



Industry Guide: API Regions

By Kin Lane ~ September, 2019

The seamless deployment and operation of APIs within multiple geographic regions is quickly become a critical need for leading API providers due to a global demand for their services and an increase in regulation and data sovereignty when doing business around the world.



About

API Evangelist is a research site established in 2010 to study the technology, business, and politics of doing APIs. It's mission is to produce regular short and long form content to help people make sense of the API economy.

apievangelist.com | [@apievangelist](https://twitter.com/apievangelist)

Powered by Tyk

This API Industry Guide is bought to you by Tyk, the Cloud Native API Gateway and Management Platform, featuring API Gateway, Analytics, Developer Portal and Dashboard.

Tyk powers billions of transactions per day for thousands of innovative organisations including AXA, Cisco, Starbucks, and the Financial Times.

Fast, flexible, and highly-performant, Tyk is available as a self-managed installation or SaaS platform, all with our original open source Gateway at their core.

We're creating the next generation of API management, so you can focus on building things, better.

tyk.io | [@tyk_io](https://twitter.com/tyk_io)



Table of Contents

| | |
|--|----|
| Industry Guide: API Regions | 1 |
| By Kin Lane - September, 2019 | 1 |
| Regional API Deployment | 4 |
| Cloud Computing Fundamentals | 5 |
| Regional Service Availability | 6 |
| Other Regional API Services | 7 |
| Why Deploy APIs In Multiple Regions | 8 |
| Multi-Region API Deployment Considerations | 10 |
| Deployment | 10 |
| Availability | 12 |
| Operations | 13 |
| Orchestration | 14 |
| Revenue | 15 |
| Governance | 16 |
| Legal | 18 |
| Health | 19 |
| Outreach and Support | 21 |
| Break Existing Consumers Down By Region | 22 |
| Awareness Of API Performance Across Regions | 23 |
| Starting Out Small With A Single Region | 24 |
| Thinking Beyond A Single API Region | 25 |

Regional API Deployment

The seamless deployment and operation of an API within multiple geographic regions is quickly becoming a critical need for leading API providers due to a global demand for their services and the increasing regulation and data sovereignty faced when doing business around the world.

There is no single correct approach to deploying APIs into multiple geographic regions, and is something that will take dedicated investment, but thankfully the top cloud computing providers, and a growing number of other more speciality API infrastructure providers have been making it easier than ever to deploy and operate your APIs exactly where you need them.

There are many reasons for deploying your APIs into multiple regions, and to help me better understand the landscape I wanted to explore what many of the reasons might be, and what the considerations throughout the API lifecycle for deploying, operating, scaling, and supporting truly multi-region, and even multi-cloud API infrastructure.

To help me better understand how APIs are being deployed, distributed, scaled, and localized at the region, zone, and city level, I wanted to take a walk through my API research and spend some time thinking deeply about how this dimension of API operations is changing how we deliver and consume APIs in 2019.

Cloud Computing Fundamentals

Regions, availability zones, cities, and location specific delivery of Internet infrastructure isn't uniquely API, and is something that has emerged out of the cloud computing evolution as a fundamental element of how we put the cloud to work.

Amazon led the march for regional cloud computing infrastructure, but Azure, Google, Heroku, and other providers have followed their lead and began making their API-driven services available in a host of geographic regions.

Like with most things in the cloud, what service providers view as an individual unit of measurement varies from platform to platform, keeping things an apples to apples comparison, but with a mix of varieties being served up. Amazon Web Services states that they operate in 69 availability ones within 22 geographic regions, with Azure operating within 54 regions in 140 countries, and Google describes what they do in 20 regions, 61 zones, 135 network locations, within 200+ countries and territories.

No matter how it is described, the top three cloud platforms provide us with an unprecedented amount of low cost resources to help us manage our API infrastructure in a much more geographically precise way.

Regional Service Availability

Each of the cloud platforms have a different approach to making their services available in different regions. Like with how regions are viewed, what is compute, storage, DNS, and other essential building blocks of cloud infrastructure will vary from platform to platform, with different levels of availability and maturity of services within each geographic region.

Each cloud platform is working to rapidly expand their geographic reach, usually starting with the essentials like compute and storage, and then slowly phasing in other services over time. Leaving a patchwork of service availability, when you look at what you can do on the ground with each platform and location.

The differences and availability of cloud services in different geographic regions is what makes operating multi-cloud an easy thing to say, but much more difficult to actually realize. Something API providers are working to abstract away the differences and friction, at the same time where some of the cloud platforms seem to be slipping towards more lock-in. While multi-region API deployment within a single cloud platform is still a significant amount of work, it is very possible to realize..

However, doing a multi-region, multi-cloud API deployment will be significantly more of a challenge, because of the nuance between platforms, regions, and the services that they offer. This is something that Kubernetes and other orchestration providers would like to solve, but for now it remains an elusive dream for most API providers.

Other Regional API Services

Beyond the top cloud platforms, other individual API providers have been following the lead of these cloud giants, and in some cases actually launching their own data centers in regions, or simply piggybacking on the regional capabilities of AWS, Azure, and Google. Demonstrating how regions are shifting the game when it comes to delivering essential resources to desktop, web, mobile and device applications.

Here are a handful of API Providers who make regional service availability to their consumers:

- [CloudFlare](#) - DNS infrastructure APIs across multiple regions.
- [Twilio](#) - Voice client tooling and APIs in multiple regions.
- [MailGun](#) - Providing email messaging APIs in multiple regions.
- [Riot Games](#) - Scaling gaming operations in multiple regions.
- [RunScope](#) - Delivering API monitoring services in multiple regions.
- [Oracle](#) - Cloud resources in a variety of global regions.
- [Whisper](#) - Layered communication resources in multiple regions.

These API providers offer real world examples of how APIs can be made available in different regions to help meet specific business objectives. Helping paint a picture of the waves that are emerging across the API sector empowering API consumers with valuable digital data, content, and other resources to build applications that service specific geographic regions. Before we dive into many of the considerations we should be thinking about when deploying APIs in multiple regions, let's first explore why API providers are deploying APIs into multiple geographic locations in the first place.

Why Deploy APIs In Multiple Regions

There are many compelling reasons for delivering APIs into multiple regions, but there are also many technical considerations, costs, and hills you will have to climb before you can ever realize your multi-regional vision. Depending on how your APIs are currently developed, published, managed, and consumed, the scope of investment in multi-region capabilities will vary, with the motivating factors behind the justification of costs and time investment being equally unstable.

Here are some of the reasons API providers are deploying resources into multiple regions:

- **Availability** - Ensuring that your consumers can always get at what they need.
- **Regulations** - There are industry regulations in place that require change.
- **Intellectual Property** - Maintaining control over your intellectual property.
- **Licensing** - There are licensing limitations over your data, content, and algorithms.
- **Sovereignty** - The country you want to operate in views data differently.
- **Localization** - Being efficient in localizing data, content, and other resources.
- **Performance** - Being able to increase the performance of your API infrastructure.
- **Experience** - Localizing and optimizing the overall experience around resources.

- **Reliability** - Establishing a reputation with your consumers for being reliable.
- **Taxes** - Able to isolate your revenue to a region where taxation is favorable to business.

These are just a few of the reasons API providers are looking at deploying, managing, scaling, and supporting APIs within multiple geographic regions. Some of these areas are more about the desires of companies, organizations, institutions, and government agencies. However, some of these areas are mandated and out of control of some companies, organizations, and institutions. As uncertainty around securely doing business on the web increases, the urgency and reasons behind delivering multi-region API infrastructure will only increase.



Multi-Region API Deployment Considerations

Moving beyond the reasons why API providers will want or need to operate multi-cloud infrastructure, I wanted to walk through what some of the considerations might be for API providers on this journey. Exploring how regional operations will shift how we do things across multiple stops along the API lifecycle. Helping set the stage for conversations around operating API infrastructure at scale across multiple geographic regions, and possibly even across multiple cloud platforms.

To produce this guide I've taken existing stops along the API lifecycle from my API industry research and used them as a scaffolding, but then got to work evaluating the challenges and friction a multi-regional approach might introduce. Shining a light on how we can be more strategic when it comes to delivering not just the technology of our APIs into multiple regions, but also look at the business and politics of operating API infrastructure in multiple regions, countries, and other location based dimensions of delivering APIs.

Deployment

The actual deployment of API infrastructure is the first stop on this journey. Thinking about the different ways in which APIs are being deployed today, and looking at some of the ways next generation technology is reducing friction for scaling across geographic regions, while also considering some of the possible constraints for realizing a multi-region approach because of how our APIs are designed and currently operated.

Here are some of the most common areas to consider when you are looking to deploy an API across multiple geographic regions, regardless of the cloud platform you choose:

- **Compute** - Having the ability to deploy the compute layer behind each API being delivered across multiple regions.
- **Serverless** - Leveraging serverless to deploy each individual API resources across

- **Containers** - Employing containers to deploy each individual API resources across multiple regions.
- **Storage** - Ensuring that you have consistent storage available across all of your API deployments.
- **Content** - Operating a content delivery network that is in alignment with the multi-regional deployment of APIs.
- **Database** - Properly sharding and distribution of databases behind API resources being deployed across multi-region API resources.

Each of the top cloud platforms provide you with services to manage the multi-regional deployment of these compute and storage resources, however, depending on how you have designed your API infrastructure, you will face different challenges when actually deploying API infrastructure across multiple regions. Depending on the legacy nature of your APIs, or if you are delivering an entirely new set of APIs, your friction in each of these areas will vary.

The most common challenge API providers face when trying to deploy API infrastructure at scale across multiple regions is in the area of database management. While the cloud platforms will help you in this task, depending on how you structure your database, and the skills you have on your team, operating the databases behind multi-cloud infrastructure will be full of gotchas. Make sure and do your homework, set up and fully test different proof of concept configurations before you settle in on one pattern for operating the database behind your multi-region APIs.

The actual deployment of APIs across multiple geographic regions is just the start. Even with the compute and storage all configured properly you will still have to have a plan for how these instances are accessed by API consumers. Something that will require a deep understanding of your API consumers, as well as a firm grasp on how to shape your Internet topology.

Availability

Once you have your deployment topology defined and setup, the next set of challenges will be all about how available your APIs are, and how you drive and route traffic to them, resulting in a variety of infrastructure needed, and patterns potentially being applied. How API providers approach availability will be where the return on investment of a multi-region API deployment will end up being realized.

Here are some of the areas of consideration when it comes to ensuring multi-region API infrastructure will actually be available to consumers in a truly multi-regional way:

- **DNS** - The addressing of APIs, and whether it uses a single domain, or employs regional specific subdomains for routing traffic.
- **Balancing** - Leveraging load balancing solutions for handling the distribution of traffic within regional or cross-regional deployments.
- **Clusters** - Defining clusters that exist across multiple zones, locations, and regions, defining boundaries that reflect or transcend real world geographies.

The choices you make at the API availability layer will define how successful you are at truly delivering API infrastructure in a multi-region way. There is no one right way to achieving multi-region availability for your APIs, so make sure and study how DNS, load-balancing, and clustering patterns, services, and tooling can be put to work. Do your homework, develop real world tests that demonstrate what the developer experience will truly be, and then invest in the approach that achieves the results you desire.

Operations

After deploying your API infrastructure across multiple regions, and dialed in the availability of these APIs, you should begin moving to think about the more operational considerations, and consider how your API operations will expand, grow, and potentially become more complex to operate reliably. Making sure you are properly investing in the operational side of things, and are able to scale operations to meet the regional demands of API consumers.

Here are a few of the primary operational areas you will need to invest in and make sure you are able to properly scale your operations to meet a multi-region demand:

- **Portals** - Retrofitting one portal for multiple locations, or publishing a multiple portals serving each individual region.
- **Authentication** - What strain will multiple regions put on your authentication services, ensuring that it will not be a bottleneck.
- **Management** - Ensuring your core API management keys, plans, rate limits, logging, reporting, and other elements will scale regionally.

These are three areas that will require enterprise grade solutions to meet the regional demands you are creating in each of these operational areas. If you can't properly on-board, secure, and manage the API access needs of multiple API regions, the reasons for investing in multi-region deployments become less clear.

Regional deployment adds an entirely new dimension to the awareness that API management already is bringing to your operations. Having a proper API management strategy, services, and tooling in place is critical to know where your consumers operate, which should inform the road-map for your multi-region API infrastructure rollout. As the number of consumers in a particular region grows you should be determining where the opportunities are for bringing APIs closer to each growing demographic, and potentially considering a new regional deployment.

API providers who do not properly manage their single regionally deployed APIs will never be able to fully realize a multi-region deployment. The maturity and sophistication of your existing API management practices will set the stage for each migration into a new region. Exponentially growing your awareness, or possibly also exponentially grown the dysfunction and chaos, depending on how well you've invested in API management across your existing API operations.

Orchestration

Now that you can properly manage not just a single instance of your APIs, but scale across multiple regions, you might want to take some time to consider how as a provider you will be able to orchestrate changes across your API infrastructure, but also empower your customers to orchestrate how they put APIs to work across their desktop, web, mobile, and device applications.

These are some of the areas of consideration when it comes to investing in multi-regional orchestration capabilities for your API infrastructure:

- **Webhooks** - Enabling region specific webhooks to be defined and triggered by consumers, helping make things happen in real time.
- **Events** - Making events available to consumers so that they can subscribe to changes across individual or multiple regional API infrastructure.
- **Streams** - Providing sustained API connectivity opportunities alongside request / response infrastructure across multiple regions.
- **Mesh** - Investing in seamless discovery, scaling, circuit-breakers, and communication across API infrastructure no matter which region it is deployed in.
- **CI/CD** - Making sure all APIs are integrated and deployed with no matter which region API infrastructure is operating in around the world.
- **Kubernetes** - If containers are in play, using Kubernetes to deploy, scale, and distribute API infrastructure at scale across multiple regions.

Depending on the scope of your API infrastructure, some of these building blocks might not apply to what you are looking to accomplish with your multiple region API deployment. However, I'd say that at least CI/CD, and possibly Kubernetes should be mandatory for most regional deployments, depending on the other deployment decision you've made in your planning process.

Webhooks, events, streaming, and service mesh tend to reflect the needs of more mature API infrastructure. Depending on the provider, some may go multi-region before they dive into these areas, but for others, they may already be providing webhooks, events, and streams to their API consumers, and will want to replicate their robust API toolbox into new regions--significantly expanding the scope of going multi-regional with API operations.

Revenue

One area that can be easily overlooked when planning your multi-region API rollout is in the area of revenue, covering both sides of the coin and making sure you are probably budgeting for the rollout, but also considering how multi-region availability can be used to generate new revenue streams. There are many examples of API providers offering multi-region availability as a premium offering, making two revenue based building blocks pretty critical.

- **Monetization** - What does all of this cost, and what is the strategy to quantifying, tracking, reporting, and reconciling what is being spent to realize a multi-regional API operation.
- **Plans** - How does multi-region deployment fit into API plans, tiers, and access levels, ensuring that new sources of revenue aren't left on the table, and consumer demand is paid attention to.

To help you understand the monetization side of this conversation, look to your cloud platform provider(s) to give you a breakdown of the resources you are putting to work--helping you manage your operational budget, which will ultimately define your monetization strategy. For the plan side of this discussion, your API management provider should have everything you need here to define the tiers of access, and allow you to use region as a variable when crafting your API plans, setting the rate limits, then billing, reporting, and auditing as required.

Governance

With API infrastructure spanning multiple regions you will also need to have a pretty solid API governance strategy in place. Helping you understand all the moving parts of your API operations, while also defining and enforcing standards for how infrastructure gets rolled out to each region being targeted. Making sure governance is addressed early on, rather than getting started after problems arise.

Here are some common building blocks to consider when it comes to multi-region governance of API infrastructure, helping manage the politics of API operations within any particular region:

- **Definitions** - Leveraging machine readable definitions for all schema and API contracts, with versioning that includes machine readable regional deployment templates ensuring consistency and policy enforcement.
- **Discovery** - Putting the machine readable definition catalog to work to make sure all schema and APIs are discoverable at any point throughout the API lifecycle.
- **Design** - Establishing API design standards that span request / response, webhooks, streaming, publish / subscribe deployment patterns across multiple protocols.
- **Policies** - Defining machine readable policies that govern the management, orchestration, monitoring, performance, and security of multi-region API deployments.

- **Blocking** - Having a regional strategy for what data and content are blocked based upon paths, channels, topics, headers, queries, bodies, and other elements of API communication.
- **Filtering** - Having a regional strategy for what data and content gets filtered as part of regular API activity, based upon the request and response bodies of each transaction.
- **Routing** - Providing the ability to route traffic between regions, based upon location, headers, content, and other determining factors.
- **Transformation** - Allowing for data and content to be transformed based upon regional requirements, apply local requirements in real time.
- **Logging** - Having a regional logging strategy that dictates what is logged, where logs are stored, and how long they are stored, including backup, restore, and compliance requirements.
- **Auditing** - Providing the ability to audit all traffic that occurs within a single region, and identify patterns of traffic across regions.
- **Government** - Having a plan for how you deal with the city, state, and federal government requests for information.
- **Reporting** - Reporting upon all activity, broken down by all relevant dimension, allowing for PDF, printed, dashboard, and portable markdown formats to be generated.

There are other building blocks listed elsewhere in this guide that will help contribute to overall API governance. API authentication, management, monitoring, performance, and other areas feed into API governance. Ideally, reporting across all areas of operations, generated throughout the life cycle of each API is aggregated as part of governance reporting. Providing a complete view of what is happening across all regions on any given day, week, or month. Allowing for global governance of API infrastructure, but also allowing for independent regional governance to be applied and realized in real-time. Demonstrating the important role regions will play in helping shape the global web, but also actively participate within spheres of the web that are still subject to existing political notions of the nation state.

Legal

With the addition of each region, you will need to have the legal department take a fresh look at each API in operation. Bringing in the lawyers to help craft new versions of the legal building blocks of API operations will be needed to help govern how things work in any single regional API deployment. Here are a handful of the most common legal building blocks present, which will be impacted by regional API deployment, management, and support.

- **Terms of Use** - What are the terms of services for each individual region.
- **Privacy Policy** - What is the privacy policy for each individual region.
Service Level Agreement - The service level agreement (SLA) for individual regions.
- **Law Enforcement Guidelines** - What are the law enforcement guidelines for each individual region.
- **Transparency Reports** - Where is the transparency report for each individual region.

Depending on your overall API portal and management approach, you might end up having a single legal department for all of your API operations, but then carve out folder for each individual region you end up operating in. If you look at some of the larger API providers like Google, Azure, and Amazon, you will be able to find examples of a very global, but also regional approach to sensibly defining your API legal department.

Health

The overall health of API infrastructure is central to any multi-region API operation. Like API management, monitoring, testing, performance and other health related tasks will define where you should be scaling regionally by helping you understand where your customers are, and where the slowdowns, traffic jams, latency, and other illnesses exist--shaping how you plan for scaling your API across multiple regions.

Here are some of the essential building blocks of quantifying the health of your API platform in a meaningful way that can be used to define which regions you deploy API infrastructure into:

- **Encryption** - Ensuring that all API traffic is encrypted in transport and in storage, in alignment with a regional SSL key strategy.
- **Monitoring** - Making sure to monitor all APIs in all regions from all regions, painting a detailed picture of how your APIs perform. Allowing you to overlay this data with where your API consumers operate, providing the information you need to thoughtfully expand your regional API expansion strategy.
- **Testing** - Similar to monitoring, you should test all APIs in all regions from all regions, making sure you have a clear understanding of whether or not all of your APIs are doing what they should be doing.

- **Performance** - Adding a 3rd dimension, you should be measuring performance of all APIs in all regions from all regions, mapping out how each API performs to customers within a specific region, but also identifying where the latency exists in the network.
- **Network** - Having a clear picture of what the networks look like between each regional API and its consumers, as well as what the pipes look like between each region. Fully understanding what the opportunities for optimization are across how the bits are moved around.
- **Security** - Ensuring that each regional API deployment companies with the overall security practices, are scanned on a regular basis, and leverage pipeline enforcement of common security patterns.
- **Maintenance** - Having regionally focused maintenance strategies for each API, making sure the availability meets the regionally defined SLA.
- **Reliability** - Defining what reliability for each API, providing the SLA, but also making sure monitoring, testing, performance, and security benchmarks are all leveraged to define overall regional API reliability.
- **Observability** - Making sure that all monitoring, testing, performance, and security reports have an externalized reporting element for partners, and for the general public when relevant. Providing observability into the overall health of the API platform, globally, and by individual regions.

Optimal API platform health is why most API providers look to a multi-region approach. Having all of these building blocks in place, and being used to understand the health of the platform is essential to crafting and evolving your multi-region deployment strategy.

The learnings from investment in this area, combined with the awareness generated from API management, as well as the legal and governance areas will shape your multi-regional approach--giving you the incentives you need to justify the deployment, management, and support of APIs in multiple regions, zones, and countries.

Outreach and Support

The overall health of an API platform will define the friction and scope of your API outreach and support. The more problems you have, the greater your support costs will be. The more problems you have, the more friction you'll have when reaching out to new and existing users. There are a handful of outreach and support related building blocks that will play a role in managing your multi-region API deployments--ensuring you have the resources allocated to meet the demands of each region you are operating in.

- **Support** - Providing self-service and direct support channels for each individual region.
- **Communication** - Have a communication plan in place for each individual region.
- **Dashboard** - Consider providing regional specific dashboards for localized users.
- **Reporting** - Producing reports on usage, and other relevant data for each region.
- **Road Map** - Break down any specific road map items for each individual region.
- **Issues** - Listing open issues for each individual region for API consumers.
- **Change Log** - Providing an up to date change log for each individual region.

When thinking of scaling API infrastructure to a multi-region scope, it is critical to consider what the outreach and support impacts will be. Make sure you are properly tailoring your outreach and support efforts to match the needs of the local audience, otherwise they might fall through the cracks of your operations. A lack of support, and not investing in a feedback loop with customers is the number one reason API programs stumble--when scaling your APIs to new markets you won't get a second chance to make a first impression.

Break Existing Consumers Down By Region

The place to begin with any multi-regional API deployment effort is to break down the regions in which your existing API consumers operate in. This data should be available to you through your API management infrastructure, allowing you to group consumers by region, and even understand with regions help you generate the most value or revenue to your platform. This is where you should be starting when it comes to crafting your multi-region API strategy.

Regional expansion should be rooted in research, and established needs. It might be to expand a reach into where your existing consumers exist, or it may be more about keeping the consumers you already have--meeting new regulatory requirements in a specific country.

Whatever the reason, make sure you have developed a sufficient amount of awareness around your existing regional consumption of API resources, and make API management the first piece of infrastructure you consider evolving to match your need when delivering multi-region API resources that will scale to meet your existing consumer needs, as well as those of your future customers just around the corner.

Awareness Of API Performance Across Regions

Once you have a handle on where your API consumers operate, get to work measuring the availability and performance of your APIs from within those regions. Setup monitoring, testing, and performance services within any of the regions you identified as having a significant amount of consumers operating in. Gather data in real-time for over a month, and develop a solid understanding of the quality of your service your largest groups of customers experience.

Once you have put yourself into your consumers shoes, and have quantified the level of service they experience, you can begin weighing which customers are the most valuable to your business, and would respond, or even pay for a higher quality of service via APIs deployed within a specific region.

Laying the groundwork for your multi-region API effort, identifying the existing consumers that would benefit from the work, but also the ones that will benefit most from improved performance, or possibly just worth keeping by becoming compliant with local and regional laws. Regardless of where you choose to expand, make sure it is a data-driven decision and not just guesswork based upon the countries you think are cool.



Starting Out Small With A Single Region

Once you do figure out which regions you want to expand to, and have a list of logical reasons why you should be going multi-regional, I recommend taking just a single high value region, and begin there. Pick the most relevant, valuable, and achievable region from your list, and get to work developing a plan for scaling your infrastructure via the most relevant cloud platform, and corresponding region. Have a firm grasp on the first place you should be operating, and get to work defining what you will need to make it real.

Begin here. Get your feet wet. Understand what it takes to do a full rollout of your API in a second region. Operate it for a couple weeks or months, until you feel like you have a handle on things, then document your approach well, step back and analyze how well you did. Once satisfied, pick the second most important region on your list and repeat your establish approach.

Repeat this process as necessary, and methodically plot out your road-map for which regions are on the horizon. Eventually you will become skilled at managing multiple regions, and supporting your growing community that will come along with this expansion. Helping you establish the multi-region foothold you are confident will help take your business to the next level.

Thinking Beyond A Single API Region

I've met many companies who still operate within the US-EAST-1 region in AWS, because it was their entry point into the cloud. Never having considered scaling into other regions, and understanding the impact on their business.

The purpose of this guide is to help these folks, as well as other growing API providers to think beyond just a single region, and feel confident that they can operate their API infrastructure in multiple regions, or even multiple cloud providers around the globe.



