và di chuyển đến thư mục đã cài đặt Vagrant
- ▶ Đường dẫn cài đặt mặc định của Vagrant: **C:\HashiCorp**



Tổ chức môi trường máy ảo bằng mã nguồn mở

| Khởi chạy Vagrant

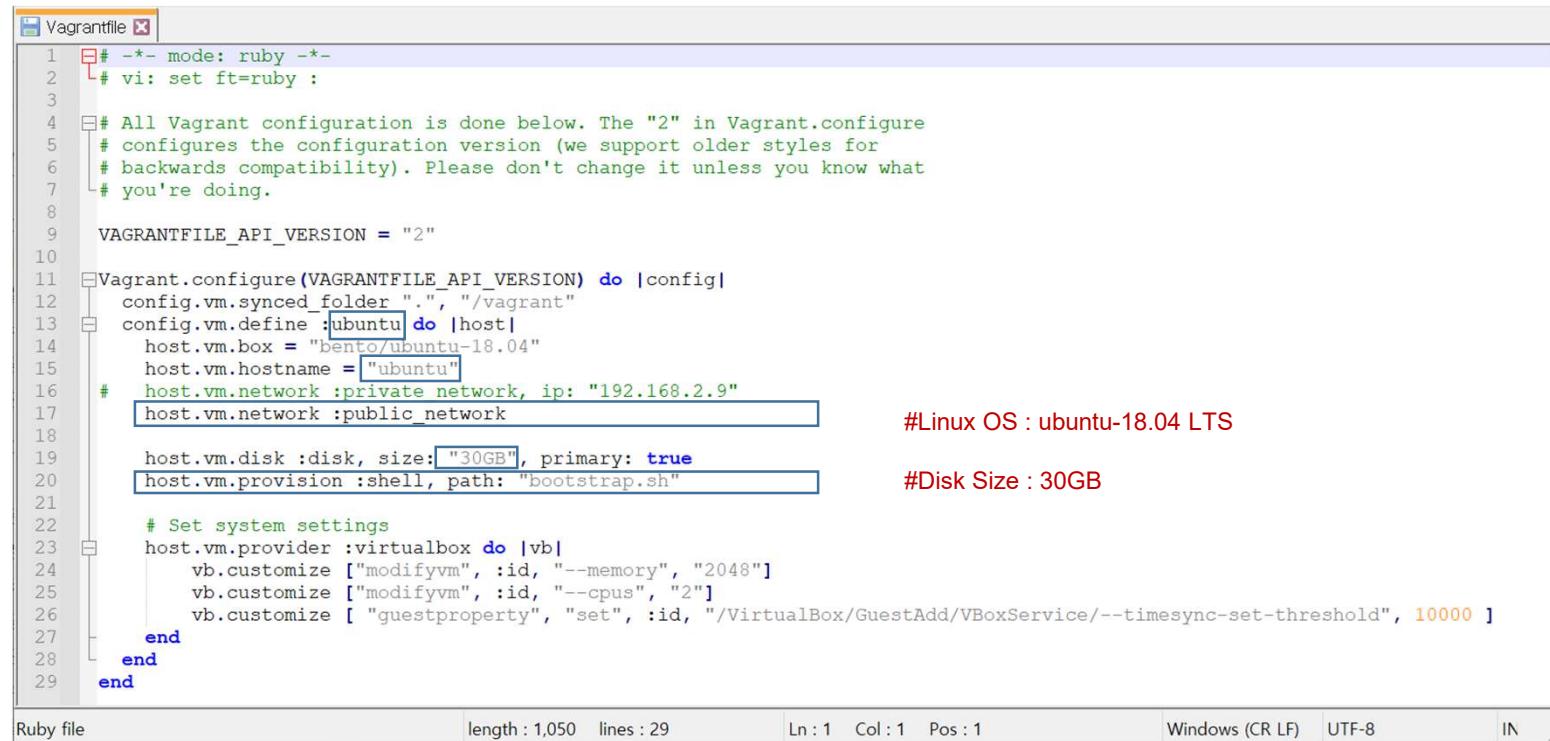
- Cách sử dụng Vagrant
 - Khi đã thành công cài đặt vagrant, mở cửa sổ Lệnh (Powershell).
 - Nếu bạn gõ vagrant -help, lệnh sau xuất hiện.
- Các câu lệnh Vagrant

vagrant init	Tạo một tệp ban đầu (Vagrantfile) để cung cấp vagrant
vagrant up	Tiến hành cung cấp dựa trên Vagrantfile đã tạo
vagrant halt	Dừng hoạt động máy ảo do vagrant quản lý
vagrant destroy	Xóa máy ảo do vagrant quản lý
vagrant ssh	Kết nối qua ssh với máy ảo do vagrant quản lý
vagrant provision	Thay đổi và áp dụng cài đặt của máy ảo do vagrant quản lý

Tổ chức môi trường máy ảo bằng mã nguồn mở

Tạo Vagrantfile

- Thực thi vagrant –init → Vagrantfile được tạo trong thư mục C:\HasiCorp\
- Sửa đổi tệp Vagrantfile được tạo bằng vagrant -init như sau.



```
# -*- mode: ruby -*-
# vi: set ft=ruby :

# All Vagrant configuration is done below. The "2" in Vagrant.configure
# configures the configuration version (we support older styles for
# backwards compatibility). Please don't change it unless you know what
# you're doing.

VAGRANTFILE_API_VERSION = "2"

Vagrant.configure(VAGRANTFILE_API_VERSION) do |config|
  config.vm.synced_folder ".", "/vagrant"
  config.vm.define :ubuntu do |host|
    host.vm.box = "bento/ubuntu-18.04"
    host.vm.hostname = "ubuntu"
    host.vm.network :private_network, ip: "192.168.2.9"
    host.vm.network :public_network
  end
  host.vm.disk :disk, size: "30GB", primary: true
  host.vm.provision :shell, path: "bootstrap.sh"
end

# Set system settings
host.vm.provider :virtualbox do |vb|
  vb.customize ["modifyvm", :id, "--memory", "2048"]
  vb.customize ["modifyvm", :id, "--cpus", "2"]
  vb.customize [ "guestproperty", "set", :id, "/VirtualBox/GuestAdd/VBoxService/--timesync-set-threshold", "10000" ]
end
end
end

Ruby file length : 1,050 lines : 29 Ln : 1 Col : 1 Pos : 1 Windows (CR LF) UTF-8 IN .vagrantfile
```

#Linux OS : ubuntu-18.04 LTS
#Disk Size : 30GB

Tổ chức môi trường máy ảo bằng mã nguồn mở

| Tạo bootstrap.sh



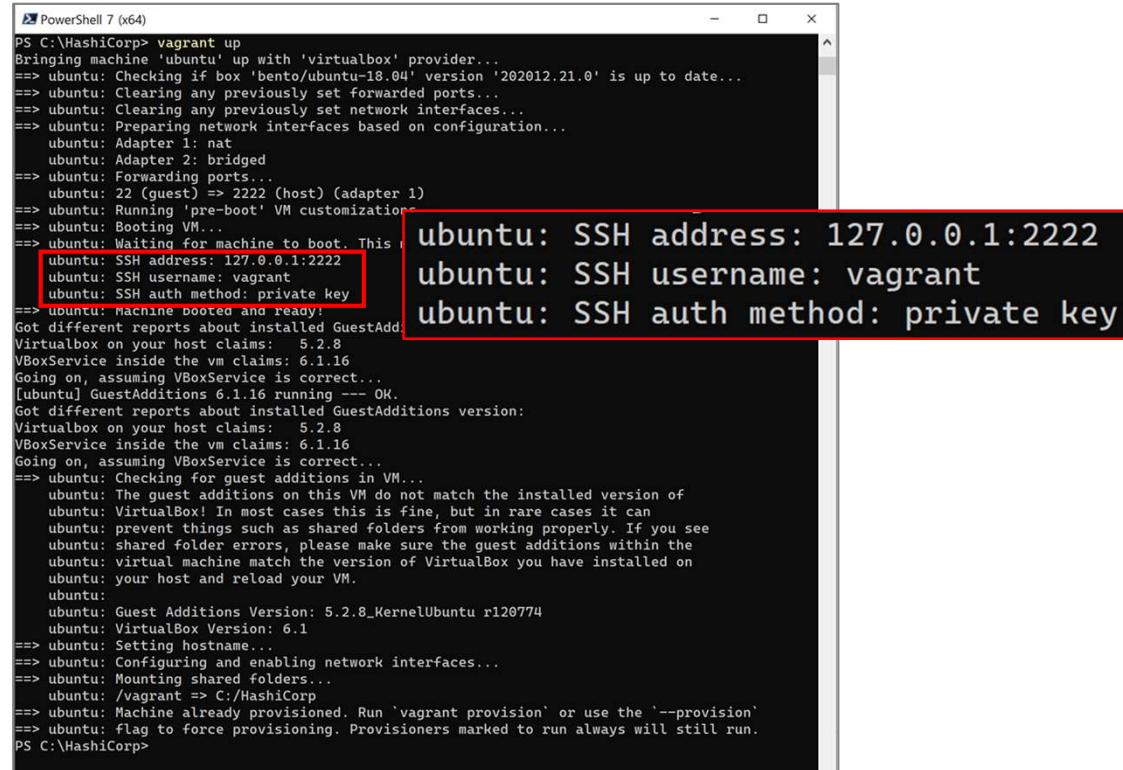
```
#!/bin/bash
set -e
rm -rf /var/lib/apt/lists/*
sed -i 's/archive.ubuntu.com/ftp.daum.net/g' /etc/apt/sources.list
apt-get update -y
```

Unix script file length : 141 lines : 8 Ln : 1 Col : 1 Pos : 1 Windows (CR LF) UTF-8 IN

Tổ chức môi trường máy ảo bằng mã nguồn mở

Khởi chạy Vagrant

- Vagrant up
 - C:\HashiCorp> vagrant up



```
PS C:\HashiCorp> vagrant up
Bringing machine 'ubuntu' up with 'virtualbox' provider...
==> ubuntu: Checking if box 'bento/ubuntu-18.04' version '202012.21.0' is up to date...
==> ubuntu: Clearing any previously set forwarded ports...
==> ubuntu: Clearing any previously set network interfaces...
==> ubuntu: Preparing network interfaces based on configuration...
    ubuntu: Adapter 1: nat
    ubuntu: Adapter 2: bridged
==> ubuntu: Forwarding ports...
    ubuntu: 22 (guest) => 2222 (host) (adapter 1)
==> ubuntu: Running 'pre-boot' VM customizations...
==> ubuntu: Booting VM...
==> ubuntu: Waiting for machine to boot. This may take a few minutes...
    ubuntu: SSH address: 127.0.0.1:2222
    ubuntu: SSH username: vagrant
    ubuntu: SSH auth method: private key
==> ubuntu: Machine booted and ready!
Got different reports about installed GuestAdditions version:
Virtualbox on your host claims: 5.2.8
VBoxService inside the vm claims: 6.1.16
Going on, assuming VBoxService is correct...
[ubuntu] GuestAdditions 6.1.16 running --- OK.
Got different reports about installed GuestAdditions version:
Virtualbox on your host claims: 5.2.8
VBoxService inside the vm claims: 6.1.16
Going on, assuming VBoxService is correct...
==> ubuntu: Checking for guest additions in VM...
    ubuntu: The guest additions on this VM do not match the installed version of
    ubuntu: VirtualBox! In most cases this is fine, but in rare cases it can
    ubuntu: prevent things such as shared folders from working properly. If you see
    ubuntu: shared folder errors, please make sure the guest additions within the
    ubuntu: virtual machine match the version of VirtualBox you have installed on
    ubuntu: your host and reload your VM.
    ubuntu:
    ubuntu: Guest Additions Version: 5.2.8_KernelUbuntu r120774
    ubuntu: VirtualBox Version: 6.1
==> ubuntu: Setting hostname...
==> ubuntu: Configuring and enabling network interfaces...
==> ubuntu: Mounting shared folders...
    ubuntu: /vagrant => C:/HashiCorp
==> ubuntu: Machine already provisioned. Run 'vagrant provision' or use the '--provision'
==> ubuntu: flag to force provisioning. Provisioners marked to run always will still run.
PS C:\HashiCorp>
```

The highlighted text shows the SSH configuration details:

- ubuntu: SSH address: 127.0.0.1:2222
- ubuntu: SSH username: vagrant
- ubuntu: SSH auth method: private key

Tổ chức môi trường máy ảo bằng mã nguồn mở

Khởi chạy Vagrant

- Vagrant ssh
 - C:\HashiCorp> vagrant ssh

```
vagrant@ubuntu: ~
PS C:\HashiCorp> vagrant ssh
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-128-generic x86_64)

Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-128-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri Jan  1 13:06:08 UTC 2021

System load:  0.0      Processes:          100
Usage of /:   4.1% of 61.80GB  Users logged in:     0
Memory usage: 6%
Swap usage:   0%          IP address for eth0: 10.0.2.15
                           IP address for eth1: 192.168.0.11

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
  https://ubuntu.com/livepatch

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento
Last login: Mon Dec 28 17:32:52 2020 from 10.0.2.2
vagrant@ubuntu:~$
```

```
선택 root@ubuntu: /home/vagrant
vagrant@ubuntu:~$ sudo su
root@ubuntu:/home/vagrant#
```

- Bạn có thể thấy rằng môi trường ảo cục bộ được kết nối trực tiếp với môi trường phát triển Ubuntu.

- Nếu bạn muốn làm việc với tài khoản người dùng siêu cấp có đặc quyền cao, hãy nhập tài khoản người dùng siêu cấp thông qua lệnh sudo su.

Tổng quan về SSH

| Cách thức hoạt động của SSH

- SSH (Secure Shell) là phiên bản bảo mật của chương trình ứng dụng Telnet được phát triển để giải quyết các sự cố bảo mật xảy ra trong quá trình đăng nhập từ xa.
- Theo mặc định, cổng 22 được sử dụng. Ngay cả khi nội dung giao tiếp được hiển thị thông qua phương pháp mã hóa, chỉ những ký tự được mã hóa không thể hiểu được mới được hiển thị.
- ssh sử dụng hai giao thức là ssh1 và ssh2.
 - ssh1 = chỉ hỗ trợ các khóa RSA, Mã hóa khóa phiên bằng khóa chung
 - ssh2 = hỗ trợ cả RSA và DSA, Chọn theo máy chủ của bạn, phương pháp mã hóa Diffie-Hellman được sử dụng.

4 đặc tính của SSH

Tính xác thực

- Người dùng phải chứng minh danh tính của họ khi truy cập máy chủ SSH. Ngoài xác thực mật khẩu truyền thống, các phương thức xác thực khóa công khai như RSA và DSA cũng được hỗ trợ.
- Bằng cách lưu trữ khóa của máy chủ được kết nối đầu tiên, máy khách SSH có thể xác định ngay cả khi máy chủ bị thay đổi.

Tính mã hóa

- SSH mã hóa dữ liệu được truyền qua mạng. Ngay cả khi ai đó can thiệp dữ liệu ở giữa quá trình, nội dung vẫn được giữ kín.
- Nó được thiết kế để cung cấp phương thức mã hóa khóa đối xứng và thêm một phương thức mã hóa mới.

Sự toàn vẹn

- Đảm bảo dữ liệu nhận qua mạng không bị thay đổi.
- Nó có thể ngăn ai đó chặn dữ liệu của bạn và gửi thông tin khác. Nó được thực hiện thông qua MAC (Mã xác thực thư).

Nén

- Nó có thể nén dữ liệu được gửi qua kết nối SSH.
- Đối với việc nhận dữ liệu, dữ liệu được giải mã và sau đó sẽ được giải nén.

Cài đặt Gulp

Tổng quan về Gulp.js

- Gulp là một trình chạy tác vụ dựa trên Node.js và nó là một thư viện được sử dụng để cài đặt nền tảng này một cách đơn giản.
- Gulp có thể dễ dàng xử lý các tác vụ lặp đi lặp lại và gây khó chịu hoặc các tác vụ cần thiết cho việc xây dựng giao diện người dùng.
- Gulp tự giới thiệu mình là Hệ thống xây dựng dựa trên luồng.
- Phương thức nhận kết quả dưới dạng sự kiện ở giữa yêu cầu thay vì nhận kết quả ngay lập tức được gọi là luồng. Vì Gulp dựa trên luồng nên nó tận dụng tốt thế mạnh của node.js và tốc độ hoạt động thực tế tương đối nhanh hơn.



- Các nhiệm vụ của gulp được nối với nhau bằng các đường ống. Tệp được làm việc trên các luồng dọc theo đường ống (stream) và nhiều tác vụ được thực hiện đồng thời song song. Cái tên Gulp và biểu tượng hình chiếc cốc có ống hút bên trong phản ánh đặc điểm của gulp.

Cài đặt PM2

| Tổng quan PM2

- PM2 là một công cụ để bắt đầu, dừng và khởi động lại các ứng dụng Node.js.
- Nếu npm là trình quản lý gói giúp cải thiện khả năng sử dụng trong Node.js, thì đó là trình quản lý quy trình giúp cải thiện khả năng cho các nhà phát triển đang sử dụng Node.js trong Back-End bằng cách sử dụng các khung như Express.
- Giải quyết các vấn đề khi sử dụng **Node.js** làm **máy chủ** trong **Back-End**
- Điều gì sẽ xảy ra nếu máy chủ đột ngột dừng trong khi dịch vụ đang được cung cấp?
- Node.js là single-threaded vậy bạn muốn dùng multi-core hay hyper-threading thì sao?
- Đối với các nhà phát triển **Back-End**, những vấn đề này **có thể được giải quyết** bằng cách sử dụng **PM2**.



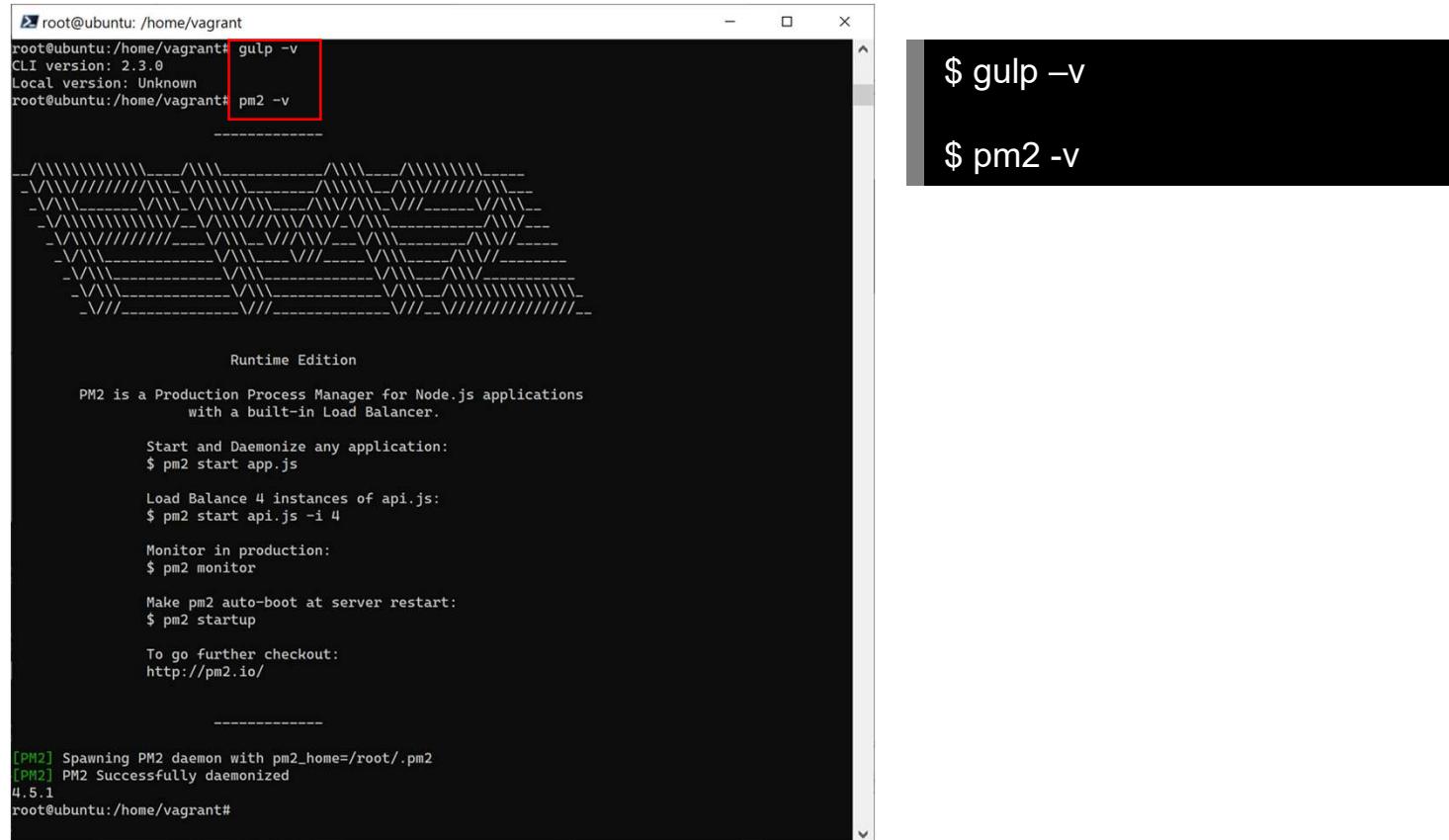
Cài đặt Gulp, PM2

```
$ sudo npm install --global gulp-cli pm2
```

```
root@ubuntu:/home/vagrant
root@ubuntu:/home/vagrant# sudo npm install --global gulp-cli pm2
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
npm WARN deprecated debug@4.1.1: Debug versions >=3.2.0 <3.2.7 || >=4 <4.3.1 have a low-severity ReDos regression when used in a Node.js environment. It
is recommended you upgrade to 3.2.7 or 4.3.1. (https://github.com/visionmedia/debug/issues/797)
npm WARN deprecated axios@0.19.2: Critical security vulnerability fixed in v0.21.1. For more information, see https://github.com/axios/axios/pull/3410
npm WARN deprecated fsevents@2.1.3: Please update to v 2.2.x
/usr/local/bin/gulp -> /usr/local/lib/node_modules/gulp-cli/bin/gulp.js
/usr/local/bin/pm2 -> /usr/local/lib/node_modules/pm2/bin/pm2
/usr/local/bin/pm2-dev -> /usr/local/lib/node_modules/pm2/bin/pm2-dev
/usr/local/bin/pm2-docker -> /usr/local/lib/node_modules/pm2/bin/pm2-docker
/usr/local/bin/pm2-runtime -> /usr/local/lib/node_modules/pm2/bin/pm2-runtime
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.1.2 (node_modules/pm2/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.3: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":
": "x64"})
+ gulp-cli@2.3.0
+ pm2@4.5.1
added 444 packages from 292 contributors in 47.962s
root@ubuntu:/home/vagrant#
```

Cài đặt Gulp, PM2

| Kiểm tra phiên bản đã cài đặt



The terminal window shows the following output:

```
root@ubuntu:/home/vagrant$ gulp -v
CLI version: 2.3.0
Local version: Unknown
root@ubuntu:/home/vagrant$ pm2 -v
$ gulp -v
$ pm2 -v
```

Runtime Edition

PM2 is a Production Process Manager for Node.js applications with a built-in Load Balancer.

Start and Daemonize any application:
\$ pm2 start app.js

Load Balance 4 instances of api.js:
\$ pm2 start api.js -i 4

Monitor in production:
\$ pm2 monitor

Make pm2 auto-boot at server restart:
\$ pm2 startup

To go further checkout:
<http://pm2.io/>

[PM2] Spawning PM2 daemon with pm2_home=/root/.pm2
[PM2] PM2 Successfully daemonized
4.5.1
root@ubuntu:/home/vagrant#

Cài đặt Nginx

Tổng quan và cài đặt Nginx

- Nginx là một phần mềm máy chủ web cung cấp các tính năng máy chủ web, proxy đảo ngược và proxy thư.
- Trong nền tảng này, mỗi dịch vụ chạy một quy trình và Nginx phải được cài đặt và định cấu hình vì các cổng được chia thành các miền phụ.



1. Thêm bộ lưu trữ APT

```
$ sudo su  
$ sudo vi /etc/apt/sources.list  
...  
# Nginx 추가  
deb http://nginx.org/packages/mainline/ubuntu/ xenial nginx  
deb-src http://nginx.org/packages/mainline/ubuntu/ xenial nginx
```

2. Tải xuống / Đăng ký Khóa bảo mật lưu trữ APT

```
$ cd /root  
$ wget http://nginx.org/keys/nginx_signing.key  
$ apt-key add nginx_signing.key  
$ rm nginx_signing.key
```

Cài đặt Nginx

| Tổng quan và cài đặt Nginx

3. Cập nhật thông tin gói nguồn APT

```
$ apt-get update
```

4. Cài đặt Nginx

```
$ apt-get install nginx
```

5. Khởi động Nginx

```
$ Dịch vụ nginx restart  
$ nginx -v
```

Cài đặt Nginx

Cài đặt nginx.conf

- cài đặt </etc/nginx/nginx.conf>
- vị trí phải được đặt bên trong máy chủ

```
http {  
    include      /etc/nginx/mime.types;  
    default_type application/octet-stream;  
  
    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '  
                  '$status $body_bytes_sent "$http_referer" '  
                  '"$http_user_agent" "$http_x_forwarded_for"';  
  
    access_log  /var/log/nginx/access.log  main;  
  
    server {  
        location ^~ /webportal {  
            proxy_pass http://localhost:8881;  
        }  
    }  
  
    sendfile      on;  
    #tcp_nopush   on;  
    keepalive_timeout  65;  
    #gzip  on;  
    include /etc/nginx/conf.d/*.conf;  
}
```

Cài đặt Nginx

Cài đặt nginx.conf

- khối vị trí

Chỉ thị **vị trí** cài đặt cách phản hồi các yêu cầu về tài nguyên trên máy chủ. Chỉ thị **server_name** cài đặt cách nginx sẽ xử lý các yêu cầu cho miền đó, chỉ thị **vị trí** xử lý các yêu cầu cho các tệp cụ thể và thư mục cụ thể.

Ex thiết lập xử lý cho các yêu cầu ví dụ như <http://example.com/blog/>
Ví dụ sau ở bên phải.

```
# /etc/nginx/sites-available/example.com

server {
    location / {
        ...
    }
    location /images/ {
        ...
    }
    location /blog/ {
        ...
    }
    location /planet/ {
        ...
    }
    location /planet/blog/ {
        ...
    }
}
```

Cài đặt Nginx

Cài đặt nginx.conf

- Sự khác biệt giữa bí danh và gốc
- Bí danh được sử dụng để thay đổi đường dẫn tệp được cung cấp bởi một URL cụ thể. Vai trò của bí danh khác với vai trò của root.

```
location /images/something/ {  
    alias /var/www/something/;  
}
```

- Ở trạng thái này, đường dẫn tệp thực tế của `http://example.com/images/something/somepath/myfile.png` trở thành `/var/www/something/sompath/myfile.png`
- `/image/thứ` được mô tả ở vị trí ở giữa sẽ bị xóa khi tạo đường dẫn.
- Mặt khác, nếu nó được cấu hình là root `/var/www/something`, thì đường dẫn tệp thực tế sẽ trở thành `/var/www/something/images/something/somepath/myfile.png`

Cài đặt Nginx

I Cài đặt nginx.conf

- Nếu bạn đặt **dấu mũ và dấu ngã (^~)** sau lệnh vị trí, nginx sẽ sử dụng lệnh phù hợp nếu nó tìm thấy kết quả khớp với chuỗi cụ thể đó, thay vì tìm kiếm kết quả khác có thể khớp. Ngay cả khi có nhiều chỉ thị đặc biệt phù hợp hơn ở nơi khác, nếu yêu cầu khớp với một trong những chỉ thị này, thì chính chỉ thị đó sẽ được sử dụng.

```
# /etc/nginx/sites-available/example.com

server {
    location ^~ /images/IndexPage/ {
        ...
    }

    location ^~ /blog/BlogPlanet/ {
        ...
    }
}
```

Bài 3.

OpenHAB (Open Home Automation Bus)

- | 3.1. Tổng quan về OpenHAB
- | 3.2. Cài đặt và chạy thử OpenHAB
- | 3.3. Định cấu hình OpenHAB để cài đặt MQTT Binding

OpenHAB là gì?

- █ OpenHAB là phần mềm mã nguồn mở miễn phí có thể giao tiếp với nhiều hệ thống tự động hóa mã nguồn mở và đóng cho nhà thông minh.
- █ Phần mềm chạy trang web trên Pi cũng như trên giao diện Android và iOS, cho phép xem trạng thái của tất cả cảm biến và có một công cụ quy tắc để thiết lập tự động hóa.
- █ Máy chủ tự động hóa mã nguồn mở cho nhà thông minh được Kai Kreuzer phát triển từ năm 2010.
- █ Cổng Home Gateway được thiết kế nhằm kết nối và phục vụ nhiều sản phẩm điện tử khác nhau tại nhà mà không bị giới hạn do kết nối vật lý (như nối tiếp, ethernet và WiFi) hoặc phần mềm (giao thức phần mềm như XMPP, MQTT và REST API).
- █ Phần mềm được triển khai bằng Java trên khuôn khổ sáng kiến Cổng dịch vụ mở (OSGi) Equinox (Eclipse PDE) và được nhiều nhà phát triển thực hiện.



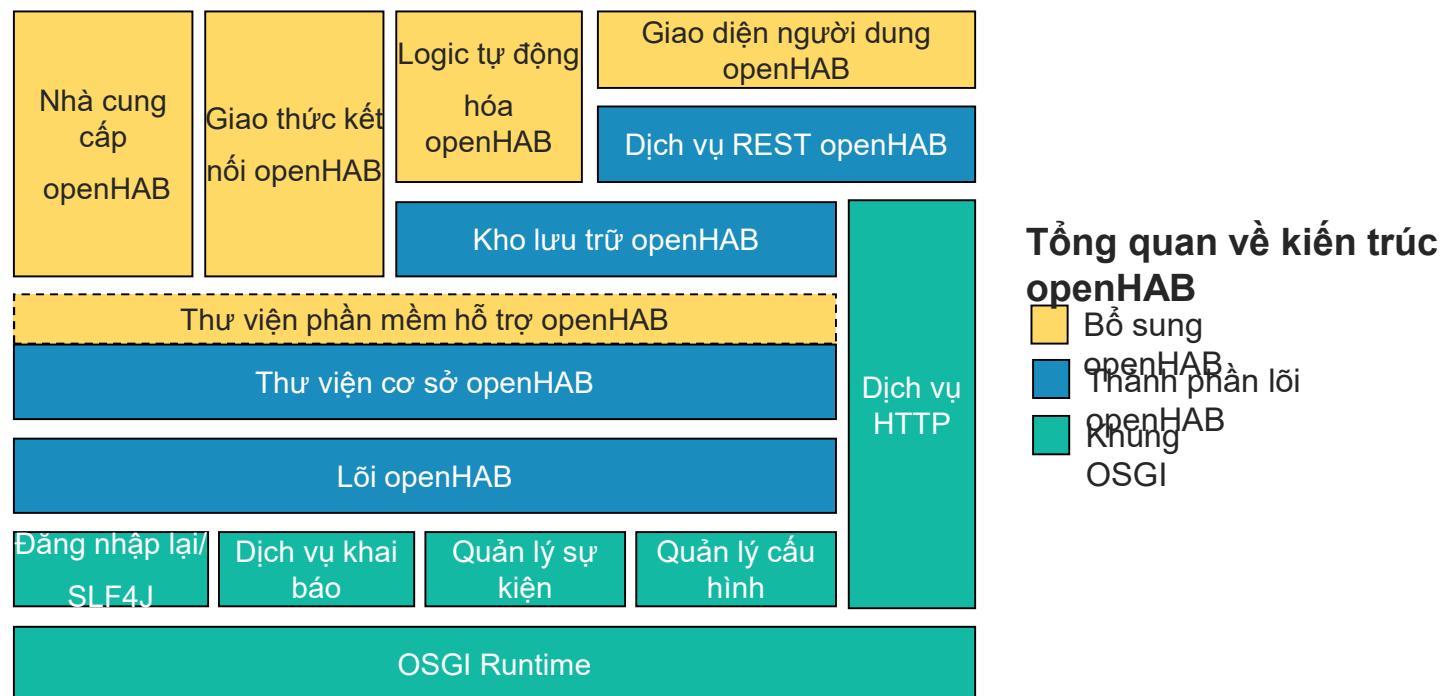
| Dự án openHAB đã thu hút sự quan tâm của cộng đồng các nhà phát triển lớn nghiên cứu chủ đề nhà thông minh từ góc độ người dùng: Các tính năng như khả năng hoạt động ngoại tuyến, bảo mật dữ liệu và khả năng tùy chỉnh được ưu tiên hàng đầu đối với một giải pháp nhà thông minh.

- **Ưu điểm:** Dự án openHAB đã đặt ưu tiên hàng đầu cho các tính năng như khả năng hoạt động ngoại tuyến, bảo mật dữ liệu và khả năng tùy chỉnh đối với một giải pháp nhà thông minh.
- **Nhược điểm:** thiết lập ban đầu phức tạp với đường cong học tập có dạng dốc đứng. Yêu cầu năng lực kỹ thuật ở một trình nhất định để thiết lập thành công.
- **Kết luận:** Giải pháp cho nhà thông minh này có những lợi ích rõ ràng so với các giải pháp hiện tại về độ tin cậy, độ trễ (thời gian để tín hiệu tiếp cận và bật/tắt thiết bị) và bảo mật dữ liệu.



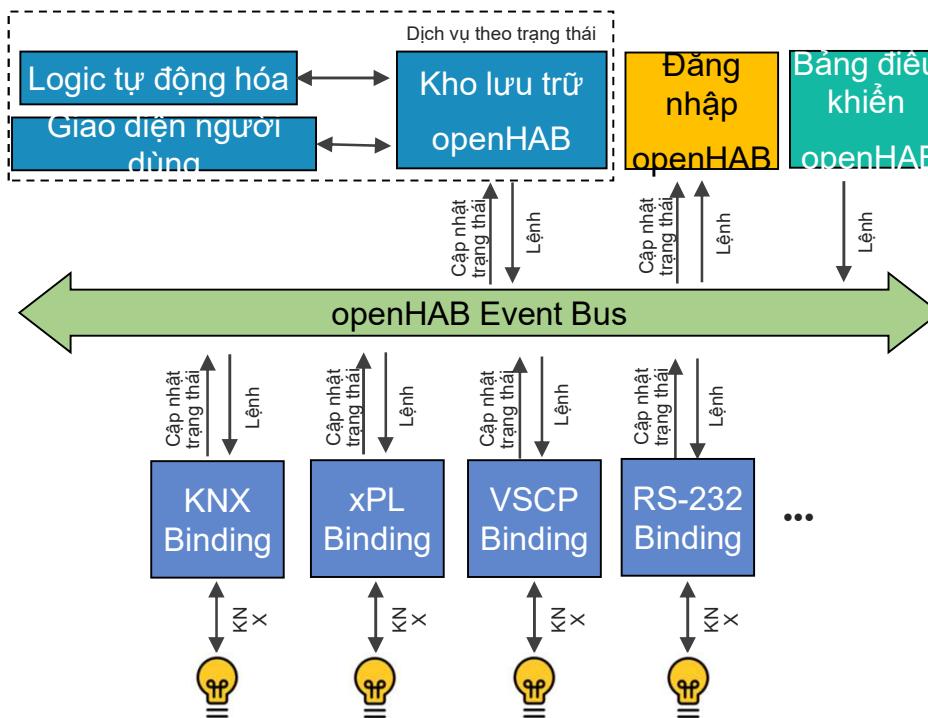
Cấu trúc

- | Phần mềm openHAB bao gồm một máy chủ openHAB runtime và một công cụ thiết kế
- | openHAB runtime is a kind of server program that operates bundle components on OSGi(Open Dịch vụ Gateway Initiative) framework, which is a Java-based automated Dịch vụ platform standard
- | Chương trình này có thể bổ sung hoặc loại bỏ hàm khi đang chạy



Cấu trúc

- Dưới đây là một số chức năng cơ bản của chương trình máy chủ. Trước tiên, máy chủ có một kênh giao tiếp kiểu Pub-Sub gọn nhẹ trong event bus.
- Event bus sử dụng dịch vụ OSGi - EventAdmin, kích hoạt các mô-đun bên trong, bao gồm giao thức nội bộ đồng bộ hóa các phần của nhiều thiết bị khác nhau, giao tiếp với nhau.



Cấu trúc

- | Hầu hết đối tượng kết nối với openHAB được gọi là thiết bị. Kho thiết bị lưu trữ thông tin về thiết bị. Ví dụ, đèn phải có khả năng lưu trữ thông tin trạng thái bật/tắt đồng thời hiển thị thông tin trạng thái khi người dùng điều khiển hoặc yêu cầu.
- | Ngoài ra, công cụ thực thi theo logic tự động hóa của openHAB cho phép thực hiện chức năng tự động hóa dựa trên trạng thái hiện tại của thiết bị theo các quy tắc và tập lệnh có sẵn.
- | Chương trình công cụ thiết kế cho phép chỉnh sửa tệp cấu hình bao gồm quy tắc tự động hóa.
- | openHAB cung cấp cấu hình văn bản hiển hình cho giao diện người dùng gọi là Sơ đồ trang. Trên Sơ đồ trang, các công cụ tiện ích xác định từng giao diện người dùng và nội dung được tổng hợp theo cấu trúc dạng cây đồng thời công cụ tiện ích hiển thị thông tin trạng thái liên quan đến các thiết bị.

ví dụ về ứng dụng phần mềm



Sprinklers, pumps, lights, cistern levels etc.



AVM Home Automation Binding



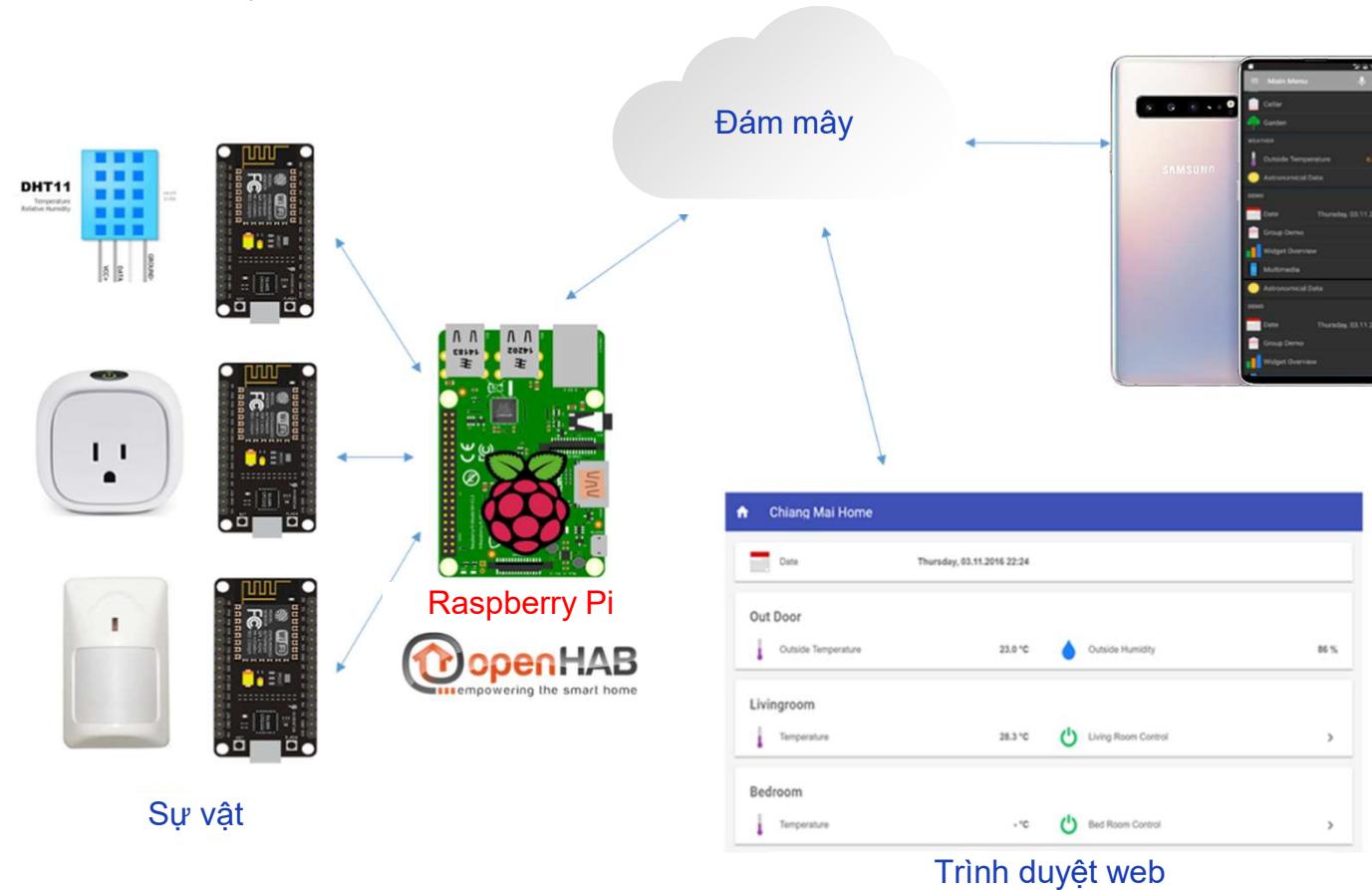
Raspberry Pi

Giao diện người dùng



Thành phần

| Nhà thông minh Raspberry Pi với OpenHAB2



Bài 3.

OpenHAB (Open Home Automation Bus)

- | 3.1. Tổng quan về OpenHAB
- | 3.2. Cài đặt và chạy thử **OpenHAB**
- | 3.3. Định cấu hình OpenHAB để cài đặt MQTT Binding

Cài đặt OpenHAB và chạy thử (1/4)

- | Đăng ký bộ nhớ để tải xuống openHAB và Phần mềm hỗ trợ
- | Mở tệp openhab.list

```
$ sudo vi /etc/apt/sources.list.d/openhab.list
```

- | Ghi nội dung dưới đây vào tệp openhab.list

```
deb http://repository-openhab.forge.cloudbees.com/release/1.6.2/apt-repo/
```

- | Cài đặt OpenHAB và phần mềm hỗ trợ

```
$ sudo apt-get update  
$ sudo apt-get install openhab-runtime openhab-addon-binding-mqtt openhab-addon-action-mail openhab-  
addon-binding-bluetooth openhab-addon-binding-serial openhab-addon-binding-weather openhab-addon-  
persistence-rrd4j
```

Cài đặt OpenHAB và chạy thử (2/4)

Tải xuống cấu hình OpenHAB bản thử nghiệm

- › Lệnh tải xuống cấu hình của openHAB bản thử nghiệm

```
$ cd ~  
$ mkdir openHAB_Demo  
$ cd openHAB_Demo  
$ wget https://github.com/openhab/openhab/releases/download/v1.6.2/distribution-1.6.2-demo-configuration.zip
```

Áp dụng cấu hình Bản thử nghiệm

- › Giải nén Distribution-1.6.2-demo-configuration.zip
- › Sao chép thư mục phần mềm hỗ trợ sang thư mục /usr/share/openhab
- › Sao chép thư mục cấu hình sang thư mục /etc/openhab
- › Sao chép tệp openhab_default.cfg sang openhab.cfg

```
$ unzip distribution-1.6.2-demo-configuration.zip  
$ sudo cp -rf addons/ /usr/share/openhab/  
$ sudo cp -rf configurations/ /etc/openhab/  
$ sudo cp /etc/openhab/configurations/openhab_default.cfg /etc/openhab/configurations/openhab.cfg
```

Cài đặt OpenHAB và chạy thử (3/4)

Chạy OpenHAB

```
$ sudo /etc/init.d/openhab start
```

Truy cập sơ đồ trang web của Bản thử nghiệm

- Kiểm tra địa chỉ IP Raspberry Pi

```
pi@raspberrypi: ~/openHAB_Demo
pi@raspberrypi: ~/openHAB_Demo $ ifconfig
eth0      Link encap:Ethernet HWaddr b8:27:eb:ee:bc:e8
          inet addr:192.168.1.229 Bcast:192.168.1.255 Mask:255.255.255.0
                  inet6 addr: fe80::ba27:ebff:feee:bce8/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                      RX packets:36160 errors:0 dropped:0 overruns:0 frame:0
                      TX packets:9318 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:50329683 (47.9 MiB) TX bytes:896165 (875.1 KiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING MTU:65536 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

pi@raspberrypi: ~/openHAB_Demo $
```

Cài đặt OpenHAB và chạy thử (4/4)

Kiểm tra màn hình Bản thử nghiệm

- Địa chỉ Raspberry Pi IP :8080/openhab.app?sitemap=demo
- 192.168.1.229:8080/openhab.app?sitemap=demo



- Khi thoát chương trình, bạn có thể thực thi lệnh exit trong osgi shell.

```
osgi> exit  
Bạn thực sự muốn dừng Equinox? (y/n: defualt=y) y
```

Bài 3.

OpenHAB (Open Home Automation Bus)

- | 3.1. Tổng quan về OpenHAB
- | 3.2. Cài đặt và chạy thử HAB
- | 3.3. Định cấu hình OpenHAB để cài đặt MQTT Binding

Định cấu hình OpenHAB để cài đặt MQTT Binding (1/6)

I Thủ nghiệm GPIO Binding và REST API (1)

- Tệp cấu hình cơ bản của openHAB là openhab.cfg. Tệp lưu trữ thông tin cấu hình cơ bản như địa chỉ IP, máy chủ thư điện tử, vị trí thư mục, v.v. Sau khi giải nén tệp, hãy sao chép tệp openhab_default.cfg sang tệp openhab.cfg.
- Đặt các tệp cấu hình riêng lẻ về phần còn lại của mục, sơ đồ trang web, quy tắc, tập lệnh, độ bền, khả năng chuyển đổi dưới dạng thư mục con trong <openhab home>/configurations.
- Một mục đại diện cho một đối tượng được kết nối. Có thể gán mục bằng các yếu tố như màu sắc, tình trạng tiếp xúc, ngày giờ, độ sáng, nhóm, số, cửa cuốn, chuỗi, công tắc, v.v.

Mục	Giải thích	Kiểu câu lệnh
Màu sắc	Thông tin màu sắc (RGB)	OnOff, IncreaseDecrease, Percent, HSB
Tiếp xúc	Mục lưu trữ trạng thái cửa ra vào/cửa sổ	OpenClosed
Ngày giờ	Ngày giờ lưu trữ (Tham khảo NTP Binding)	
Công tắc điều chỉnh độ sáng đèn	Mục chứa giá trị phần trăm của Công tắc điều chỉnh độ sáng đèn	OnOff, IncreaseDecrease, Percent
Group	Nhóm các mục thành một nhóm	-
Số	Lưu trữ các giá trị dưới dạng chữ số	Thập phân
Cửa cuốn	Kiểu mù	UpDown, StopMove, Percent
Chuỗi	Lưu trữ văn bản	Chuỗi
Công tắc	Thường dùng ở đèn (bật/tắt)	OnOff

Định cấu hình OpenHAB để cài đặt MQTT Binding (2/6)

| Thủ nghiệm GPIO Binding và REST API (2)

- Ngôn ngữ có định dạng đơn giản định nghĩa mục đã tạo bằng cách sử dụng khuôn khổ Xtext eclipse, công cụ phát triển Ngôn ngữ miền chuyên biệt (DSL). Nên đặt các tệp định nghĩa mục trong thư mục configuration/items. Các mục phải được xác định theo định dạng sau:

Kiểu mục Tên mục [“Nhãn”] [<Biểu tượng>] [(Nhóm1, Nhóm2, ...)] [{Cấu hình Binding}]

Định cấu hình OpenHAB để cài đặt MQTT Binding (3/6)

Tạo tệp openhab.cfg

```
$ sudo cp /etc/openhab/configurations/openhab_default.cfg  
/etc/openhab/configurations/openhab.cfg
```

Định nghĩa tệp openhab.cfg cho MQTT Broker

```
$ sudo vi /etc/openhab/configurations/openhab.cfg
```

Thêm cụm bên dưới vào dòng 423 và 427 (Ghi địa chỉ IP của Raspberry Pi)

```
mqtt:broker.url=tcp://192.168.1.229:1883  
mqtt:broker.clientId=openhab
```

※ Đăng ký máy chủ openHAB trên Mosquitto Broker

Định cấu hình OpenHAB để cài đặt MQTT Binding (4/6)

I Sơ đồ trang web

- Chủ thể tạo ra các phần tử giao diện người dùng để hiển thị các mục.
- Các yếu tố dưới đây có thể được bao gồm trong sơ đồ trang web.

Phần tử	Giải thích
Colorpicker	Một tiện ích có thể chọn màu từ bảng màu
Chart	Thêm một đối tượng biểu đồ xuất dữ liệu nhật ký
Frame	Một khu vực bao gồm các thành phần hoặc khung sơ đồ trang web khác
Group	Xuất tất cả các phần tử của nhóm đã cho được xác định trong tệp lưu trữ vật phẩm
Image	Hình ảnh cho ra
List	Danh sách cho ra
Selection	Truy cập trang mới được lựa chọn trong số các giá trị được xác định làm tham số
Setpoint	Có thể được thay đổi theo các giá trị hiển thị. Cấu hình sân khấu hoặc tối đa/tối thiểu là có khả năng
Slider	Slide đầu ra
Switch	Chuyển mục
Text	yếu tố văn bản
Video	Hiển thị video
Webview	Hiển thị trang web

Định cấu hình OpenHAB để cài đặt MQTT Binding (5/6)

| Thêm sơ đồ trang web vào openHAB

```
$ sudo vi /etc/openhab/configurations/sitemaps/kaizen.sitemap
```

```
sitemap kaizen label="Main Menu"
{
Frame label="MQTT"{
Switch item=mqttsw1 label="MQTT Switch 1"
Switch item=mqttsw2 label="MQTT Switch 2"
Switch item=mqttsw3 label="MQTT Switch 3"
Text item=Office_Temp
Text item=Office_Humidity
}
}
```

Định cấu hình OpenHAB để cài đặt MQTT Binding (6/6)

| Thêm mục vào openHAB

```
$ sudo vi /etc/openhab/configurations/items/kaizen.items
```

```
Group All
Switch mqttsw1 "Switch 1" (all)
{mqtt=">[broker:/wiznet/sw1:command:on:1],>[broker:/wiznet/sw1:command:off:0]"}
Switch mqttsw2 "Switch 2" (all)
{mqtt=">[broker:/wiznet/sw2:command:on:1],>[broker:/wiznet/sw2:command:off:0]"}
Switch mqttsw3 "Switch 3" (all)
{mqtt=">[broker:/wiznet/sw3:command:on:1],>[broker:/wiznet/sw3:command:off:0]"}
Number Office_Temp "Temperature [%.1f °C]" <temperature>(all)
{mqtt="<[broker:/wiznet/temperature:state:default]"}
Number Office_Humidity "Humidity [%.1f °C]" <humidity>(all)
{mqtt="<[broker:/wiznet/humidity:state:default]"}  

```

Bài 4.

Trực quan hóa dữ liệu

- | 4.1. Khái niệm trực quan hóa và tầm quan trọng
- | 4.2. Công cụ trực quan hóa mã nguồn mở: Grafana

Tổng quan về trực quan hóa dữ liệu

- | Trực quan hóa dữ liệu đề cập đến quá trình thể hiện và cung cấp kết quả phân tích dữ liệu một cách trực quan để dễ hiểu. Chính là truyền tải hiệu quả thông tin được hiển thị trong phân tích.
- | Để hiểu về trực quan hóa dữ liệu, trước tiên bạn cần hiểu về trực quan hóa thông tin.
 - Trực quan hóa thông tin thường có nghĩa là thể hiện trực quan dữ liệu quy mô lớn và phi số lượng bằng cách sử dụng màu sắc, số liệu thống kê (biểu đồ, đồ thị, v.v.), hình ảnh, v.v.
 - Phương pháp trực quan hóa thông tin: trực quan hóa thời gian, trực quan hóa phân phôi, trực quan hóa mối quan hệ, trực quan hóa so sánh, trực quan hóa không gian
 - Hiệu ứng trực quan hóa thông tin:

- ① Có thể khiến người xem quan tâm
- ② Có thể giúp đưa ra phán đoán tình huống nhanh chóng bằng cách rút ngắn thời gian tiếp nhận thông tin
- ③ Có khả năng lan truyền thông tin nhanh chóng
- ④ Có thể nhớ dữ liệu trong thời gian dài hơn ngay cả khi nhìn thấy nó một lần
- ⑤ Có khả năng giao tiếp hiệu quả về thông tin

Tổng quan về trực quan hóa dữ liệu

| Phương pháp trực quan hóa cốt quan trọng

Danh mục	Các phương pháp trực quan hóa quan trọng
Hình dung tạm thời	Biểu đồ thanh, Biểu đồ thanh xếp chồng, Biểu đồ chấm
Trực quan hóa phân phối	Biểu đồ tròn, biểu đồ Donut, TreeMap, biểu đồ liên tục xếp chồng
Trực quan hóa mối quan hệ	Biểu đồ tán xạ, Biểu đồ bong bóng, Biểu đồ
Trực quan so sánh	Bản đồ nhiệt, Biểu đồ sao, Tọa độ song song, Chia tỷ lệ đa chiều
Trực quan hóa không gian	Ánh xạ bản đồ

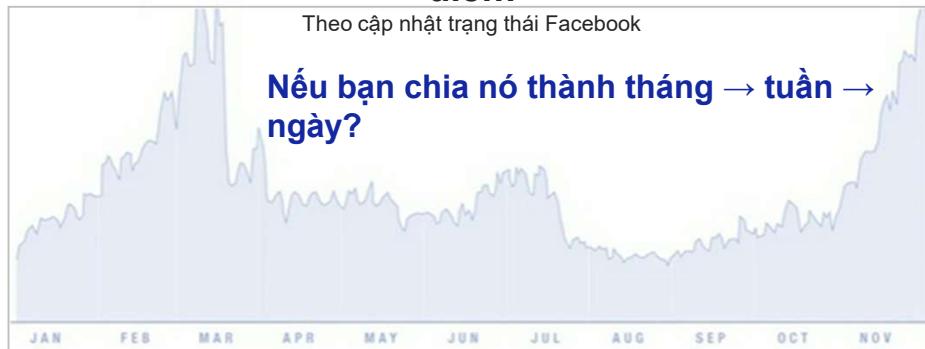
Phương pháp trực quan hóa

| hình dung tạm thời

- Dữ liệu về thời gian là rất tự nhiên đối với chúng tôi. Đặc điểm nhất của dữ liệu liên quan đến thời gian, dữ liệu chuỗi thời gian, là xu hướng thịnh hành.
- Theo thời gian, mọi người thay đổi suy nghĩ, nhân khẩu học thay đổi và doanh nghiệp mở rộng.
 - Nếu bạn đo lường và ghi lại mức độ những thay đổi này đã xảy ra, nó sẽ trở thành dữ liệu chuỗi thời gian.
 - Việc tìm ra các xu hướng thay đổi đòi hỏi cái nhìn bao quát hơn là tập trung vào dữ liệu riêng lẻ. Thay vì chỉ tập trung ý nghĩa của một giá trị trong một phần, khi ta xem xét toàn bộ ngữ cảnh, một câu chuyện có ý nghĩa hơn có thể được truyền tải.

Thời gian chia tay cao

điểm



- Có bất kỳ trường hợp ngoại lệ do các sự kiện đặc biệt?
- Có phần nào nổi bật đặc biệt không?
- Có thay đổi đột ngột nào không (tăng hay giảm đi kèm)?
- Có những mẫu nhất định?
 - Góc nhìn và phán đoán của nhà phân tích dữ liệu là cần thiết

[Hình] Khi nào người ta chia tay nhiều nhất (Nguồn: TED_David McCandless - Vẻ đẹp của trực quan hóa dữ liệu)

Phương pháp trực quan hóa

I Trực quan hóa phân phối

- Dữ liệu tỷ lệ tương tự như dữ liệu chuỗi thời gian, nhưng đơn vị phân chia của dữ liệu tỷ lệ không phải là thời gian, mà là phân loại và số nhánh.
 - Số lượng nhánh: Các lựa chọn hoặc hậu quả có thể xảy ra.
 - Đặc điểm chung: Tối đa, Tối thiểu, Phân phối tổng thể
- Nếu chúng ta cố gắng biểu thị dữ liệu phân phối calo của một bữa ăn, thì thực phẩm tiêu thụ nhiều calo nhất và thực phẩm tiêu thụ ít calo nhất tương ứng với mức tối đa và tối thiểu.
- Khi bạn kiểm tra xem lượng calo được tiêu thụ đồng đều từ các chất dinh dưỡng khác nhau như chất béo, protein và carbohydrate hay liệu lượng tiêu thụ có phụ thuộc vào bất kỳ chất dinh dưỡng nào hay không, bạn có thể xem xét sự phân bổ tổng thể.

Phương pháp trực quan hóa

| Trực quan hóa phân phối



- Sơ đồ cây thể hiện cách thức và lý do hàng tỷ đô la được chi tiêu - size of spending
- Màu sắc - động cơ chi tiêu (tím: tranh chấp, đỏ: quyên góp, xanh lục: thu nhập)

Bằng cách sử dụng sơ đồ cây, có thể hiểu thông tin phân phối thông qua kích thước của hình chữ nhật, so sánh cũng như phân tích sự phân phối giữa các phân loại phụ thông qua màu sắc.

[Figure] Worldwide Billion Dollar Spending

(Source: TED_David McCandless - The beauty of data visualization)

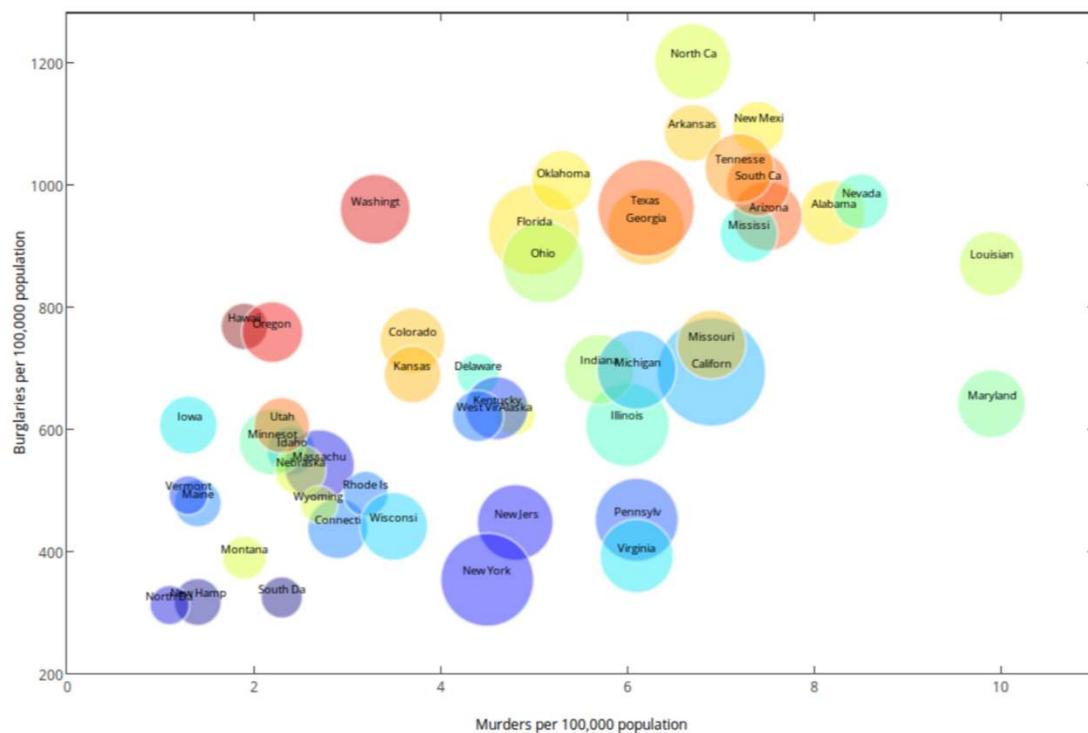
Phương pháp trực quan hóa

| Trực quan hóa mối quan hệ

- Thống kê là nghiên cứu tìm kiếm mối quan hệ giữa các dữ liệu. Có sự tương đồng nào giữa các nhóm không? Điều gì về các nhóm nhỏ trong một nhóm lớn? Trong số các mối quan hệ trong thống kê, mối quan hệ được biết đến rộng rãi nhất cho đến nay là mối tương quan.
 - Tương quan đề cập đến một mối quan hệ như chiều cao và cân nặng (nói chung, cao hơn có xu hướng nặng hơn) (đôi khi cao hơn nhưng nhẹ cân hơn).
- Con số thể hiện rõ nhất sự trực quan hóa mối quan hệ là một biểu đồ bong bóng. Ví dụ dưới đây được thể hiện bằng cách sử dụng dữ liệu về tần suất tội phạm theo tiểu bang ở Hoa Kỳ và có thể hiểu được mối quan hệ giữa bốn biến số trong khoảng thời gian cực kỳ ngắn.

Phương pháp trực quan hóa

Trực quan hóa mối quan hệ



[Figure] Murder and crime frequency by state
(Source: <https://goo.gl/FufbhD>)

- Trực quan hóa thông tin là một loại nén kiến thức bao gồm một lượng lớn kiến thức và thông tin thu được từ nhiều nguồn dữ liệu trong một không gian nhỏ.
- Giờ đây, có thể phân tích hoặc tạo hoạt ảnh có tính tương tác cho các biểu đồ bong bóng (bằng cách thay đổi chúng ngay lập tức từ một quan điểm này sang một quan điểm khác) bằng nhiều công cụ phân tích trực quan, cho phép phân tích và ra quyết định nhanh hơn.

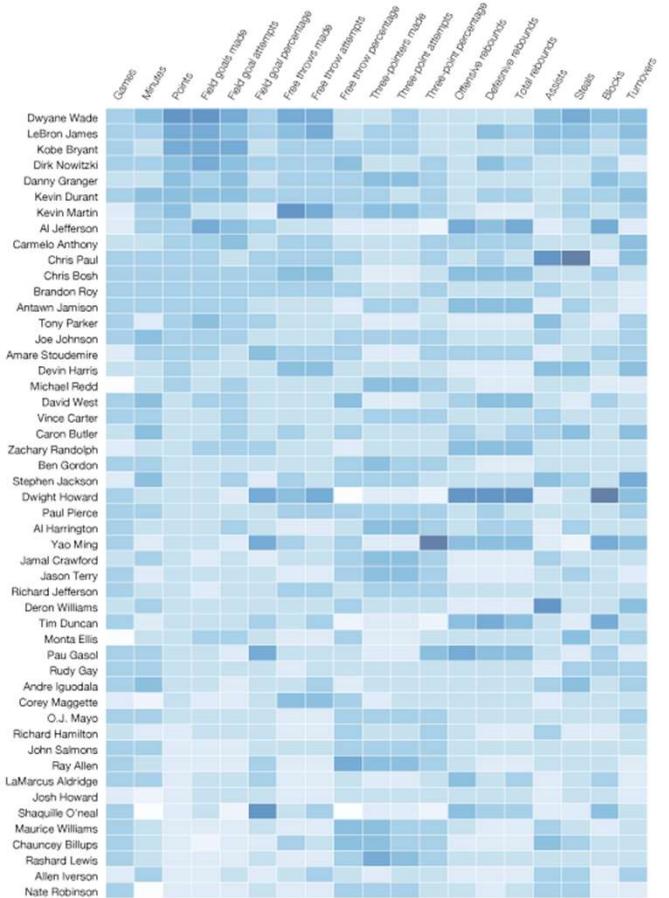
trục x - số vụ giết người trên 100.000 dân, trục y - kẻ trộm trên 100.000 dân, kích thước vòng tròn - số dân Hoa Kỳ theo tiểu bang, màu sắc - số tội theo tiểu bang

Màu đỏ càng đậm, càng nhiều tội phạm và màu xanh càng đậm thì càng ít tội phạm.

Phương pháp trực quan hóa

Trực quan hóa so sánh

- So sánh với một biến đơn lẻ thì rất dễ, nhưng nếu bạn có một trăm ngôi nhà và một trăm con mèo để phân loại thì sao? Điều gì sẽ xảy ra nếu bạn cần so sánh số phòng ngủ, diện tích sân và chi phí bảo trì vì có nhiều biến số hơn để so sánh?
 - Do đó, cần phải kiểm tra danh sách so sánh đơn vị bằng số lượng đối tượng được so sánh nhân với số lượng biến được so sánh.
- Vì có nhiều mục phân tích trong bản đồ nhiệt, nên cần xác định điểm bắt đầu trước khi phân tích.
 - Biểu đồ bên phải đặt điểm số (Điểm ở cột thứ ba), kỷ lục quan trọng nhất trong bóng rổ, làm điểm bắt đầu phân tích và sắp xếp theo thứ tự giảm dần.
 - Dựa trên điểm bắt đầu của phân tích, bạn có thể khám phá sự khác biệt hoặc tương đồng giữa các số liệu được ghi lại.
 - Ưu điểm của bản đồ nhiệt là bạn có thể xem toàn bộ dữ liệu trong nháy mắt. Ngay cả khi dữ liệu bạn có bao gồm một số biến, bạn có thể tìm thấy một phạm vi ngoại lệ bằng cách phân loại lẫn nhau hoặc chia chúng theo Đơn vị.

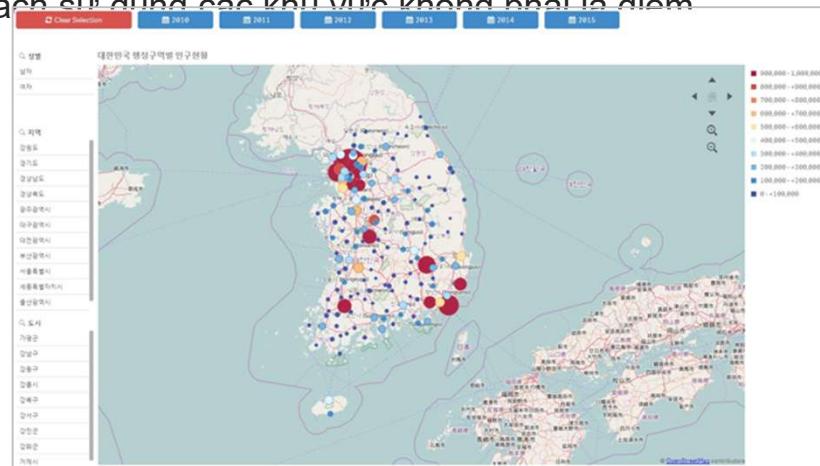


[Figure] Best NBA player record-heatmap chart as of 2008-2009 season (X-axis- Player's performance item, Y-axis- Player name, color-score, Source: <https://goo.gl/AlT7sa>)

Phương pháp trực quan hóa

| Trực quan hóa không gian

- Bản đồ là một nhánh của trực quan hóa sử dụng trực giác được lưu truyền rộng rãi. So sánh một vị trí trên bản đồ với một vị trí khác tương tự như biểu đồ phân tán trực quan hóa mối quan hệ. Tuy nhiên, không giống như biểu đồ phân tán, bản đồ sử dụng vĩ độ và kinh độ thay vì tọa độ x và y của biểu đồ để xác định tọa độ dưới dạng điểm. Mỗi quan hệ kết nối giữa điểm A và điểm B trên bản đồ có thể được đo bằng khoảng cách hoặc thời gian di chuyển.
- Nhiều bản đồ có thể thể hiện các khía cạnh khác nhau của thời gian. Thông qua hình ảnh động hoặc thay đổi sang một góc nhìn khác, bạn có thể thấy những thay đổi về giá bất động sản hoặc tốc độ tăng dân số trong một khu vực cụ thể.
- Vì các thành phố, tiểu bang, quốc gia và lục địa có ranh giới nhất định nên dữ liệu có thể được biểu thị bằng cách sử dụng các khu vực không phải là điểm

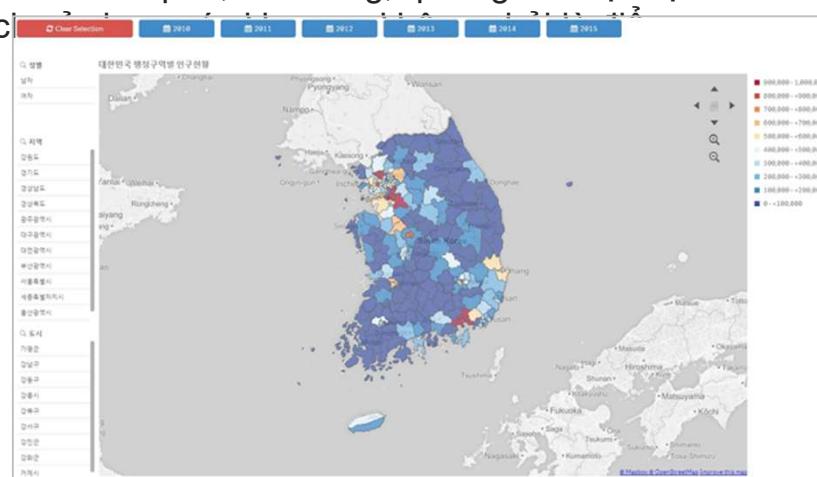


[Hình] Tình trạng dân số theo khu vực hành chính của Hàn Quốc (Biểu đồ bong bóng) (Quy mô bán hàng - Kích thước của hình tròn, Dân số - Màu của hình tròn) - Các chấm hiển thị trên bản đồ chỉ biểu thị một vị trí và các chấm được tập trung tại các khu vực có tọa độ dày đặc làm giảm khả năng đọc .

Phương pháp trực quan hóa

| Trực quan hóa không gian

- Bản đồ là một nhánh của trực quan hóa sử dụng trực giác được lưu truyền rộng rãi. So sánh một vị trí trên bản đồ với một vị trí khác tương tự như biểu đồ phân tán trực quan hóa mối quan hệ. Tuy nhiên, không giống như biểu đồ phân tán, bản đồ sử dụng vĩ độ và kinh độ thay vì tọa độ x và y của biểu đồ để xác định tọa độ dưới dạng điểm. Mỗi quan hệ kết nối giữa điểm A và điểm B trên bản đồ có thể được đo bằng khoảng cách hoặc thời gian di chuyển.
- Nhiều bản đồ có thể thể hiện các khía cạnh khác nhau của thời gian. Thông qua hình ảnh động hoặc thay đổi sang một góc nhìn khác, bạn có thể thấy những thay đổi về giá bất động sản hoặc tốc độ tăng dân số trong một khu vực cụ thể.
- Vì các thành phố, tiểu bang, quốc gia và lục địa có ranh giới nhất định nên dữ liệu có thể được biểu thị bằng các



[Hình] Tình trạng dân số theo khu vực hành chính của Hàn Quốc - Biểu đồ hiển thị trên bản đồ bằng cách chia màu theo khu vực (Biểu đồ Choropleth): Không giống như biểu đồ bong bóng, dữ liệu chỉ có thể được thể hiện bằng màu sắc nên chúng ta chỉ cần tập trung vào một chỉ số và phân tích nó .

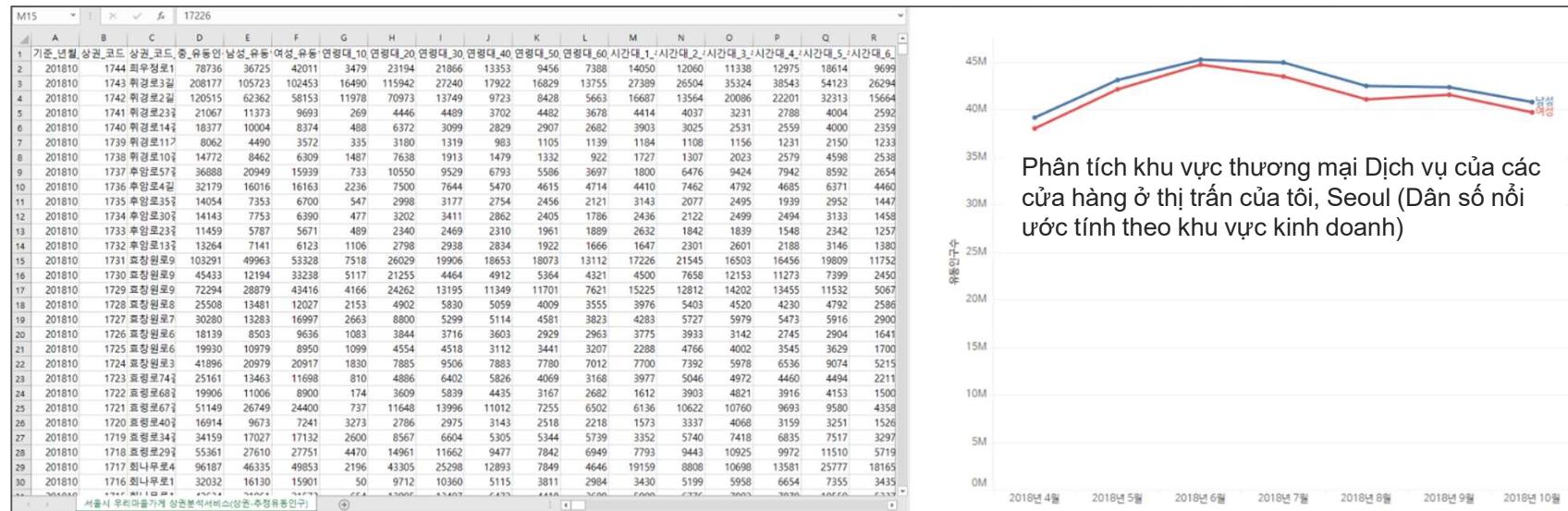
Tầm quan trọng của trực quan hóa

Một lượng lớn dữ liệu có thể được xem trong khoảng thời gian cực kỳ ngắn.

Các loại trực quan hóa dữ liệu khác nhau

- Chúng giúp bạn dễ dàng tìm và hiểu ý nghĩa của dữ liệu.
- Dựa trên kích thước, vị trí hoặc màu sắc của các số liệu được sử dụng trong biểu đồ, chúng ta có thể so sánh kích thước của dữ liệu, hiểu được sự phân bố hoặc tìm mức độ liên quan.

→ **Bạn có thể khám phá những câu chuyện có ý nghĩa và thông tin chi tiết về dữ liệu ẩn trong dữ liệu.**



Tầm quan trọng của trực quan hóa

| Bất kỳ ai cũng có thể dễ dàng tìm thấy thông tin chuyên sâu về dữ liệu, ngay cả khi không có chuyên môn về phân tích dữ liệu

Ý nghĩa của các mẫu trực quan hóa dữ liệu

- Hình dạng, kích thước, vị trí hoặc màu sắc của các yếu tố trực quan hóa. Dựa trên mẫu trực quan của biểu đồ, bạn có thể nhanh chóng tìm thấy các giá trị ngoại lệ đang có xu hướng thay đổi trong dữ liệu và phân biệt chúng với các giá trị khác.
- Bất kỳ ai cũng có thể dễ dàng tìm thấy thông tin chi tiết về dữ liệu dựa trên các mẫu trực quan mà không cần có kiến thức chuyên môn về toán học hoặc thống kê để phân tích dữ liệu.
 - Thông qua trực quan hóa, phạm vi sử dụng dữ liệu và phạm vi tạo giá trị mới được mở rộng.
- Theo cách tương tự, chúng tôi sử dụng biểu đồ trực quan hóa dữ liệu và dẫn dắt câu chuyện dựa trên các mẫu biểu đồ trực quan sẽ giúp lưu lại trong trí nhớ của mọi người trong một thời gian dài, và chúng tôi sử dụng phương pháp trực quan hóa để truyền tải một cách mạnh mẽ thông tin chỉ dựa trên cơ sở dữ liệu đến với người khác.
- Con người là loài động vật có thị giác cao và nhận thức đầu vào trực quan một cách tự nhiên nhanh hơn bất kỳ phương pháp nào khác. (Hơn 80% thông tin mà một người có được thông qua các giác quan là thông qua thị giác)
 - Chúng tôi có thể hiểu tại sao trực quan hóa dữ liệu rất hiệu quả đối với chúng tôi.

Tầm quan trọng của trực quan hóa

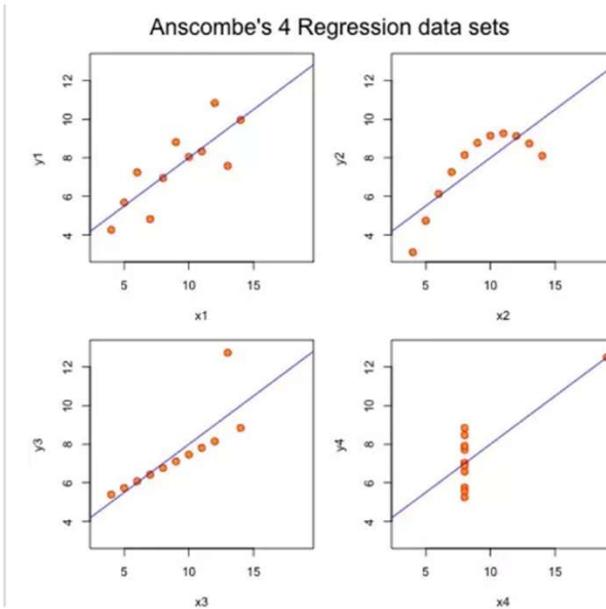
| Có thể rút ra kết quả phân tích dữ liệu chính xác hơn so với thống kê tóm tắt

- Trực quan hóa dữ liệu không chỉ được sử dụng với mục đích đơn thuần là đưa ra kết quả phân tích dữ liệu mà còn là một phương pháp thăm dò dữ liệu để **phân tích chính xác**. (Biểu đồ dữ liệu: kết hợp các biến dữ liệu, khám phá dữ liệu theo các tiêu chuẩn tính toán khác nhau cho dữ liệu số và thông tin dựa trên các mẫu biểu đồ)
- Phân tích trực quan có nghĩa là trực quan chủ yếu được sử dụng trong quá trình phân tích dữ liệu.
- Các mẫu trực quan → Hiểu dữ liệu chính xác → Khám phá dữ liệu → Rút ra những hiểu biết sâu sắc

I	II	III	IV
10	8.04	10	7.46
8	6.95	8	6.77
13	7.58	13	12.74
9	8.81	9	7.11
11	8.33	11	7.81
14	9.96	14	8.84
6	7.24	6	6.08
4	4.26	4	5.39
12	10.84	12	8.15
7	4.82	7	6.42
5	5.68	5	5.73

▼ ▼ ▼ ▼

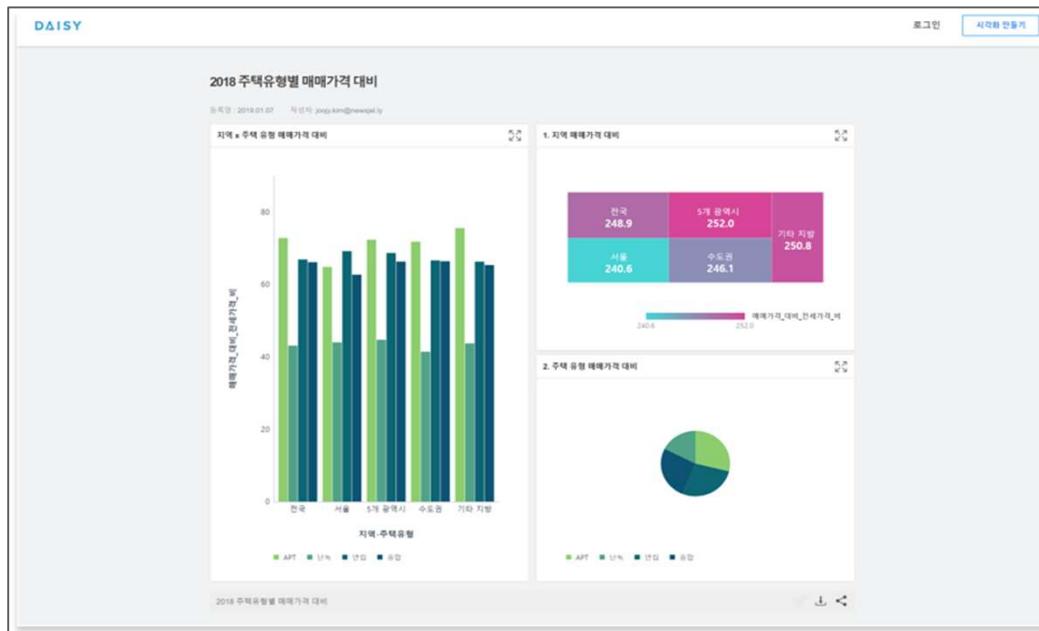
Mean of X 11.0 **Correlation between X and Y** 0.875
Variance of X 10.0 **Linear regression** $y=3.0+0.5x$
Mean of Y 7.5
Variance of Y 3.75



Tầm quan trọng của trực quan hóa

| Chia sẻ hiệu quả thông tin chuyên sâu về dữ liệu cho phép ra quyết định dựa trên dữ liệu đó

- ▶ Trực quan hóa như một cách để tận dụng dữ liệu
 1. Đầu tiên, nó được sử dụng để rút ra những hiểu biết sâu sắc trong quá trình phân tích dữ liệu, sau đó còn được dùng để chia sẻ những kiến thức về dữ liệu với nhiều người → Báo cáo, thuyết trình
 2. Sử dụng bảng điều khiển dữ liệu



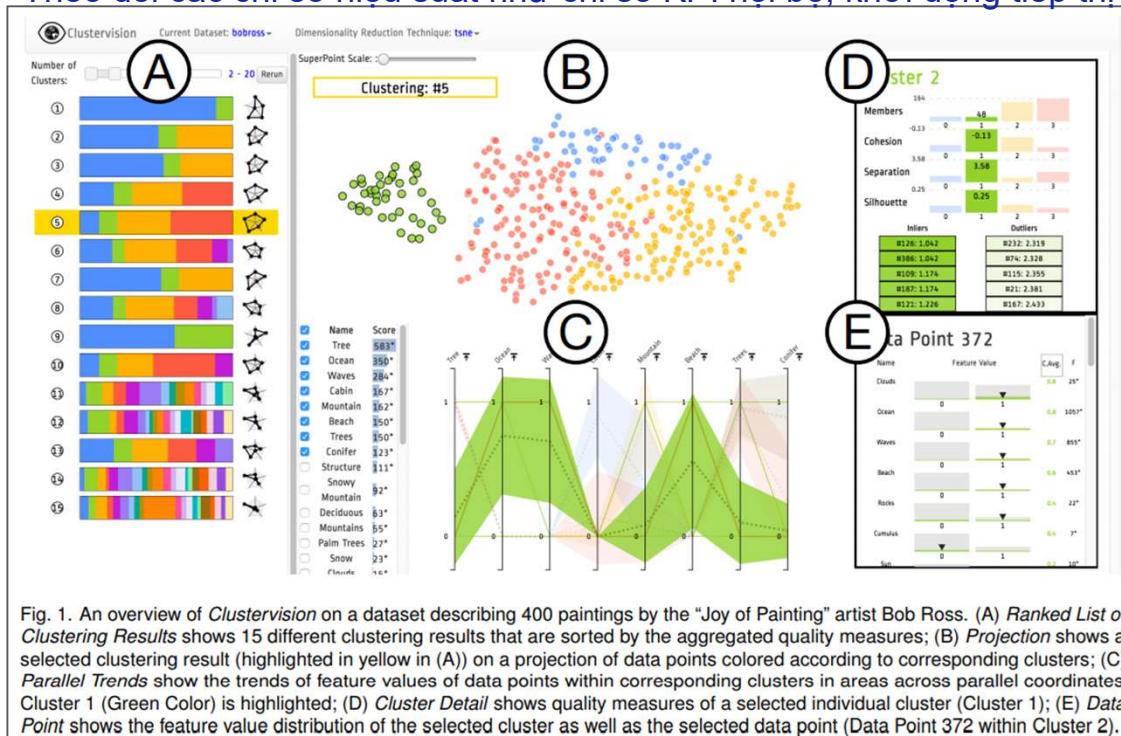
- ▶ Bảng điều khiển dữ liệu là một bảng bao gồm một số biểu đồ và bảng trực quan. Nó được sử dụng để theo dõi các chỉ số dữ liệu quan trọng (KPI).

Tầm quan trọng của trực quan hóa

| Có vô số lĩnh vực và cách để sử dụng trực quan hóa dữ liệu

- Ở cấp độ công nghiệp và doanh nghiệp, trực quan hóa dữ liệu đang được giới thiệu để tăng cường sử dụng dữ liệu.

→ Theo dõi các chỉ số hiệu suất như chỉ số KPI nội bộ, khởi động tiếp thị dựa trên dữ liệu, v.v.

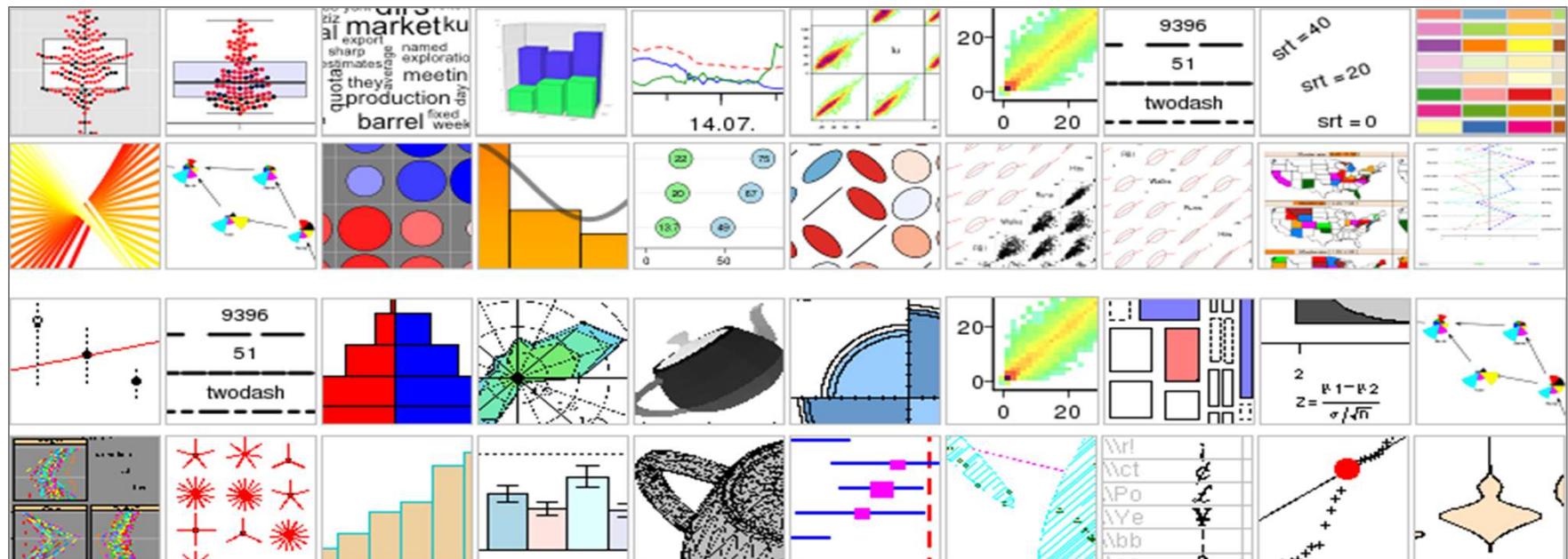


- Ưu điểm lớn nhất của trực quan hóa dữ liệu là bất kỳ ai cũng có thể sử dụng dữ liệu và rút ra những hiểu biết sâu sắc mà không gặp nhiều khó khăn.

Công cụ trực quan hóa mã nguồn mở ⋯ R

I Các khía cạnh khác nhau của ngôn ngữ R

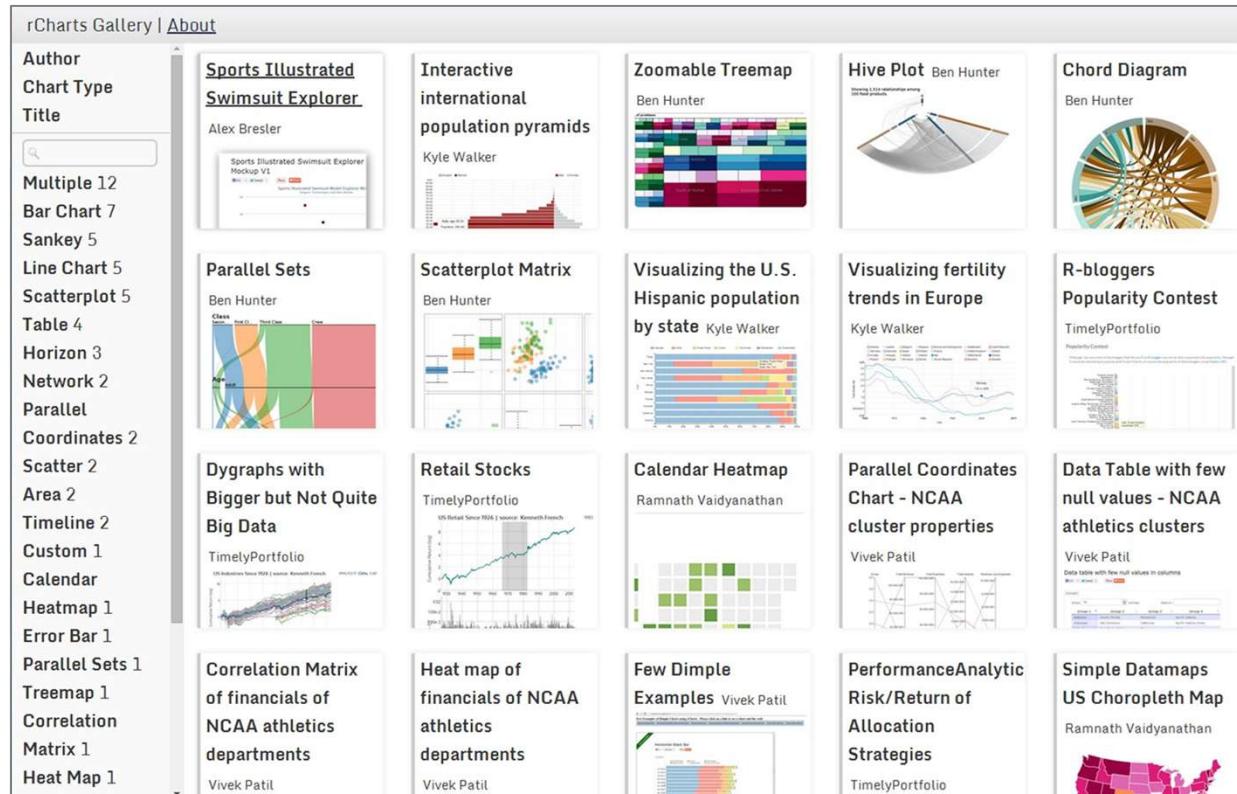
- R như một công cụ phân tích toán học/thống kê
- R như một ngôn ngữ lập trình
- R như một công cụ trực quan



Công cụ trực quan hóa mã nguồn mở ⋯ R

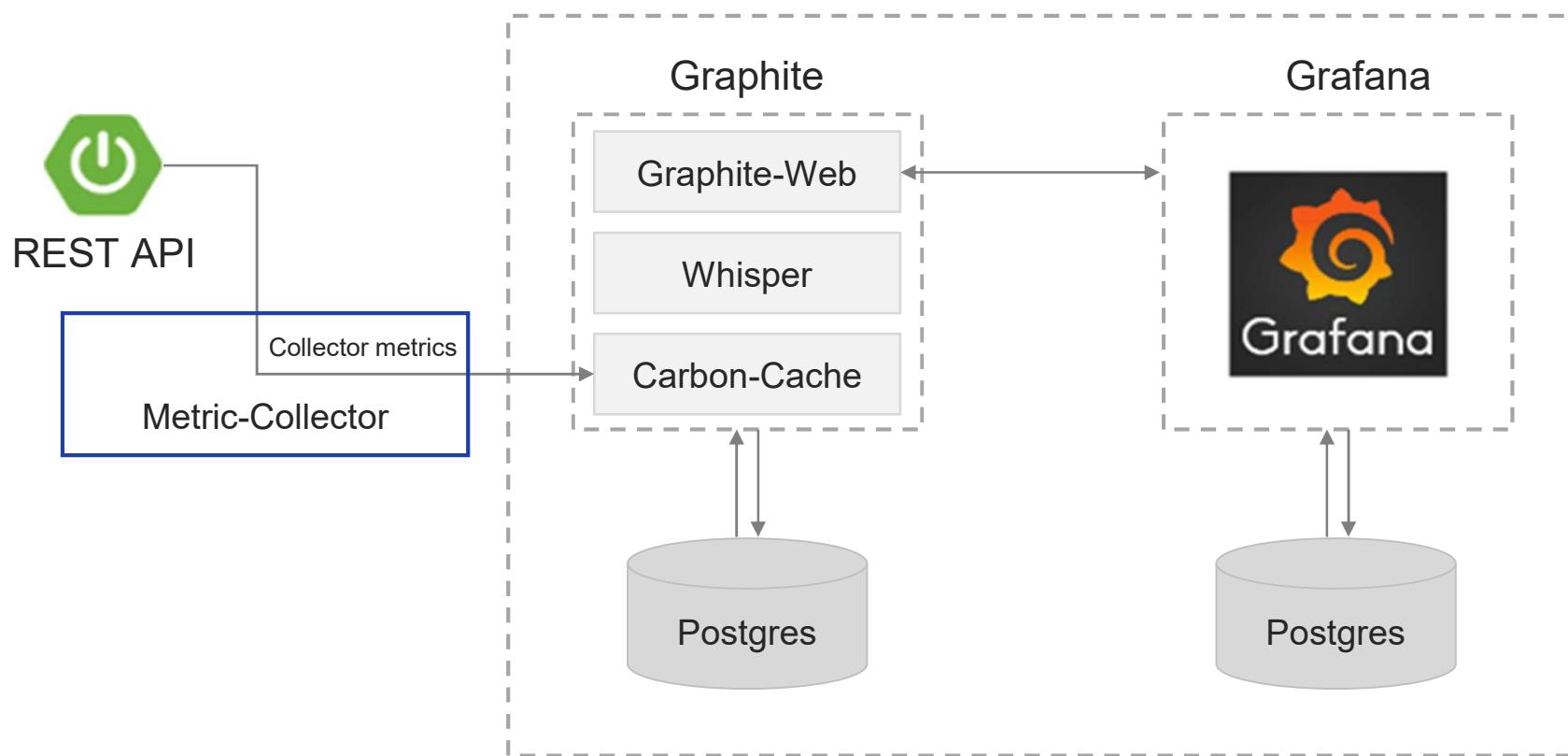
Ngôn ngữ R và tính năng hiển thị

- Kết hợp phân tích và trực quan hóa



Công cụ trực quan hóa mã nguồn mở ... Graphite

| Kiến trúc



Công cụ trực quan hóa mã nguồn mở ... Graphite

- | Graphite về cơ bản là một kho lưu trữ dữ liệu đặc biệt liên tục tích lũy thời gian và dữ liệu trong một không gian tên cụ thể. Nó cũng là một cơ sở dữ liệu chuỗi thời gian được tối ưu hóa để thu thập dữ liệu chuỗi thời gian.
- | Nó là một trong những công cụ hấp dẫn để thu thập số liệu hệ thống.
- | Trong số các dự án Graphite, có một mô-đun API cung cấp các tệp biểu đồ dưới dạng API có tên là Graphite-Web hoặc cung cấp dữ liệu số. Dựa trên điều này, nhiều ứng dụng bảng điều khiển mã nguồn mở đang được phát triển.
- | Tất cả những gì bạn cần làm là sắp xếp dữ liệu trên Graphite và tùy chỉnh bảng điều khiển theo ý thích của bạn.
- | Trong bài học này, chúng ta sẽ sử dụng một công cụ bảng điều khiển có tên là Grafana và xây dựng một hệ thống giám sát thông qua hình ảnh trực quan do công cụ này cung cấp.

Công cụ trực quan hóa mã nguồn mở ... Graphite

Components of Architecture

Collector

- Module để tích lũy dữ liệu trên graphite
- Do tính chất của cơ sở dữ liệu chuỗi thời gian, cần có ba loại dữ liệu: dữ liệu, không gian tên, thời gian, dữ liệu và thông tin này được gửi đến Carbon-cache.

Carbon-cache

- Bộ thu thập dữ liệu nhận dữ liệu do bộ thu thập gửi và lưu trữ trong Whisper

Whisper

- Các mô-đun thực sự lưu trữ và đọc dữ liệu trong hệ thống tệp

Graphite-Web

- Đọc dữ liệu được lưu trữ trong Whisper thông qua giao thức HTTP và xuất ra ở định dạng tệp hình ảnh hoặc dữ liệu.
- Cung cấp chức năng API và bảng điều khiển để cung cấp dữ liệu theo mặc định

Bài 4.

Trực quan hóa dữ liệu

- | 4.1 Khái niệm trực quan hóa và tầm quan trọng
- | 4.2 Công cụ trực quan hóa mã nguồn mở: Grafana

Tổng quan

- █ Bảng điều khiển đa năng mã nguồn mở, Trình tạo đồ thị
- █ Bảng điều khiển đa năng mã nguồn mở, Trình tạo đồ thị
- █ Hỗ trợ graphite, influxdb và opentsdb làm nền tảng phụ trợ
- █ Ngày càng xuất hiện nhiều công cụ hơn những dấu gạch ngang dựa trên graphite nhưng ngày nay Grappana là công cụ phổ biến nhất.



Grafana



Thiết kế bảng điều khiển với Grafana

Chạy Grafana

- | Trước tiên, bạn cần Elasticsearch để chạy Grafana.
- | Elasticsearch được dùng để lưu trữ cấu hình, không phải lưu trữ dữ liệu.

```
$ docker pull dockerfile/elasticsearch  
$ docker run -d -p 9200:9200 -p 9300:9300 dockerfile/elasticsearch
```

- | Chạy Grafana

```
$ docker pull nacyot/grafana  
$ docker run -d -p 8001:8000 nacyot/grafana
```

Tệp Dockerfile

- | Grafana là ứng dụng web được xây dựng trên AngularJS, được tạo ra dưới dạng dự án JavaScript.
- | Ứng dụng không có tầng máy chủ như Rails hoặc Django.
- | Nhìn vào quá trình cài đặt, tải ứng dụng Grafana, cài đặt tại vị trí thích hợp và chạy dựa trên máy chủ Apache.

Tệp Dockerfile

```
FROM centos:centos6
MAINTAINER Name <email>

RUN rpm -iUvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
RUN yum update -y
RUN yum -y install httpd git
RUN cd /opt; git clone https://github.com/grafana/grafana.git
RUN cd /opt/grafana; git checkout $(git describe --tags `git rev-list --tags --max-count=1`)

RUN rm -f /etc/localtime
RUN cp -p /usr/share/zoneinfo/Japan /etc/localtime
```

Mặc dù các phần khác nhìn chung tương tự nhau nhưng bạn có thể đặt giờ địa phương trong Dockerfile. Nếu không sử dụng giờ địa phương thì vùng chứa Docker sẽ sử dụng thời gian UTC.

```
$ docker run -i -t ubuntu bash
root@1f27c50b400d:/# date
Fri Sep 10 01:05:48 UTC 2019
```

Tệp Dockerfile(2)

```
RUN mkdir -p /opt/grafana/src/config  
  
ADD ./config.js /opt/grafana/src/config.js  
ADD ./grafana.conf /etc/httpd/conf.d/grafana.conf  
ADD ./setup_configs.sh /opt/grafana/setup_configs.sh  
ADD ./run.sh /opt/grafana/run.sh  
  
ENV ES_API_HOST 172.17.42.1  
ENV ES_API_PORT 9200  
ENV GRAPHITE_API_HOST 172.17.42.1  
ENV GRAPHITE_API_PORT 8000  
  
WORKDIR /opt/grafana  
  
EXPOSE 8000  
CMD ./setup_configs.sh && ./run.sh
```

config.js và setup_configs.js

- ─ config.js
 - ─ Máy chủ Elasticsearch mà nó chỉ định ở đây được sử dụng đơn giản để lưu trữ cấu hình môi trường
 - ─ Dữ liệu từ Graphite.
 - ─ Thông tin máy chủ được lưu trữ trong các biến môi trường được thay thế trong config.js

```
define(['settings'],  
  function (Settings) {  
    "use strict";  
  
    return new Settings({  
      elasticsearch: "ES_API_HOST:ES_API_PORT",  
      datasources: {  
        graphite: {  
          type: 'graphite',  
          url: 'GRAPHITE_API_HOST:GRAPHITE_API_PORT',  
          default: true,  
          render_method: 'GET'  
        }  
      },  
      default_route: '/dashboard/file/default.json',  
      timezoneOffset: null,  
      grafana_index: "grafana-dash",  
      panel_names: [  
        'text',  
        'graphite'  
      ]  
    });  
  });
```

config.js và setup_configs.js

setup_configs.conf

```
#!/bin/bash

[ -f /opt/grafana/src/config/config.js ] && cp /opt/grafana/src/config/config.js /opt/grafana/src/config.js

sed -i -e "s/ES_API_HOST/${ES_API_HOST}/g" ./src/config.js
sed -i -e "s/ES_API_PORT/${ES_API_PORT}/g" ./src/config.js
sed -i -e "s/GRAFITE_API_HOST/${GRAFITE_API_HOST}/g" ./src/config.js
sed -i -e "s/GRAFITE_API_PORT/${GRAFITE_API_PORT}/g" ./src/config.js
```

- Trong trường hợp gắn tệp cấu hình riêng biệt, hãy đảm bảo ưu tiên tệp config.js tương ứng
- Theo mặc định, hình ảnh này giả định kết nối với máy chủ Elasticsearch và Graphite, tuy nhiên Graphite có thể chỉ định nhiều máy chủ làm phần mềm phụ trợ dữ liệu hoặc InfluxDB.
- Phần còn lại thay thế cấu hình máy chủ.

grafana.conf và run.sh

Cấu hình Apache

```
Listen 80
<VirtualHost *:80>
    ServerName grafana
    DocumentRoot "/opt/grafana/src"
</VirtualHost>
```

- Grafana không có máy chủ web riêng, bạn nên chạy ứng dụng bằng công cụ x hoặc apache.

run.sh

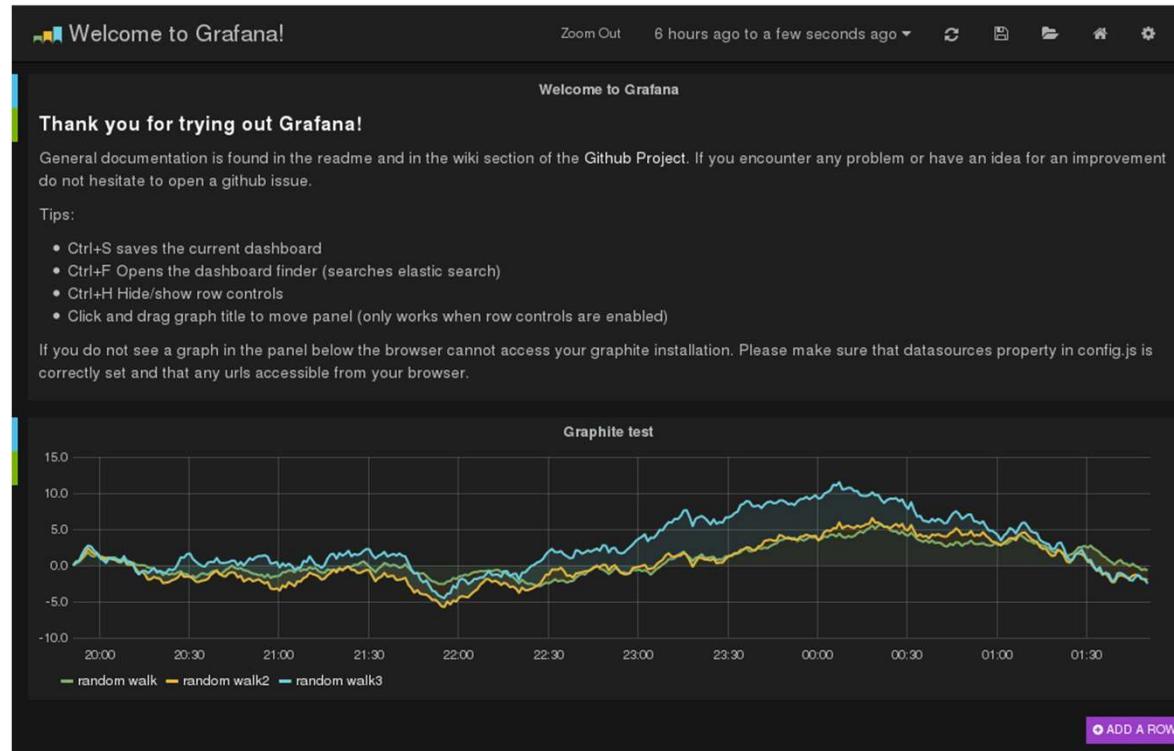
```
#!/bin/bash

/usr/sbin/httpd -d . -f /etc/httpd/conf/httpd.conf -e info -DFOREGROUND
```

- run.ch là tệp chạy phần mềm Apache. - Tùy chọn **DFOREGROUND** chạy Apache ở nền trước, không phải trình nền.

Sử dụng Grafana

- | Trước đó, chúng ta đã kết nối nội bộ với cổng 8001 khi chạy phần mềm. Truy cập 0.0.0.0:8001 trên trình duyệt web.
- | Kiểm tra xem có hoạt động bình thường không.



Grafana - đồ thị randomWalk

Mã nguồn Grafana

Tệp Dockerfile

- <https://github.com/nacyot/docker-graphite>

Hình ảnh Docker tham khảo

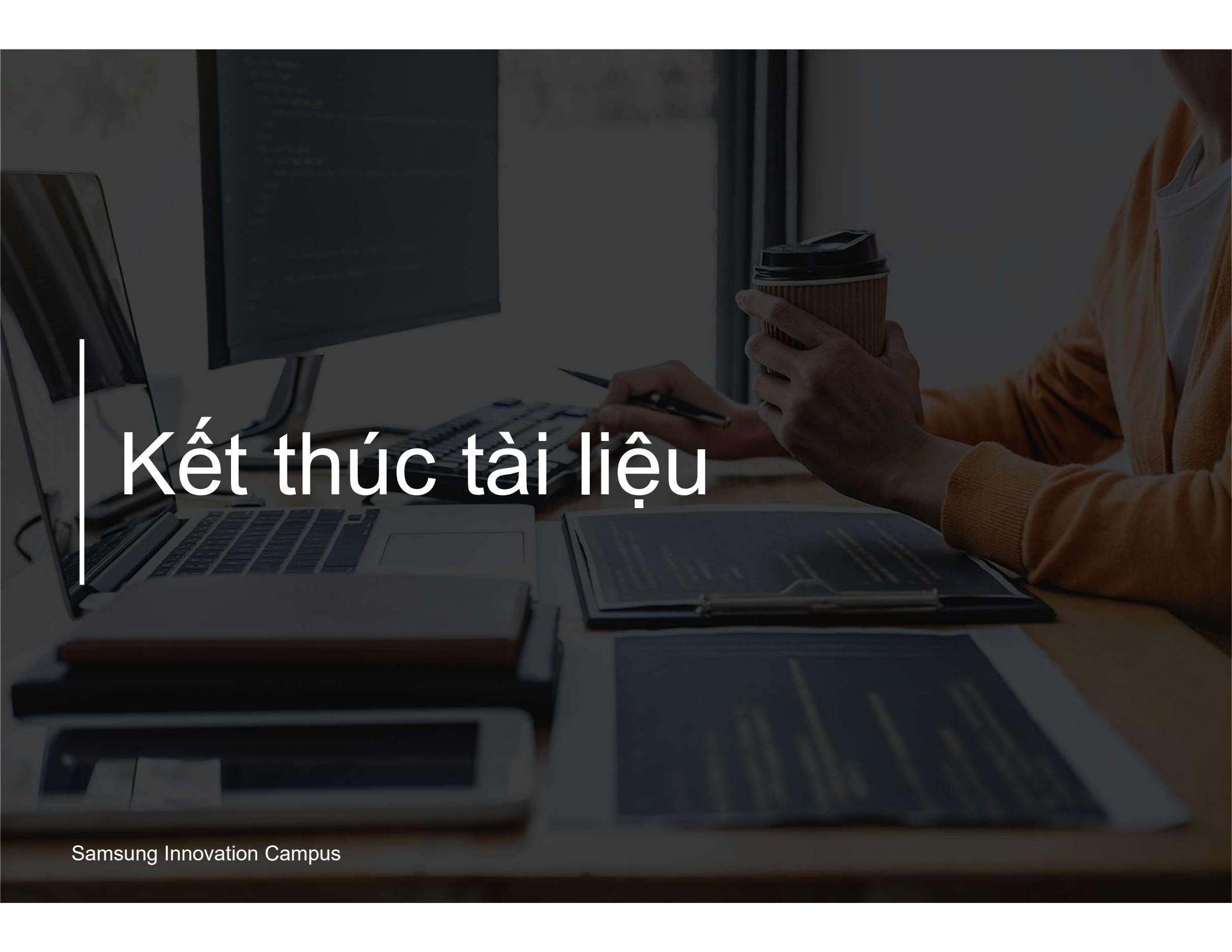
- <https://registry.hub.docker.com/u/nacyot/whisper/>
- <https://registry.hub.docker.com/u/nacyot/carbon-cache/>
- <https://registry.hub.docker.com/u/nacyot/graphite-web/>
- <https://registry.hub.docker.com/u/nacyot/grafana/>

Công cụ giám sát mã nguồn mở

- | Bạn sẽ không gặp vấn đề gì lớn khi sử dụng những hình ảnh để thử nghiệm.
- | Với Docker, mọi thứ mô tả ở trên đều được nén thành năm câu lệnh sau.

```
$ docker run --name whisper nacyot/whisper
$ docker run -d -p 2003:2003 -p 2004:2004 -p 7002:7002 --volumes-from whisper -e NODE_NAME=cache
nacyot/carbon-cache
$ docker run -d -p 8000:80 -e CARBONLINK_HOSTS="172.17.42.1:7002" --volumes-from whisper
nacyot/graphite-web
$ docker run -d -p 9200:9200 -p 9300:9300 dockerfile/elasticsearch
$ docker run -d -p 8001:8000 nacyot/grafana
```

- | Graphite là công cụ giám sát mã nguồn mở, cùng nhiều công cụ khác thiết lập lên một hệ sinh thái khá tốt.
- | Có thể sử dụng các công cụ như Collected, Diamond, Metricsd và Sensu để thu thập số liệu từ tất cả máy chủ được quản lý và nếu tải nặng hơn, bạn có thể mở rộng quy mô như đã đề cập ở trên hoặc sử dụng Statsd làm máy chủ đệm.

A photograph of a person's hands and arms. They are wearing a yellow long-sleeved shirt. In their left hand, they hold a white paper coffee cup with a black lid. In their right hand, they hold a black pen. They are resting their hands on a light-colored wooden desk. On the desk, there is an open laptop with its screen facing the viewer, showing some code or text. To the right of the laptop is a black computer monitor. In front of the laptop is a black keyboard. To the right of the keyboard is a stack of several thick, dark-colored books. A small portion of a white smartphone is visible on the far left edge of the desk.

Kết thúc tài liệu



Together for Tomorrow!
Enabling People

Education for Future Generations

©2022 SAMSUNG. All rights reserved.

Samsung Electronics Corporate Citizenship Office holds the copyright of book.

This book is a literary property protected by copyright law so reprint and reproduction without permission are prohibited.

To use this book other than the curriculum of Samsung Innovation Campus or to use the entire or part of this book, you must receive written consent from copyright holder.