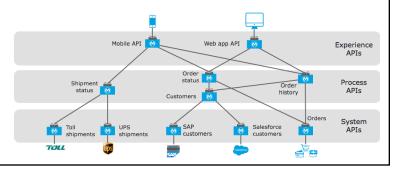


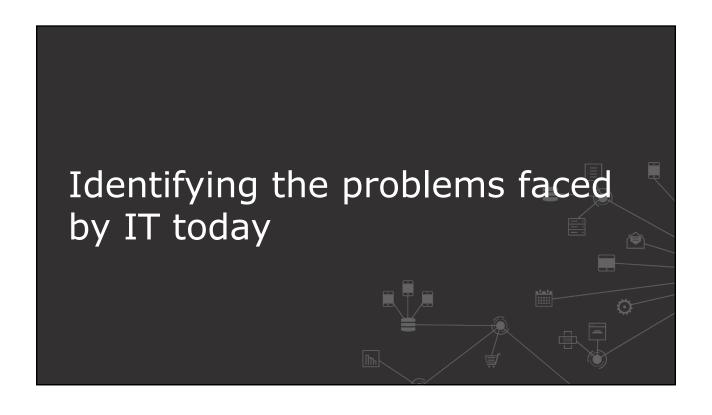
## Module 1: Introducing Application Networks and API-Led Connectivity

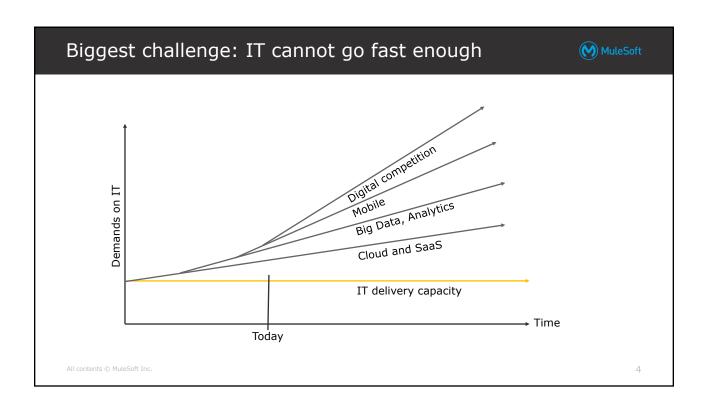
#### At the end of this module, you should be able

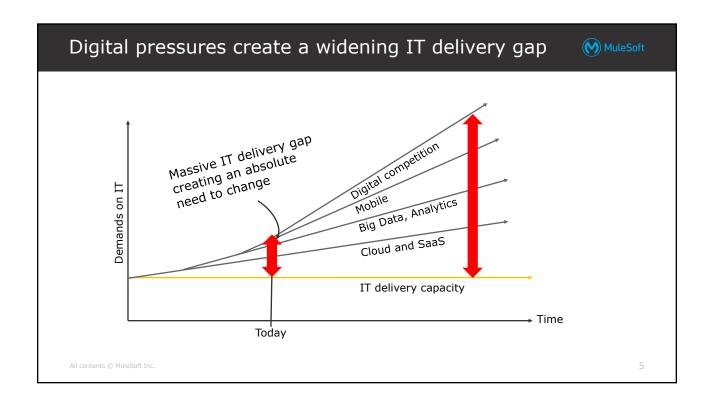


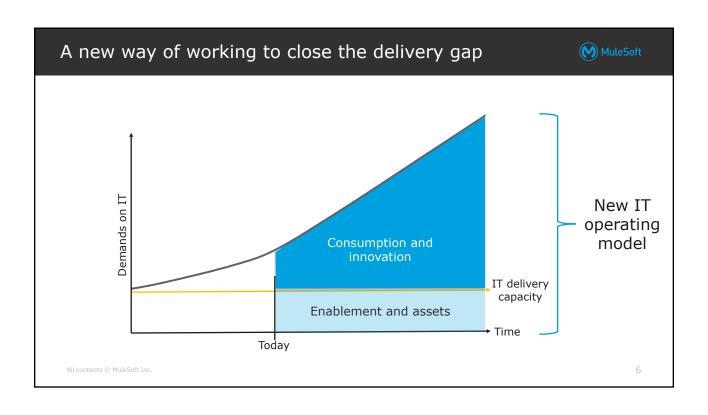
- Explain what an application network is and its benefits
- Describe how to build an application network using API-led connectivity
- Explain what web services and APIs are
- Explore API directories and references
- Make calls to secure and unsecured APIs

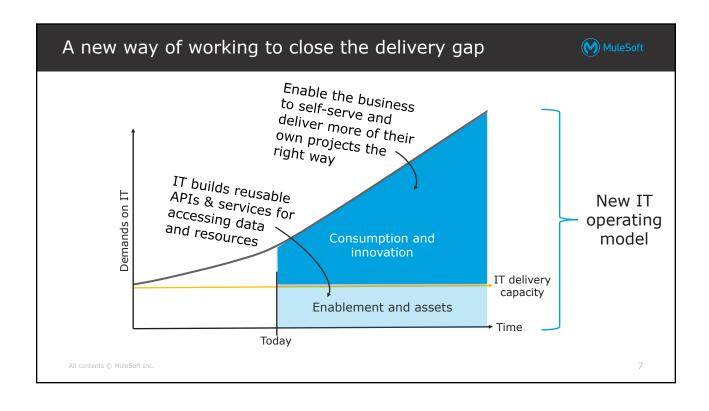




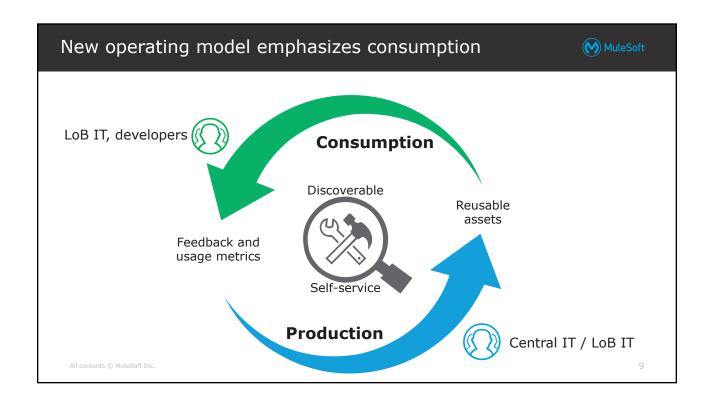


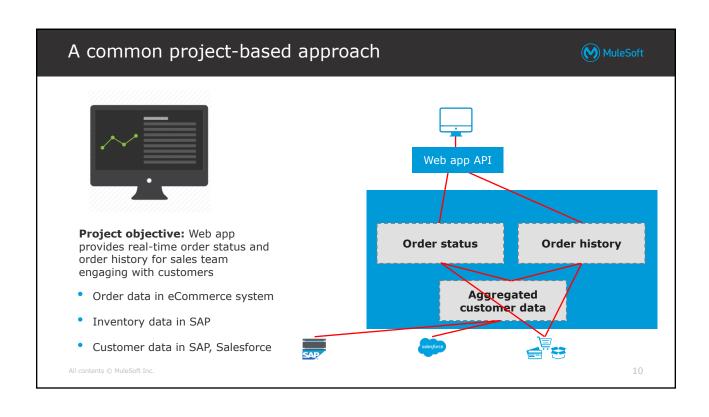


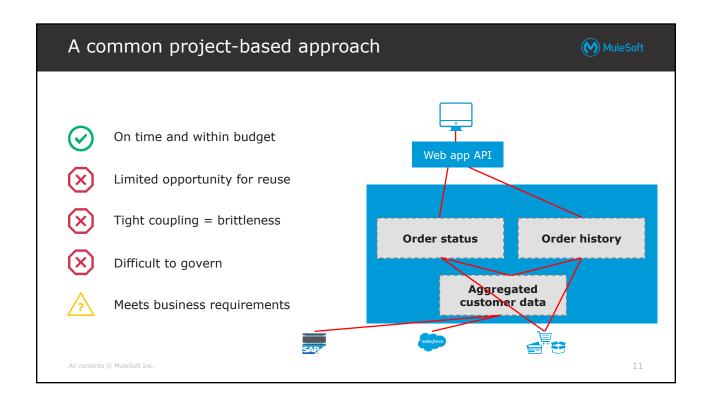


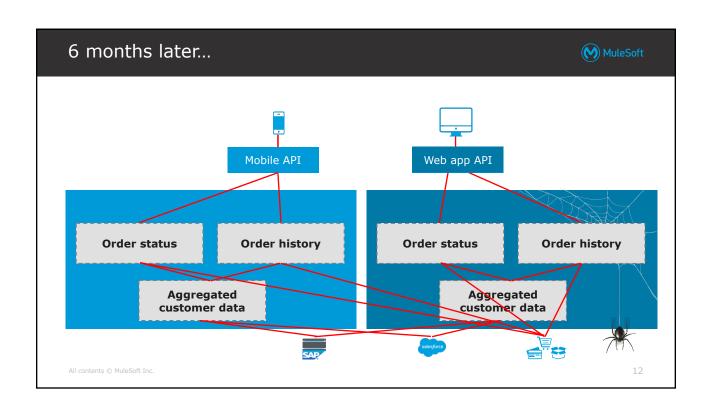


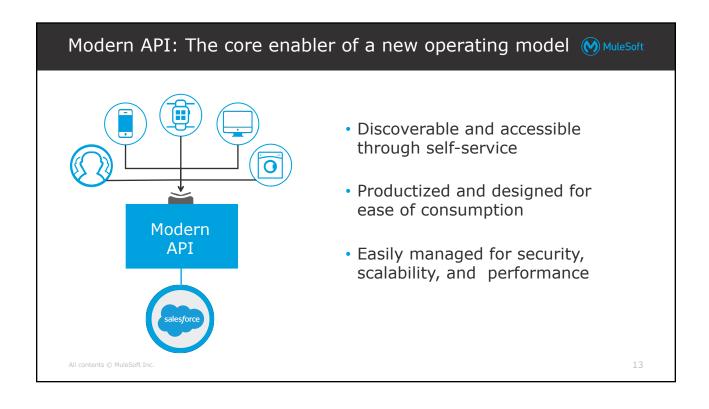


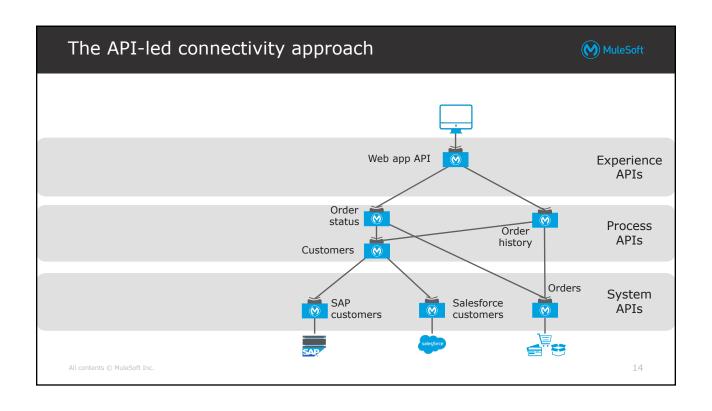


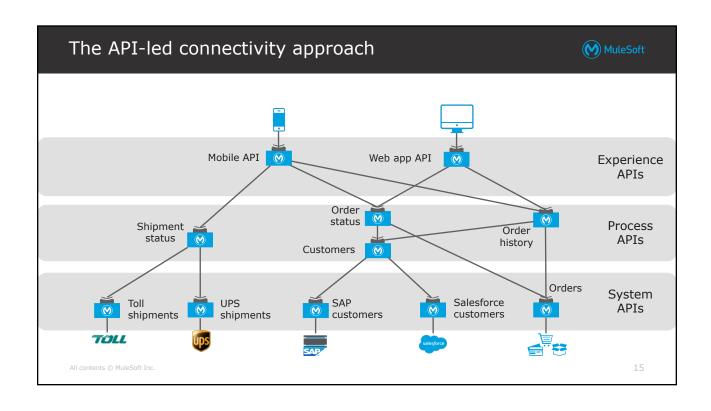


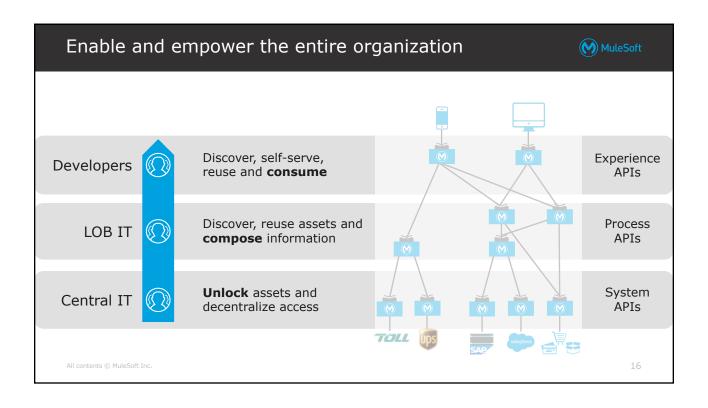


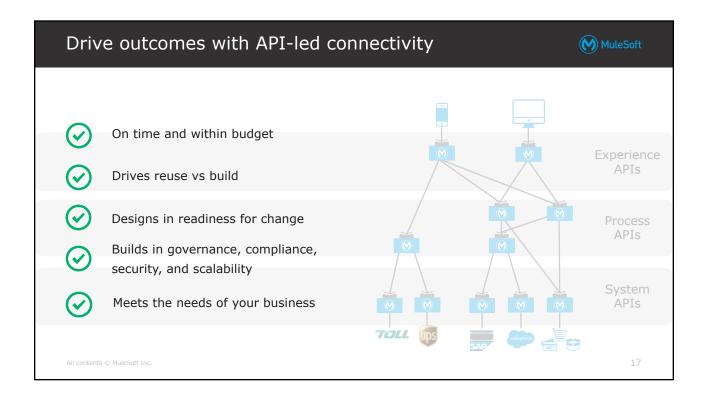


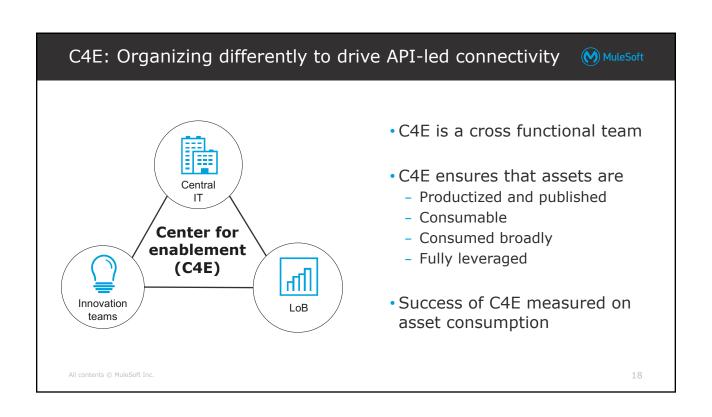


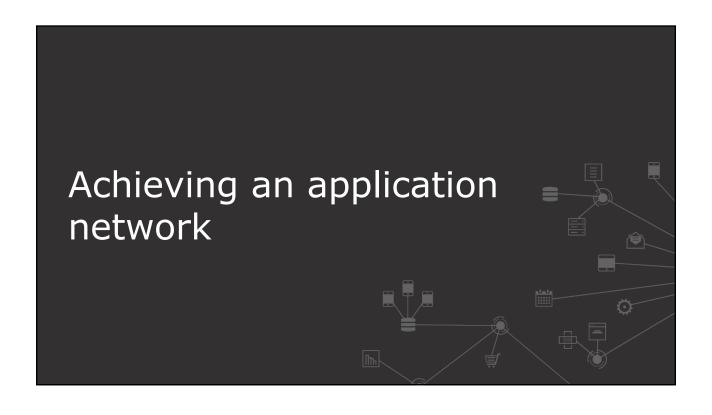


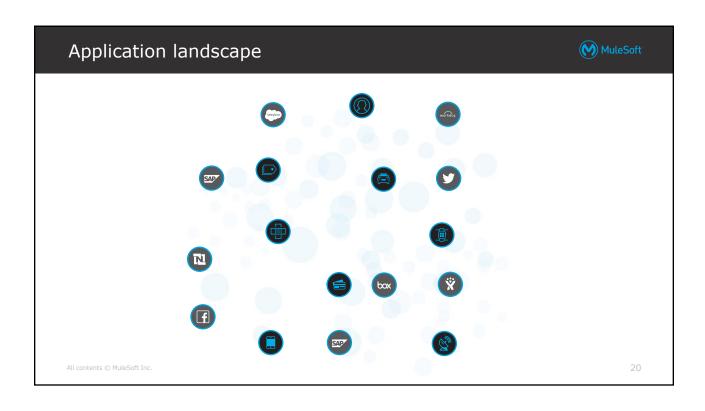


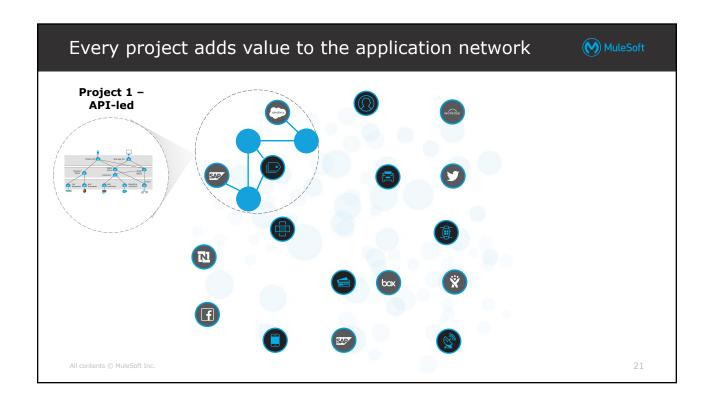


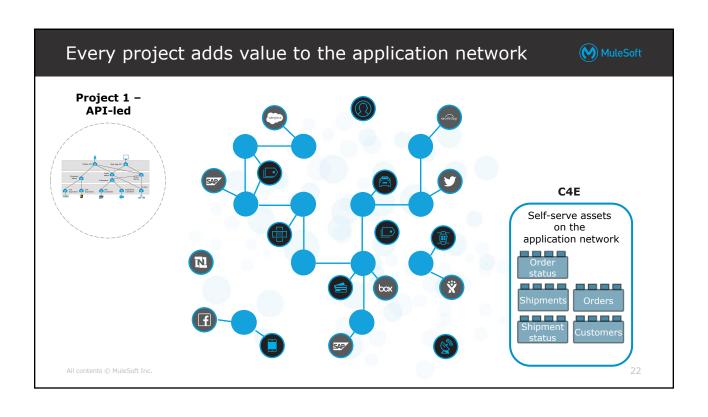


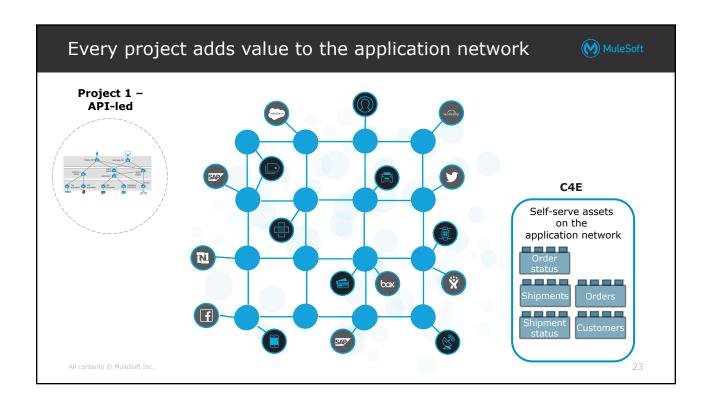


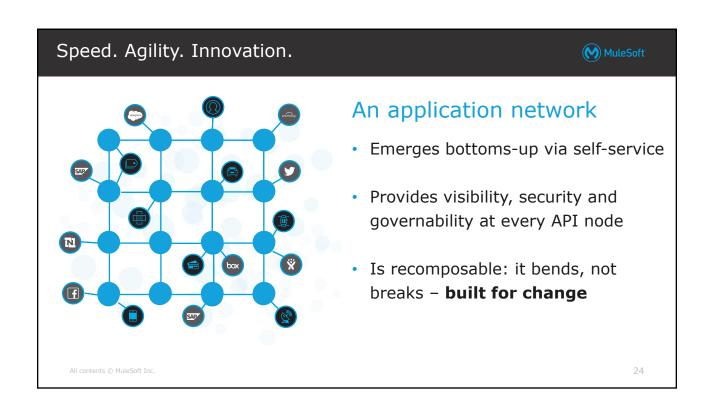


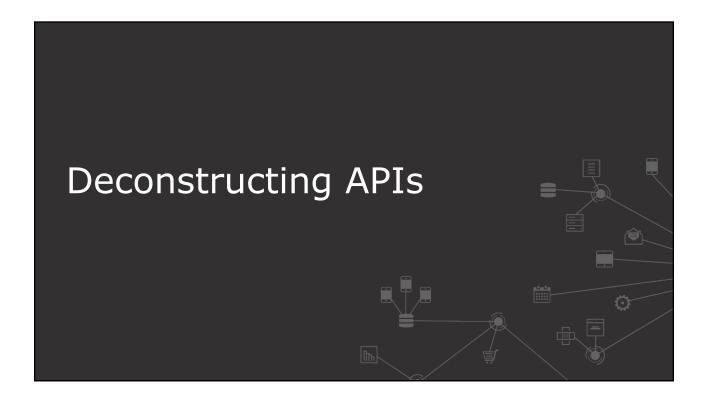












#### What exactly is an API?



- An API is an Application Programming Interface
- It provides the information for how to communicate with a software component, defining the
  - Operations (what to call)
  - Inputs (what to send with a call)
  - Outputs (what you get back from a call)
  - Underlying data types
- It defines functionalities independent of implementations
  - You can change what's going on behind the scenes without changing how people call it

#### What do people mean when they say API?



They could be referring to a number of things...

#### 1. An API interface definition file

- Defines what you can call, what you send it, and what you get back

#### 2. A web service

The actual API implementation you can make calls to or the interface of that API implementation

#### 3. An API proxy

 An application that controls access to a web service, restricting access and usage through the use of an API gateway



#### What is a web service?



- Different software systems often need to exchange data with each other
  - Bridging protocols, platforms, programming languages, and hardware architectures
- A web service is a method of communication that allows two software systems to exchange data over the internet
- Systems interact with the web service in a manner prescribed by some defined rules of communication
  - How one system can request data from another system, what parameters are required, the structure of the return data, and more

All contents © MuleSoft Inc.

#### The parts of a web service



- The web service API
  - Describes how you interact with the web service
  - It may or may not (though it should!) be explicitly defined in a file
  - It could be any sort of text in any type of file but ideally should implement some standard API description language (or specification)
- The web service interface implementing the API
  - Is the code providing the structure to the application so it implements the API
  - This may be combined with the actual implementation code
- The web service implementation itself
  - Is the actual code and application

#### Two main types of web services



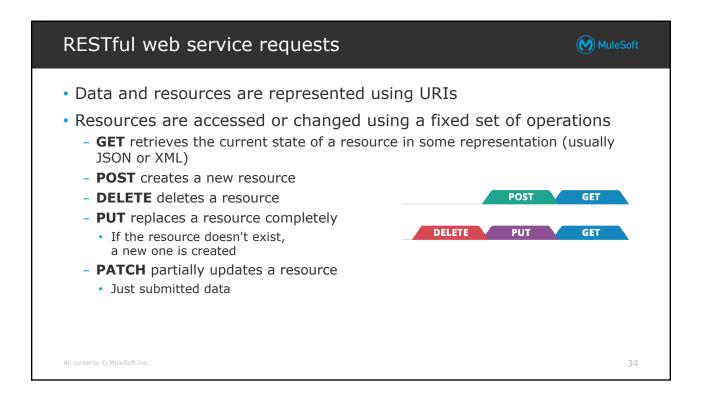
- SOAP web services
  - Traditional, more complex type
  - The communication rules are defined in an XML-based WSDL (Web Services Description Language) file

#### RESTful web services

- Recent, simpler type based on representational state transfer (REST) based communications
- Use the existing HTTP communication protocol



# REST stands for Representational State Transfer An architectural style where clients and servers exchange representations of resources using standard HTTP protocol Other systems interact with the web service using the HTTP protocol The HTTP request method indicates which operation should be performed on the object identified by the URL GET, POST, DELETE, PUT, PATCH



#### Example RESTful web service calls



- (GET)/companies
- (GET)/companies?country=France
- (GET)/companies/3
- (POST)/companies with JSON/XML in HTTP body
- (DELETE)/companies/3
- (PUT)/companies/3 with JSON/XML in HTTP body

All contents © MuleSoft Inc.

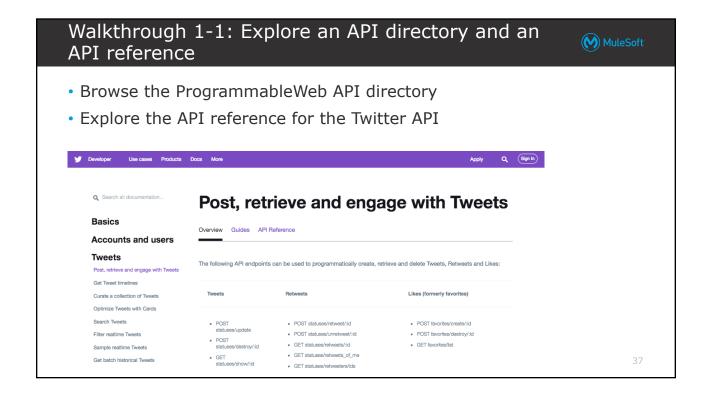
35

#### Example RESTful web service response

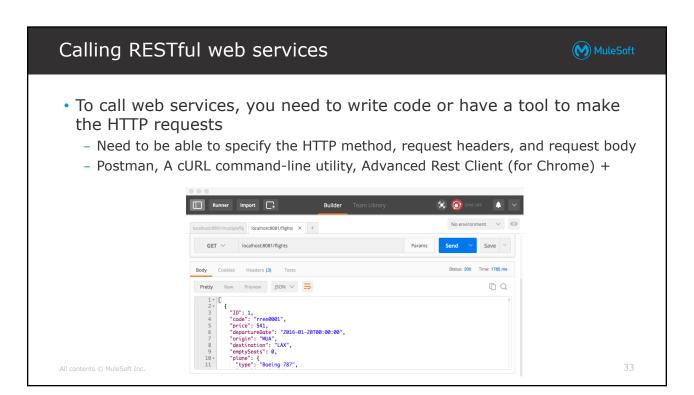


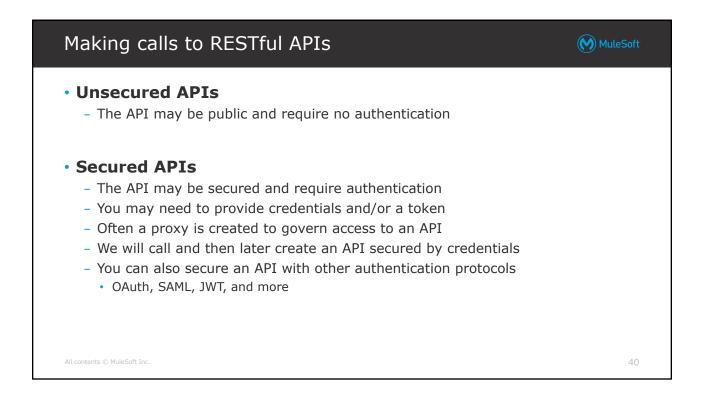
- JSON (JavaScript Object Notation)
  - A lightweight data-interchange format (without a lot of extra XML markup )
  - Human-readable results (usually JSON or XML)
  - Supports collections and maps











### Getting responses from web service calls

MuleSoft

- RESTful web services return an HTTP status code with the response
- The status code provides client feedback for the outcome of the operation (succeeded, failed, updated)
  - A good API should return status codes that align with the HTTP spec



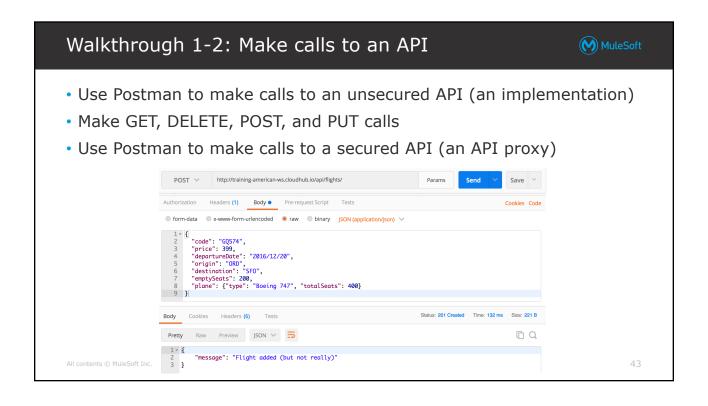
#### Common HTTP status codes

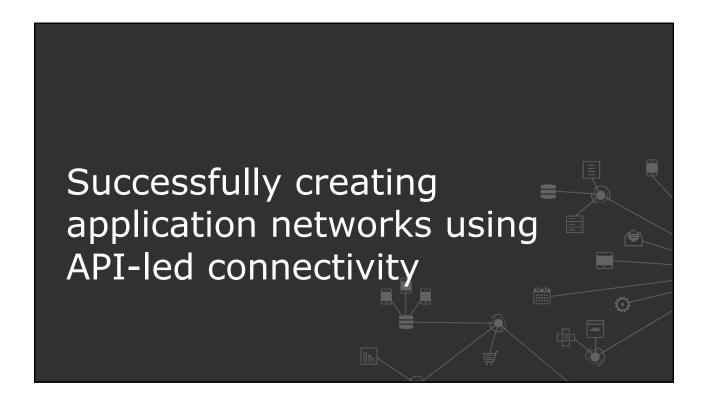


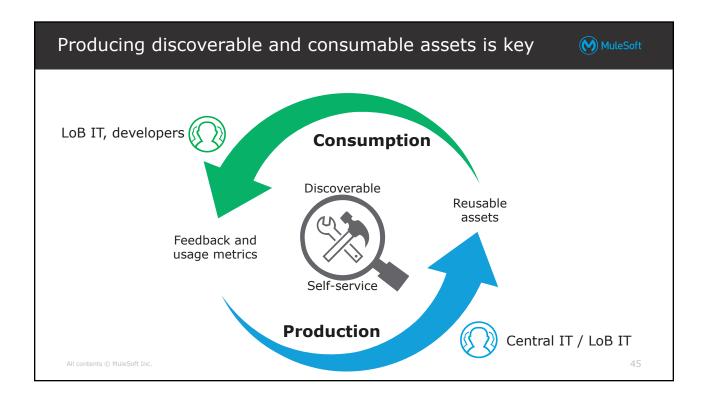
41

Code	Definition	Returned by
200	OK – The request succeeded.	GET, DELETE, PATCH, PUT
201	Created – A new resource or object in a collection.	POST
304	Not modified – Nothing was modified by the request.	PATCH, PUT
400	Bad request – The request could not be performed by the server due to bad syntax or other reason in request.	All
401	Unauthorized – Authorization credentials are required or user does not have access to the resource/method they are requesting.	All
404	Resource not found – The URI is not recognized by the server.	All
500	Server error – Generic something went wrong on the server side.	All

21



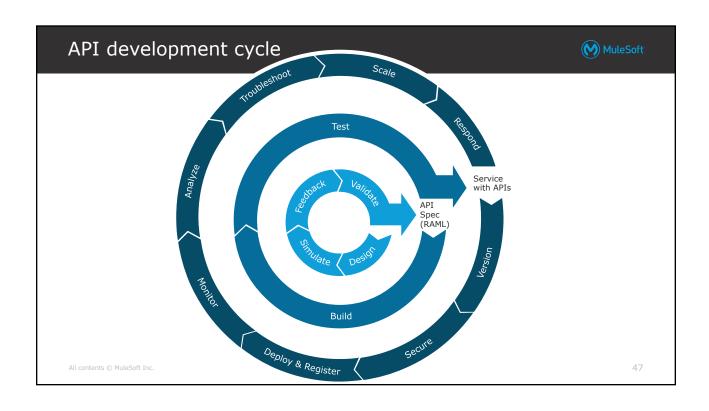


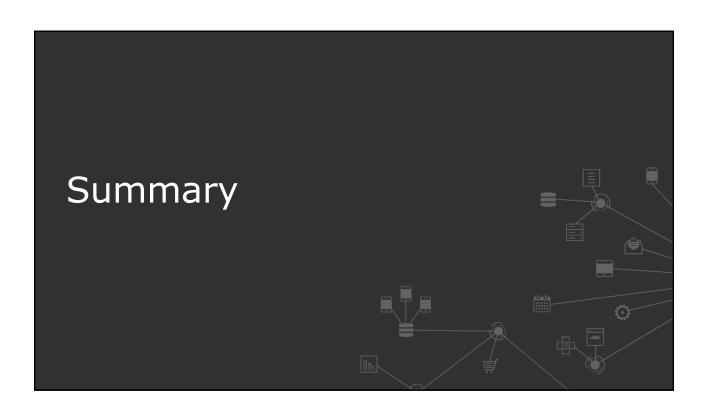


#### Designing for API success



- Create APIs that developers can find and want to use and share with others
  - Design the API for the business use cases it will fulfill, not to model the backend services or applications they expose
  - Focus on performance of client applications and user experience
- Take an API design-first approach!
- Get API design right before investing in building it
  - Define it iteratively getting feedback from developers on its usability and functionality along the way
  - Building the implementation of an API is time consuming and expensive to undo





#### Summary



- Companies today need to rapidly adopt and develop new technologies in order to stay relevant to customers & keep competitive
- IT needs to be able to rapidly integrate resources and make them available for consumption
  - An API-led connectivity approach can help achieve this
- To drive API-led connectivity, create a C4E (Center for Enablement)
  - A cross-functional team to ensure assets across the organization are productized, published, and widely consumed
- An application network is a network of applications, data, and devices connected with APIs to make them pluggable and to create reusable services

All contents © MuleSoft Inc. 49

#### Summary



- A web service is a method of communication that allows two software systems to exchange data over the internet
- An API is an application programming interface that provides info for how to communicate with a software component
- The **term API** is often used to refer to any part of RESTful web service
  - The web service API (definition file)
  - The web service interface implementing the API
  - The web service implementation itself
  - A proxy for the web service to control access to it
- **RESTful** web services use standard HTTP protocol and are easy to use
- The HTTP request method indicates which operation should be performed on the All content object identified by the URL 50