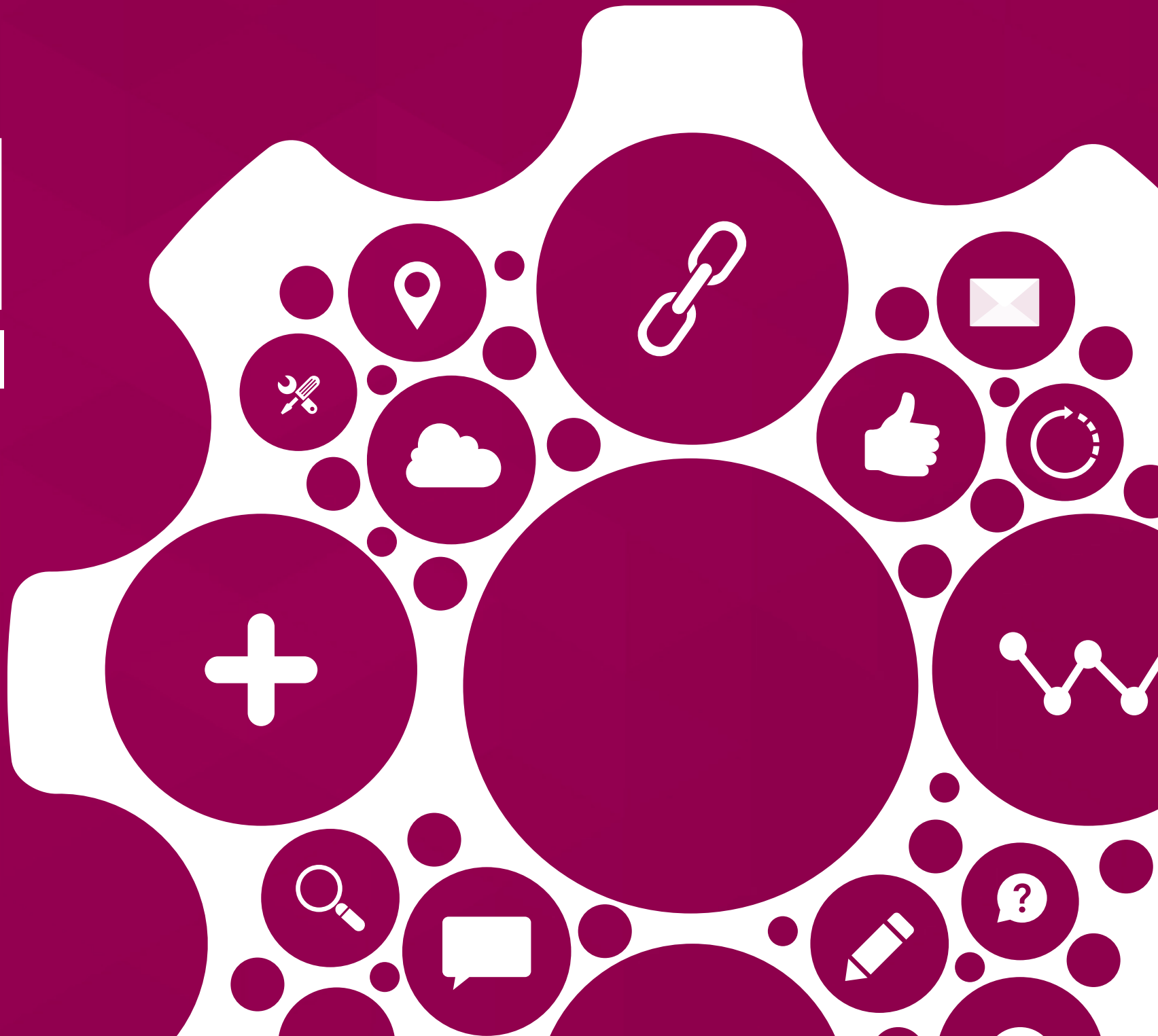


# THE STATE OF API INTEGRATION

REPORT

2017





**Happy Integrating.**

Mark Geene  
CEO - Cloud Elements  
[@mgeene](#)



# THERE HAS BEEN AN EXPONENTIAL PROLIFERATION OF APIs THROUGHOUT THE YEARS.

Our vision at Cloud Elements is to make it faster and easier for developers to connect and work with any API. Each new service or application is an island of data, causing developers to be overrun with new APIs, resources and information. And with each new API, developers are challenged with determining how it works and what value writing to the API brings.

This report helps to address the proliferation of APIs by providing trends, insights on ease of integration, data on where the industry is strong, and where it is going next.

## DATA COLLECTION METHODOLOGY

The data presented here had been collected and analyzed between September 2016 to March 2017, and comes from Cloud Elements API Integration platform. Over 107 public endpoints, 58 beta endpoints, 28,000 individual instances and 1.6 Billion+ API Calls were surveyed to extract the following results. In addition to the Cloud Elements platform, complementary research provided by ProgrammableWeb API Directory 'Growth in Web APIs Since 2005', the Datanyze Market Share Report of 2016, and SmartBear State of API Report 2016. Industry expert contributors include **Ross Garrett, Sr. Director Product Marketing at Cloud Elements and Kin Lane, The API Evangelist**. It will help all developers navigate the recent explosion of APIs and the implications of API integrations to work more efficiently in 2017 and beyond.

**We'll cover:**

- 1.** The Growth of API Integration
- 2.** API Integrations | Compare and Contrast
- 3.** Current Scorecard of API Integration

# MEET THE CONTRIBUTORS



**MARK** **GEENE**

CEO At Cloud Elements

Mark Geene, co-founder & CEO of Cloud Elements, is recognized as a leading visionary for unifying the

world's APIs by defining the first API Integration platform. This revolution of API integration is the next frontier in API Management as developers address the challenge of consuming millions of disparate private and public APIs. In addition, to being a pioneer in the API world, Mark is a well-recognized leader in applying the Lean Startup methodology and has earned awards from Outside Magazine, Apex, Stevie, and Denver Gazelle for his outstanding work.



**ROSS** **GARRETT**

At Cloud Elements

Ross Garrett is the Head of Product Marketing at Cloud Elements - responsible for market strategy, product

positioning and evangelism. He is a well-known speaker at developer events and other industry conferences. Ross has over 10 years of product and marketing leadership experience in the integration space, most recently at Push Technology and previously with Axway, CA and Layer 7.



**KIN** **LANE**

API Evangelist

I am the API Evangelist. Not in the sense that I'm evangelizing a single API to you--In the sense that APIs are important

for everyone to be aware of. I'm paying attention to not just the technical, but the business and politics of the web API movement. I share my insights by blogging on the business of APIs at [apievangelist.com](http://apievangelist.com), politics of APIs at [apivoice.com](http://apivoice.com) and you can find more information about me at [kinlane.com](http://kinlane.com).

TABLE OF CONTENTS

WORD FROM OUR CEO  
MEET THE CONTRIBUTORS

01  
INTRODUCTION

- 06 The State of API Integration  
Important Findings
- 07 A Look Ahead

02  
THE GROWTH OF APIs

- 09 SOAP Isn't Dead Yet
- 10 Proliferation of Public APIs
- 11 Demand of Cloud Services by Category
- 12 Growing Demand of Emerging Category Leaders

03  
COMPARE AND CONTRAST

- 20 Authentication
- 22 Metadata Discovey
- 24 Bulk
- 26 Events and Synchronization
- 27 Webhooks

04  
API CALCULATOR

- 31 CRM Endpoint Scorecards

05  
CLOSING

- 36 About Cloud Elements

CHAPTER 1

# INTRODUCTION



## INTRODUCTION

# THE STATE OF API INTEGRATION

API integration is one of the most critical aspects of application development today. It's not enough that your application exposes a great, developer friendly API, you often need to integrate the APIs exposed by other products and services - allowing these differing interfaces to work well together is fundamental to a reliable application experience for your users. A good example is data synchronization - where business events need to be communicated between products and services in a uniform way, even though each API offers a different integration model or data structure.

This means developers must navigate a sometimes-bewildering set of features and variations across different APIs, and figure out a way to consolidate them into one coherent, streamlined data model. With so many different products and services out there - each with their own API—this API integration report aims to highlight the features and typical integrations you should design for within your applications and APIs.



## IMPORTANT FINDINGS

This report takes a look back at the state of API integration in 2016, while providing guidance for the future. One area that keeps coming up year after year is the balance of power between SOAP and REST. There is no doubt that REST dominates today, but there's still a remarkable percentage of SOAP APIs out there that can't be ignored - at least 15% based on our experience. The report will also take a snapshot across API consumption in various categories over the past year - looking at this data from an enterprise integration perspective shows not all APIs available are used equally. The usage across these categories provides key information about the importance and potential growth of different sectors of the market.



...remarkable percentage of SOAP APIs out there that can't be ignored - at least 15% based on our experience.

The report will also examine specifications across different APIs, and how design decisions made by API product managers can have a significant impact on how easy it is for you to consume data from these products and services. For example, Webhooks is still a niche feature amongst the most commonly used APIs, despite the fact that over 80 percent of developers indicate they prefer using Webhooks over comparable methods. The good news is that marketing, document storage and e-commerce services are leading the way here, encouraging others to follow this trend.

---

## INTRODUCTION

---



### A LOOK AHEAD

This report also gives us an opportunity to look forward to 2017 and the trends that are set to affect APIs and application integration. A trend that we've come to expect over the past few years is around public APIs - they've become prolific amongst products and services wanting reach the "long-tail" of application developers. For 2017, all the data suggests we'll see another year of record growth in this segment.

Within enterprise integration, a need to bridge application data from similar types of products (E.g. CRM) is driving significant growth. This is occurring both within and without corporate IT - which is impacting the requirements placed on integration technologies and even the APIs themselves. For example, we see increased API consumption and data transformation for services in the CRM, marketing automation, cloud storage and e-commerce segments. And a growing need to simplify, speedup and automate common integration workflows - such as creating new CRM records for a purchases from new customers.

Of course API technologies aren't standing still either. We've mentioned developer preference for Webhooks to handle asynchronous eventing between application services, and this is gaining a place within popular API documentation languages such as OpenAPI Specification v3.0. Hypermedia is also given a slight nod within OAS v3.0, with some basic linking capabilities being described. With adoption of this new specification set to takeoff during 2017, we can perhaps expect to see a greater understanding of (and appreciation for) hypermedia flows among API product managers.

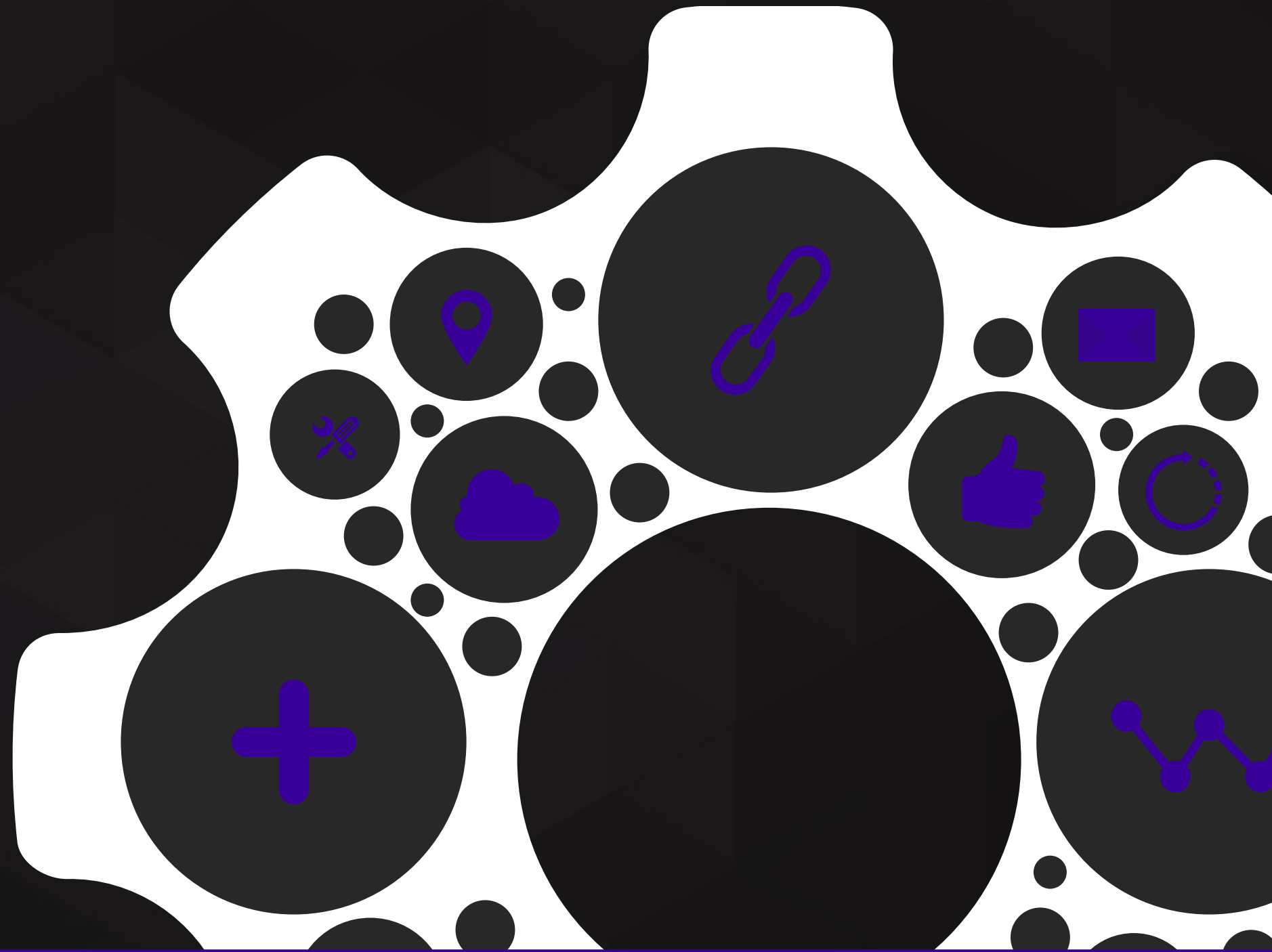


---

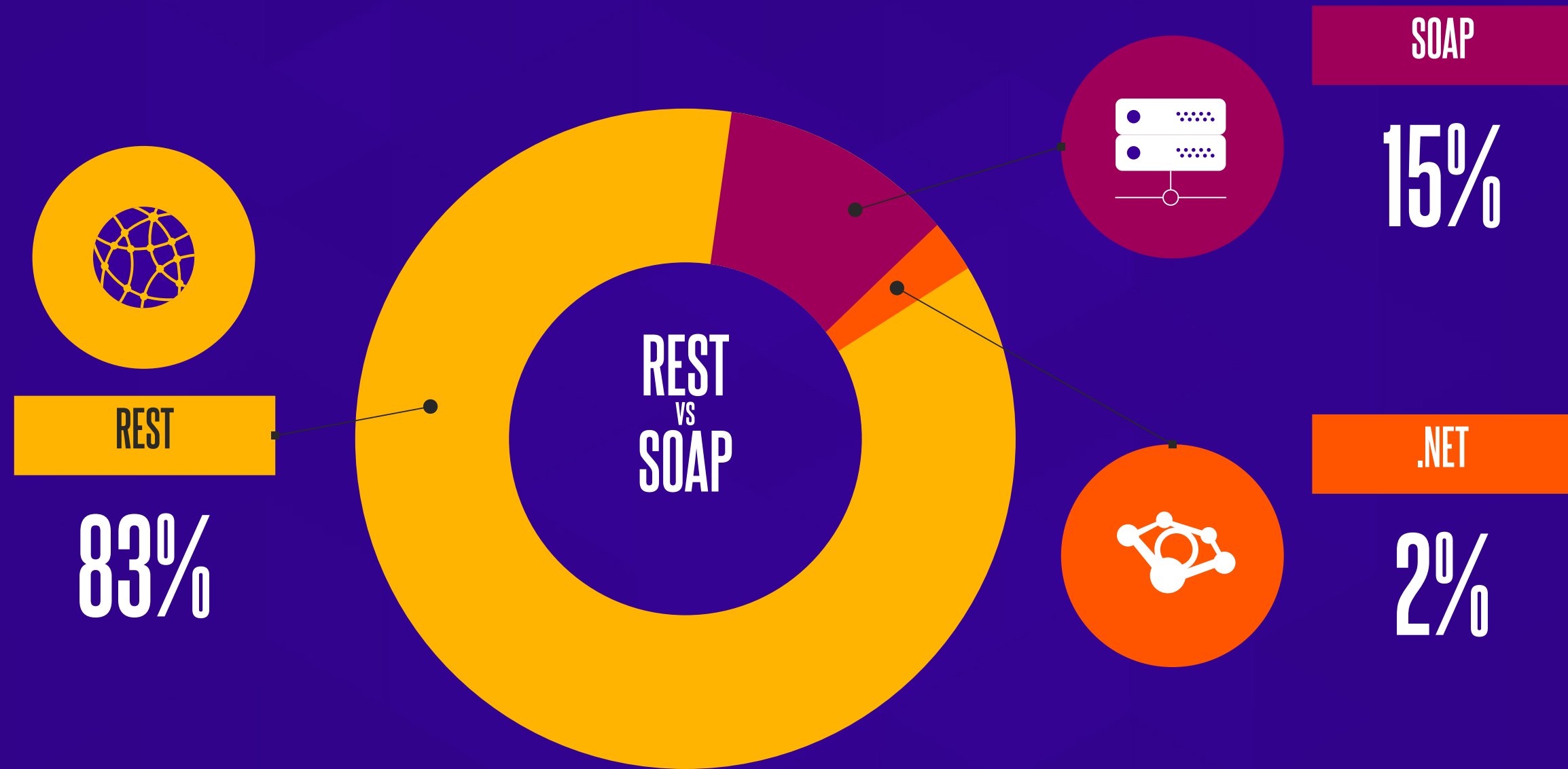
## CHAPTER 2

---

# GROWTH OF APIs



# SOAP ISN'T DEAD YET



## GROWTH OF APIs

# PROLIFERATION OF PUBLIC APIs



The proliferation of APIs is a leading indicator for the demand of API Integration. According to ProgrammableWeb, in a matter of 6 years, web API counts increased **758% from 2010 - 2016**. With API Services proliferating at an accelerating pace, new and interesting challenges for developers continue to arise. Each API is unique, has its own resource definitions, data model schema, error handling, paging structure and more.

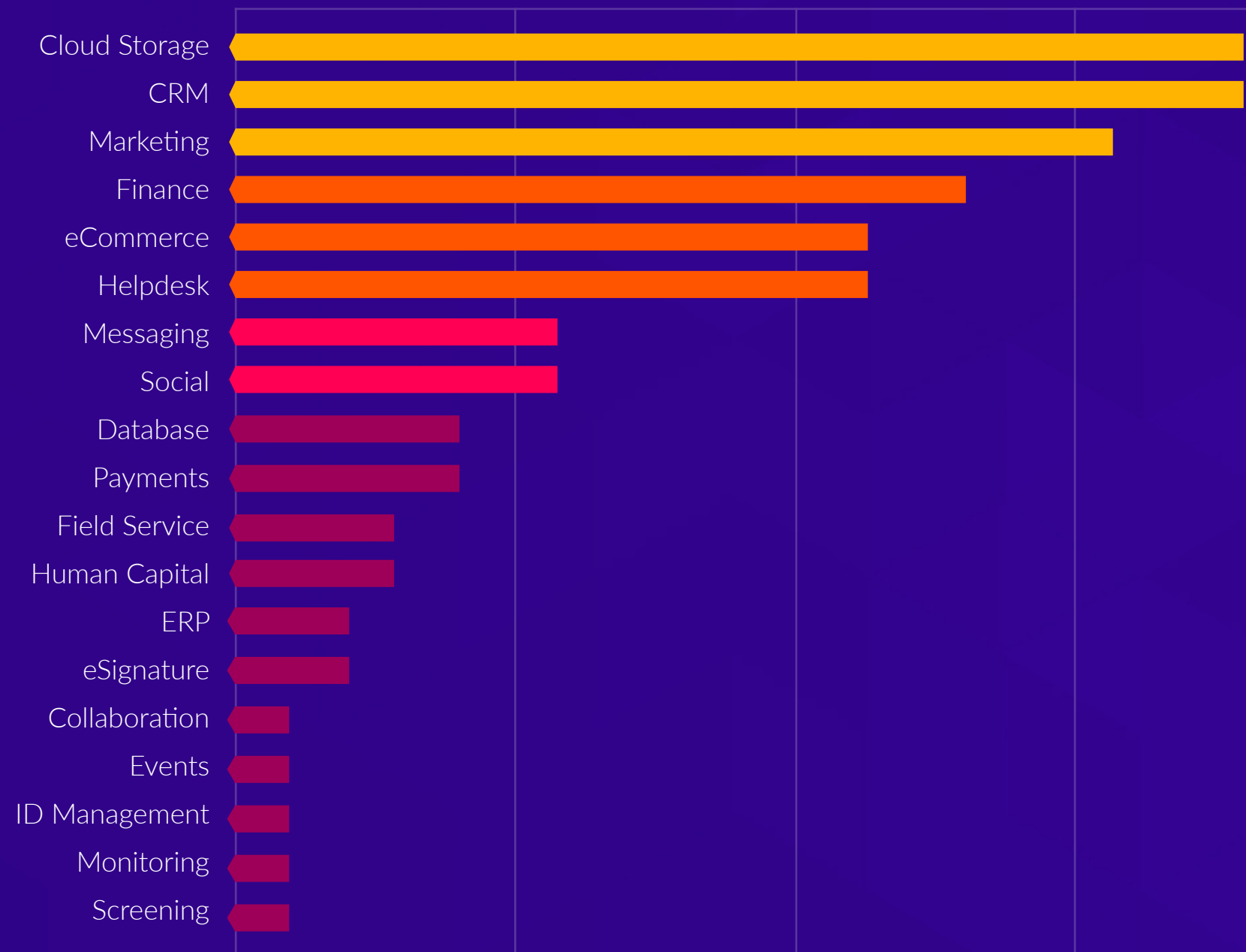
API Management vendors have conditioned us to believe that our APIs are living on their own little island, and we must self-develop and publish each in order to monetize the results for company's sake. In reality, apps are increasingly combining your services with APIs purchased from vendors.

Your APIs, in combination with the APIs of SaaS, PaaS, IaaS and IoT services, are building the next generation of Web, SaaS, Mobile and Internet of Things (IoT). These new composite apps help to automate workflows among multiple services, based on events and triggers. Citizen Integrators, comprised of customers, partners and employees, want apps to work with the other cloud services they use without having to spend hours learning your unique API, and composite APIs make this happen.



**Source:** <https://www.programmableweb.com/news/programmableweb-api-directory-eclipses-17000-api-economy-continues-surge/analysis/2017/03/13>

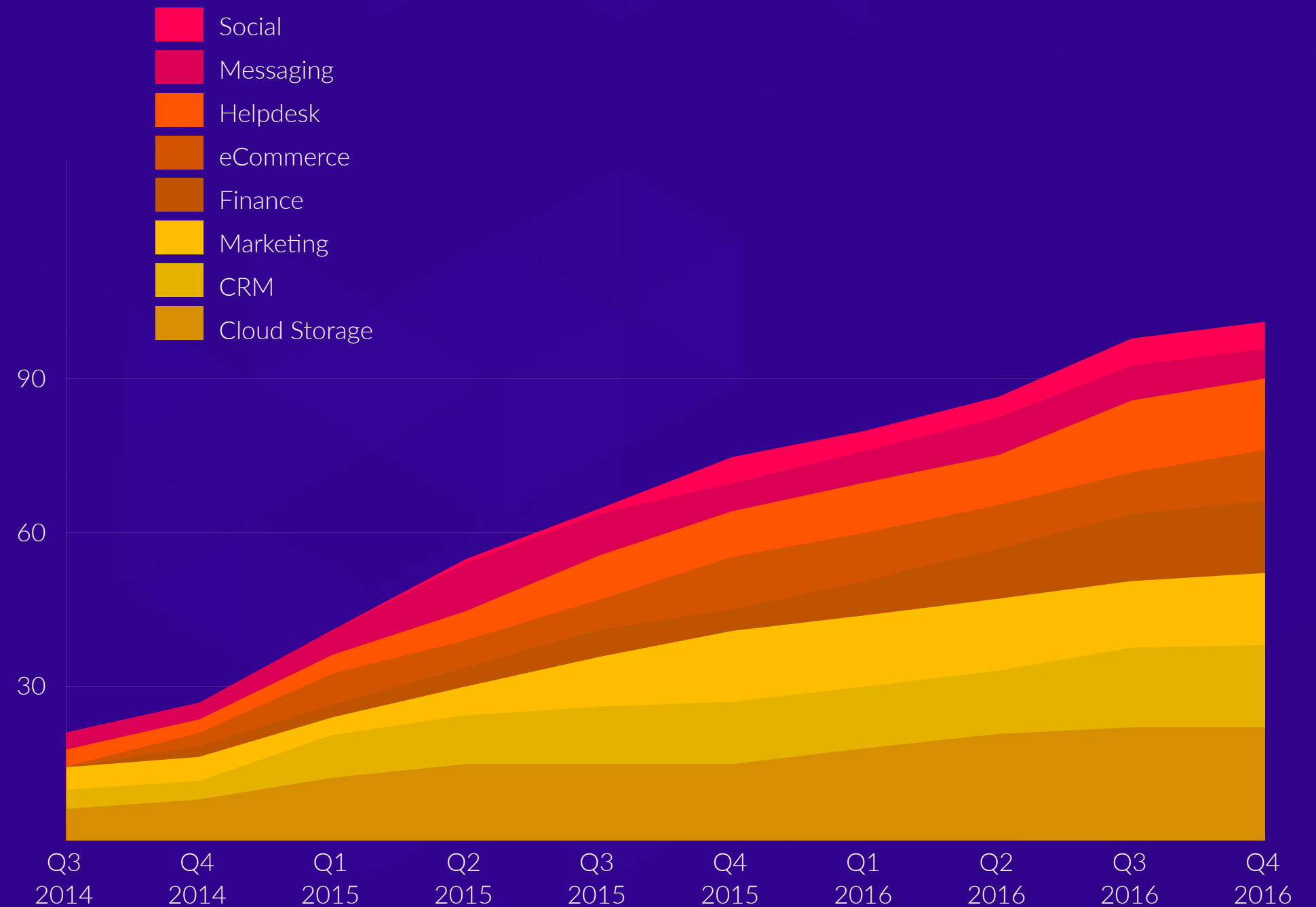
# DEMAND OF CLOUD SERVICES BY CATEGORY



As the demand for integration between cloud categories increases, the need for accompanying APIs spanning across all categories of cloud service continues to boom. While the majority of the demand for APIs fall under the Cloud Storage category, in mid 2015, the industry saw a spike in demand for CRM API use, creating the need for APIs across several cloud service categories.

# GROWING DEMAND OF EMERGING CATEGORY LEADERS

A growing demand for integration can be measured by emerging technologies across multiple categories. Most notably we have seen dramatic growth in demand for finance applications and helpdesk applications in the latter half of 2016.

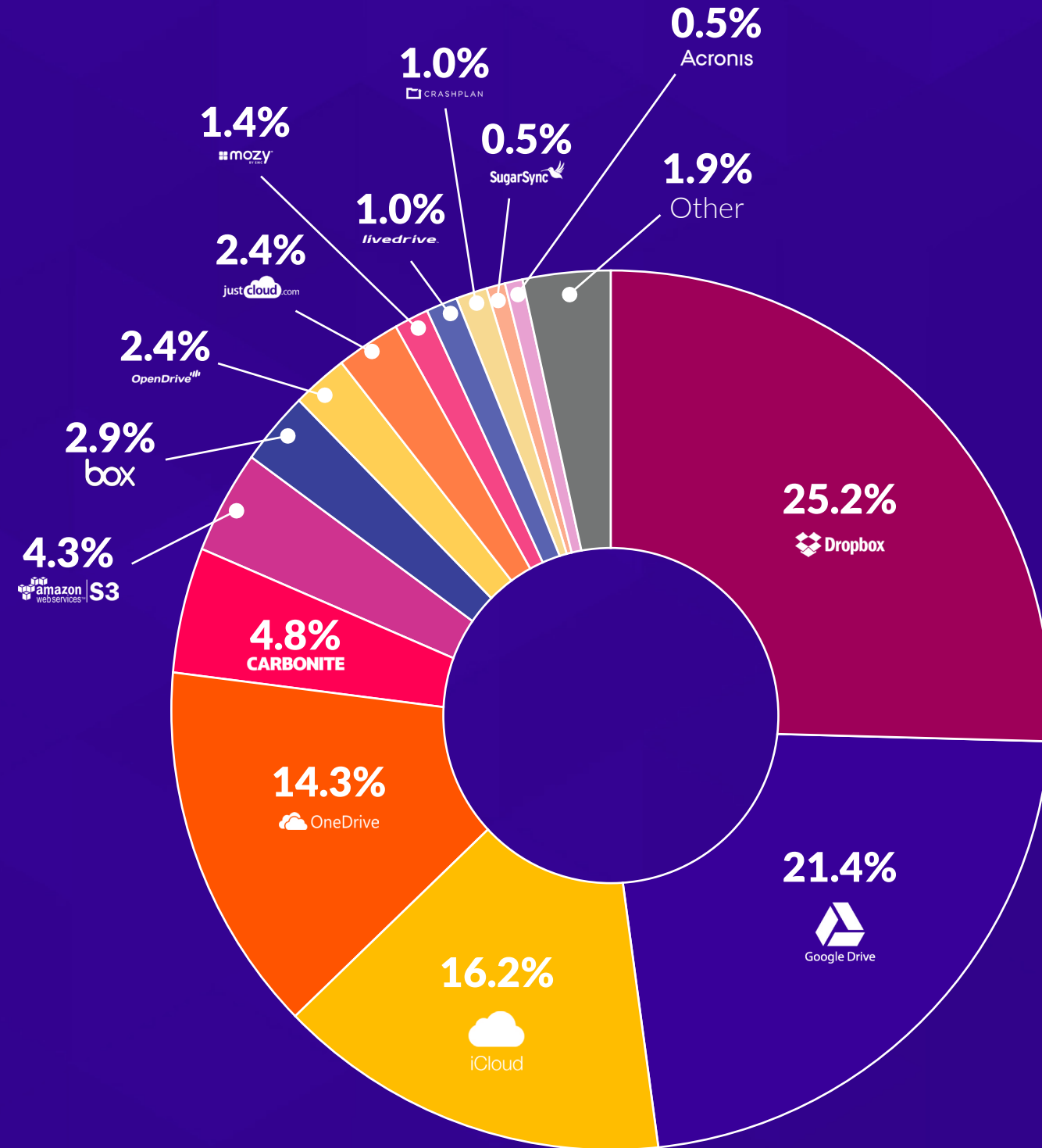


# CLOUD STORAGE

## CATEGORY MARKET LEADERS

The percentage to the right represents the Cloud Storage and Document market share generated by Datanyze. It is calculated by the number of websites using given technology divided by the total number of websites using any of the technologies selected in the left column; websites limited to the Alexa top 1M.

Integration roadmaps are often prioritized based on customer demand for integrations to current software market leaders. A critical aspect of integration design is to consider who the current market leader is the space and ensure your application is futureproof for any new entrants to the market that could soon takeover.

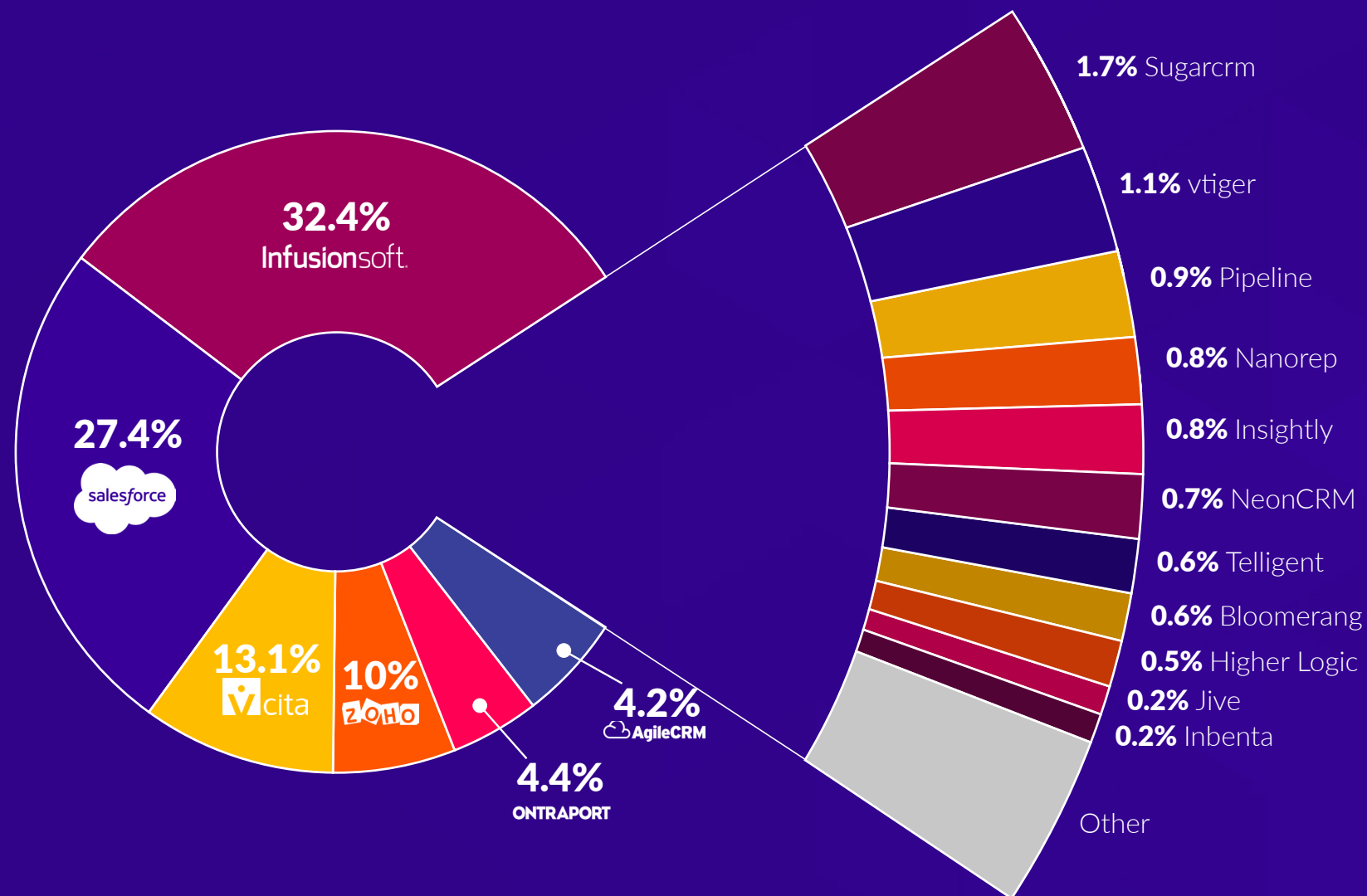


# CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

## CATEGORY MARKET LEADERS

The percentage to the left represents the CRM market share generated by Datanyze. It is calculated by the number of websites using given technology divided by the total number of websites using any of the technologies selected in the left column; websites limited to the Alexa top 1M.

Integration roadmaps are often prioritized based on customer demand for integrations to current software market leaders. A critical aspect of integration design is to consider who the current market leader is within the space and ensure your application is futureproof for any new entrants to the market that could soon takeover.

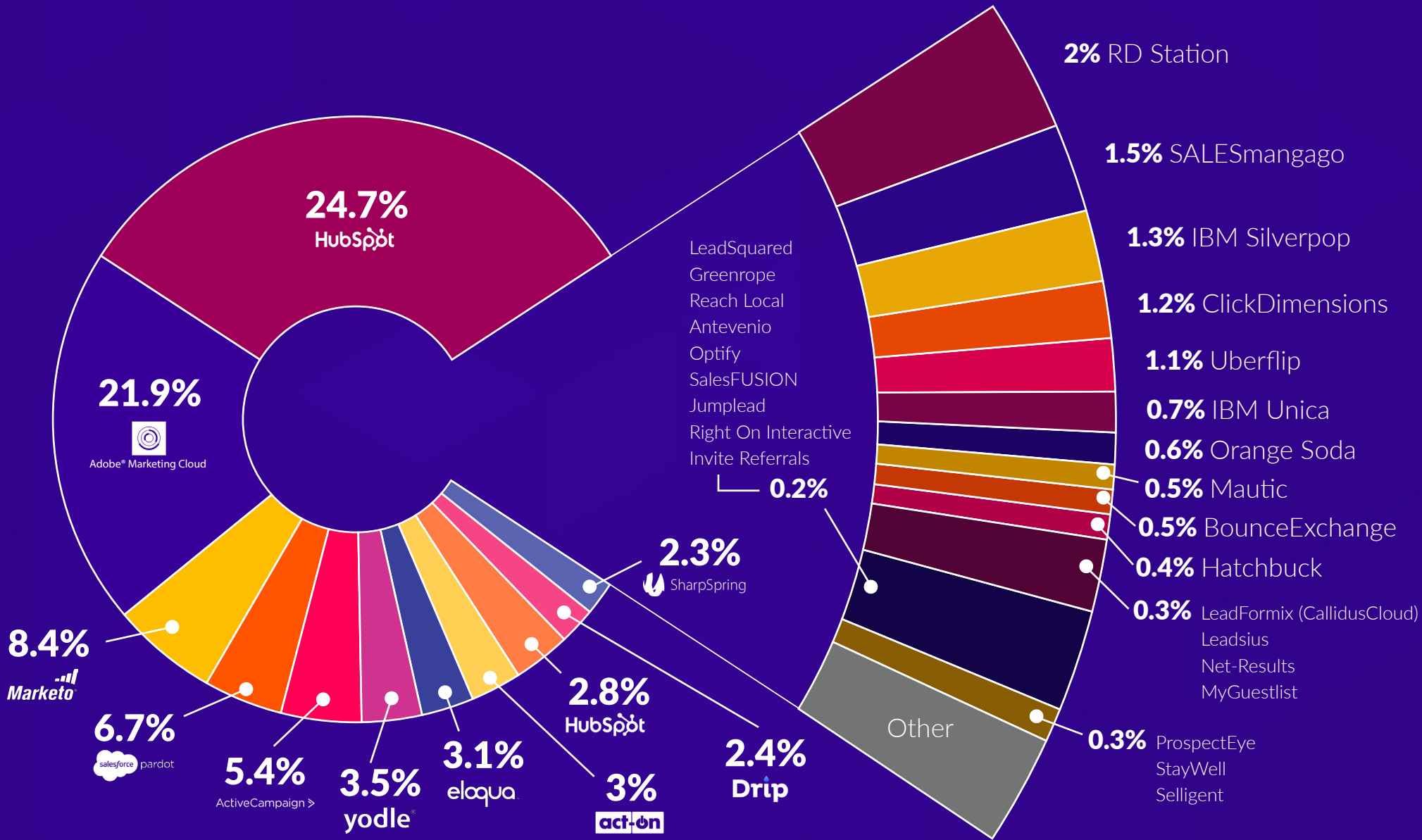


# MARKETING AUTOMATION

## CATEGORY MARKET LEADERS

The percentage to the right represents the Marketing market share generated by Datanyze. It is calculated by the number of websites using given technology divided by the total number of websites using any of the technologies selected in the left column; websites limited to the Alexa top 1M.

Integration roadmaps are often prioritized based on customer demand for integrations to current software market leaders. A critical aspect of integration design is to consider who the current market leader is within the space and ensure your application is futureproof for any new entrants to the market that could soon takeover.

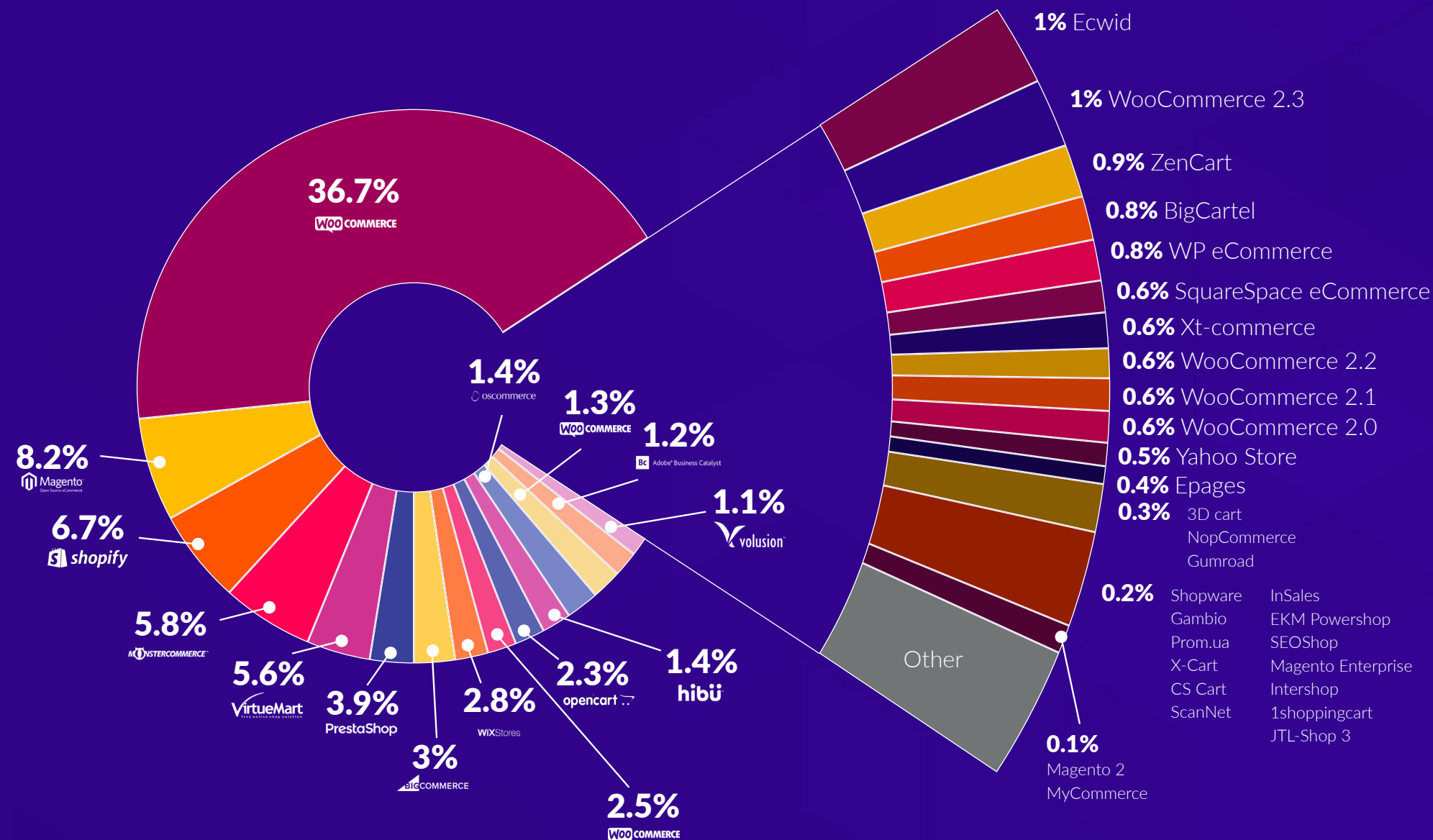


# FINANCE & ECOMMERCE

## CATEGORY MARKET LEADERS

The percentage to the left represents the finance market share generated by Datanyze. It is calculated by the number of websites using given technology divided by the total number of websites using any of the technologies selected in the left column; websites limited to the Alexa top 1M.

Integration roadmaps are often prioritized based on customer demand for integrations to current software market leaders. A critical aspect of integration design is to consider who the current market leader is within the space and ensure your application is futureproof for any new entrants to the market that could soon takeover.



# API INTEGRATIONS COMPARE AND CONTRAST



---

A NOTE FROM  
THE EXPERT

---



**Kin Lane**  
API Evangelist

“

Cloud Elements has their finger on the pulse of the API sector in their report, highlighting the API growth in the areas that matter most to businesses today, but also by identifying the biggest challenges we face in ensuring APIs continue to drive the same growth we've enjoyed over the last decade. This is what I do as the API Evangelist, focus on what has been driving the growth, but highlight the deficiencies across the sector that might be holding things back, and in the way of seeing the scale we expect of the API economy.

With the growth in the number of public APIs, and resulting amount of valuable digital resources being made available online, standardizing how we secure access to these resources is growing more important each day. Identity, authentication and access management will continue to be the biggest challenge we all face, and be the number one area that companies will need to be investing in they expect to achieve success in their API journey.

Discovery is another area of the API lifecycle that will continue to plague API providers, consumers, and the businesses and people they are serving in coming years. With the growing number of both public and private APIs, we need to be investing in crafting, storing, and sharing of metadata describing our APIs, as well as the operations surrounding them--without discovery, we won't continue to see the growth we've enjoyed in the space over the last decade.

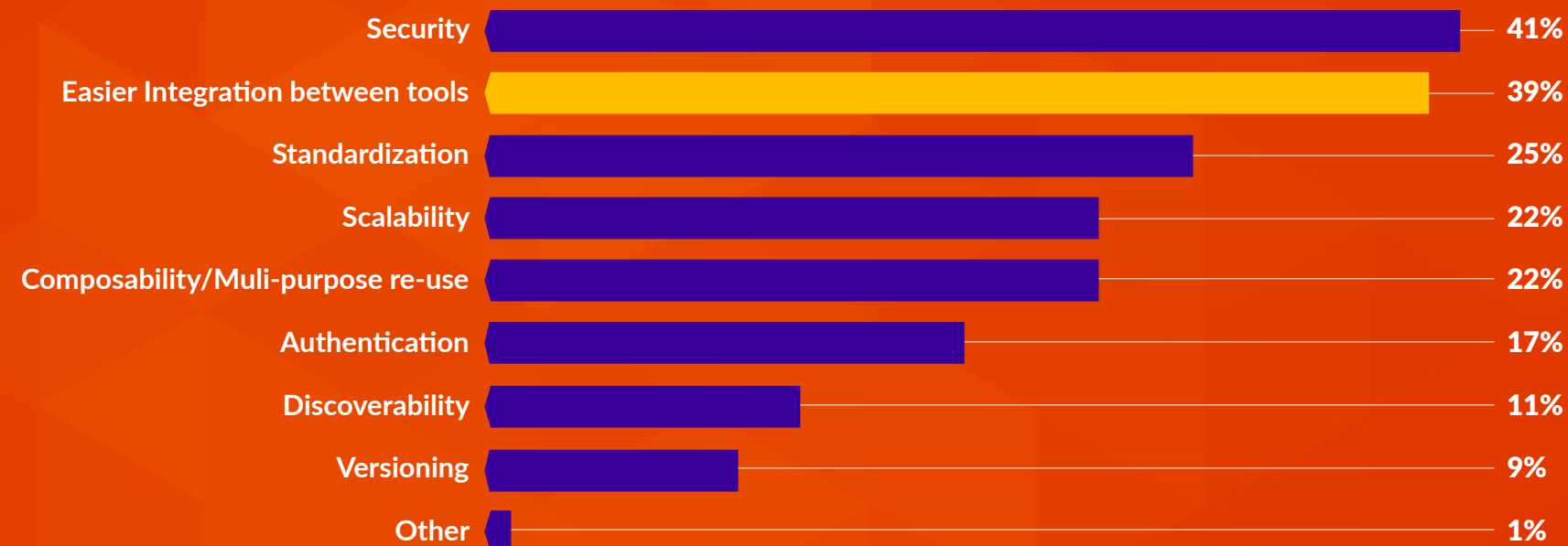
As this State of API Integrations Report highlights, the strong areas of growth have been in the most critical areas of doing business online today. The deployment and availability and the management of these resources is just the beginning. We need to continue orchestrating, synchronizing, and ensuring these resources are where we need them, when we need, them, across the growing number of cloud platforms we are depending on to make our businesses continue to operate in today's fast-paced digital environment.

Cloud Elements' State of API Integrations Report provides us with a good look at the growth in the API space, but I support it because it gives an honest look at the hard work we still have ahead of us. It can be easy to get excited about the growth we've made in the last decade with new areas like cloud computing, social, and critical business areas like storage, CRM, marketing, and finance, but the real opportunities lie ahead once we invest more in authentication, discovery, bulk operations, and making sure our resources are where they need to be, when it matters to our business.

# THE BIGGEST CHALLENGES IN API TECHNOLOGY THAT NEED TO BE SOLVED...

API integration technology, best practices, and lessons learned have come a long way, but there are still many obstacles that stand in the way of developers and organizations getting the most out of the API Economy, or areas where API providers can improve the developer experience.

From the research provided by SmartBear in their State of APIs Report 2016, **thirty-nine percent of enterprises want easier integration between the tools they use**, and they wanted it yesterday. These companies have more than 1,031 internal cloud services, so getting every app to work together is a major need.



## API technologies also continue to evolve -

both in terms of standards and best practice. For example, a move toward event-driven integration models has demonstrated a surge in the support for and use of Webhooks. Just publishing or writing to an API is only **the tip of the iceberg** in terms of integration.

Developers all too often are left to deal with various aspects of API complexity themselves, such as **authentication and authorization**, metadata **discovery**, **bulk** support, and **event** synchronization.

By addressing these four challenges that cause friction around API integration, enterprises, app providers and developers can take advantage of the efficiency and opportunity offered by a seamless, fully connected application ecosystem.



# AUTHENTICATION IS THE KEY

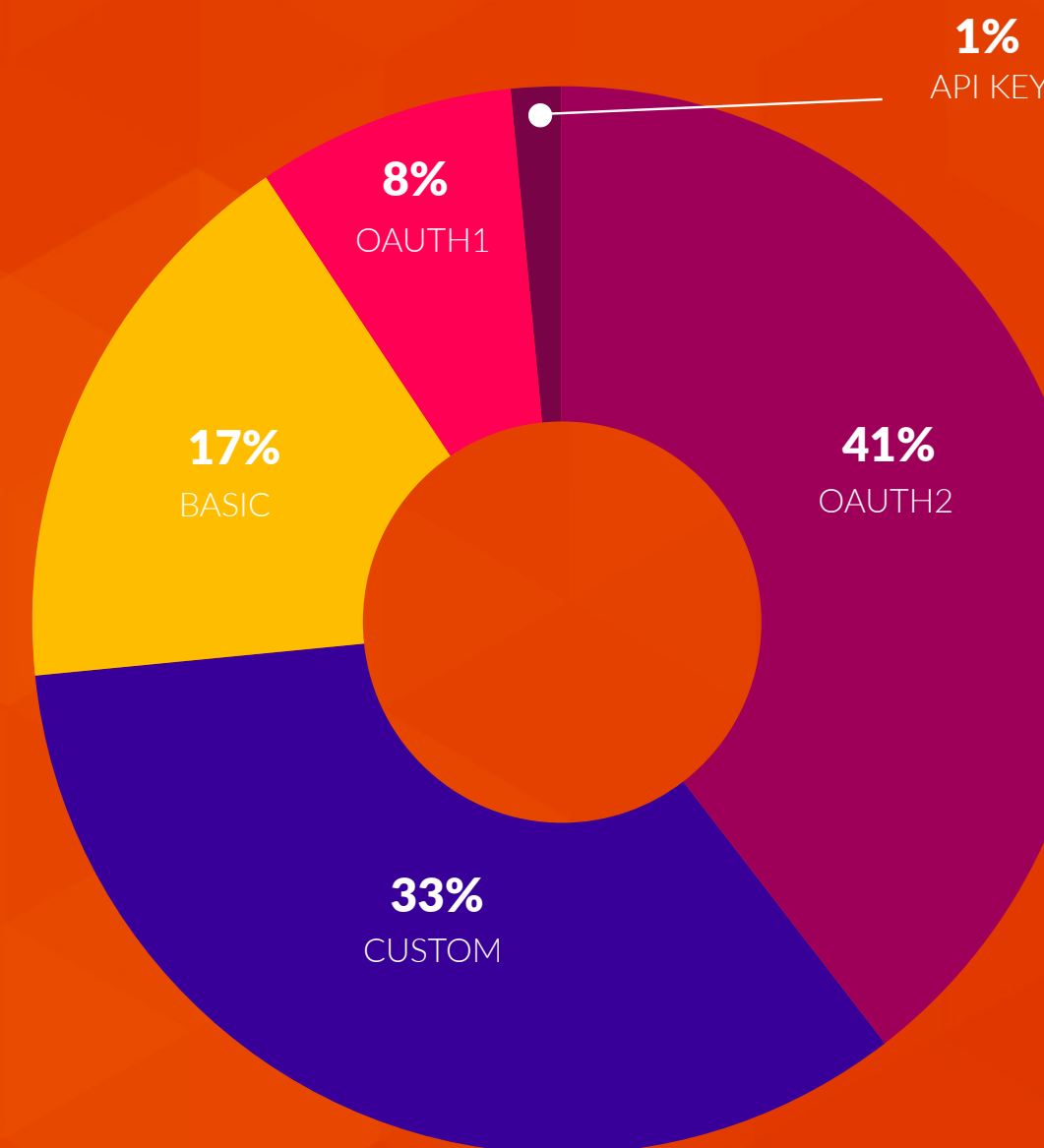
Authentication and authorization are fundamental to exposing data and services via APIs, providing not only simple authentication, but also identity based access control. This lets the API provider know which client is connecting and what features or data they have access to.

## TYPES OF AUTH

With so much concern today around data security, many application providers are moving away from simple single-factor authentication methods and requiring an additional step to verify the identity of the client. This often means obtaining and submitting an additional code after the simple username and password have been accepted.

Currently the market is moving quickly to multi-factor and delegated auth mechanisms based on OAuth, but there are still many APIs out there today relying on Basic Auth (17 percent), API Keys (1 percent) or some custom implementation of API Key & Secret (33 percent).

Looking more closely at OAuth usage, our statistics show that the unloved and complicated OAuth 1.0 (and 1.0a) is used by only 8 percent of providers today compared with the more ubiquitous OAuth 2.0 at over 41 percent.



# SUPPORT FOR OAUTH

The preference for OAuth is well placed. This mechanism is token-based and far more secure than relying only on usernames and passwords which can be easily shared or spoofed. Access tokens can be time-bounded, or have other criteria (such as location) taken into account, meaning access can be easily revoked or challenged if such criteria are not met.

In addition, OAuth 2.0 allows the provider to authorize access to specific API functionality or data through the use of a feature called “scope”. Simply put, an access token is granted the scope to perform certain actions. For example, some clients may only be granted read-only access to data via the API, and this limitation of scope will be associated with their access token. Whereas a different client may be granted write access also, again configured via token they have been granted.

The level of flexibility offered by OAuth 2.0 makes it the ideal choice for authentication and authorization of clients.



## CLOUD ELEMENTS SUPPORT FOR AUTHENTICATION

Cloud Elements has normalized authentication for each type of the authentication used by the endpoint (e.g., Basic, OAuth 1.0, OAuth 2.0, WS-Security, API Key, etc). There are inherent authentication workflow variations depending on the type of authentication (e.g., basic credentials vs. OAuth 2.0) but the authentication types are implemented consistently. Cloud Elements uses a normalized API token based approach with a user and organization secret, combined with an Element token associated with the specific authenticated account of the Element (known as an “Element Instance” (see [term definitions](#)). Switching endpoints within a Hub is as simple as swapping out a single element token.

# METADATA DISCOVERY

With such proliferation of applications and services used in enterprise or application development and the sheer number of APIs available today, developers must carry the burden of understanding the details of each integration. This is becoming exponentially more time consuming.

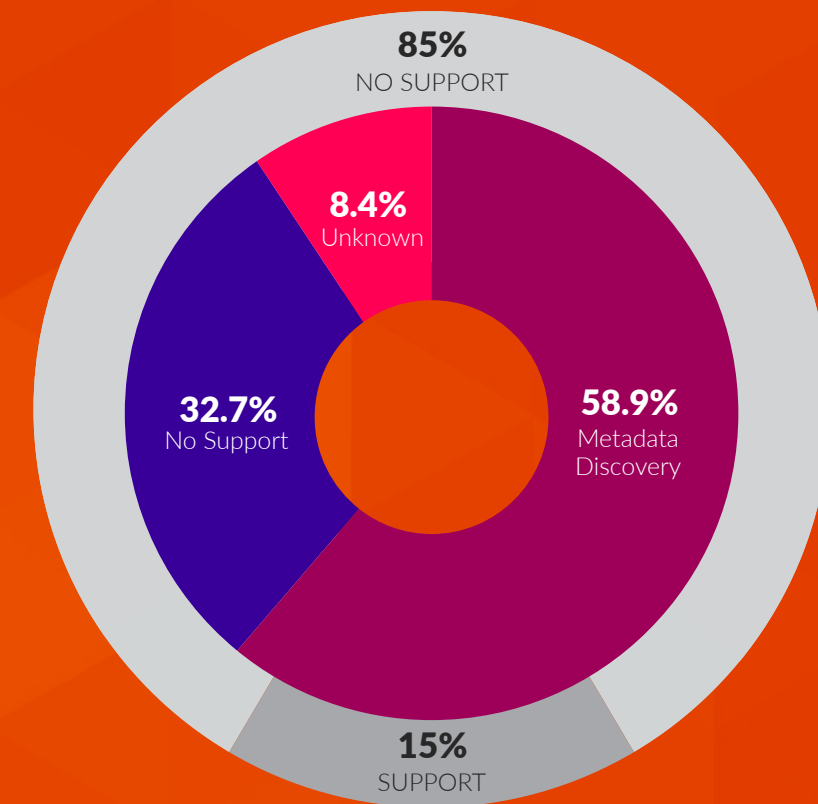
In addition, many of the applications in use offer the ability to customize or add to the data they store for a given entity. For example, in Salesforce it's possible to customize the fields associated with customer account, contact, or lead, etc.

To help relieve some of the pain, API providers are increasingly offering a capability known as metadata discovery, so that data models and resource structures can be accessed and understood programmatically. Taking the Salesforce example again, the metadata interface makes it possible to simply “ask” what the API can do and in what format a client should expect to receive responses and data objects. If a custom field is added in this case, the metadata can be used again to discover this addition.

## Native Support of Metadata Discovery

Percentage of API Endpoints that Support Metadata Discovery  
Provided by Cloud Elements

We assume across the industry <15% of APIs support metadata discovery.



**In our research we have found that 58% of endpoints support metadata discovery.**

One caveat to this research is that it's heavily weighted by the custom data integrations provided by Cloud Elements. We are bound to support more endpoints that natively support custom data, therefore more endpoints that natively support metadata discovery. While we have found that 58% of APIs support discovery, our sense is that industry-wide, this number is far less.

# METADATA DISCOVERY

Even though the number of API metadata interfaces are increasing, unfortunately the amount is still relatively small. From our statistics, we see application providers offering this most commonly in the CRM and Marketing Automation domains - likely because these platforms often need to handle custom data.

The challenge of metadata discovery shouldn't be confused with some of great work being done around API documentation languages - such as the Open API Initiative (formerly known as Swagger). While these efforts have greatly improved the developer experience and usability for many APIs, and enabled automated documentation and code-gen, ultimately, developers are still challenged with the task of reading and learning virtual reams of documentation in order to integrate with the APIs they need to use.

API product managers should consider how to enable metadata discovery for their APIs - even where the data model is static. Discovery interfaces provide the foundation for easy data manipulation - if the structure of a data object can be made available programmatically, it then becomes possible to transform data from one source to another. This transformation capability is fundamental to enterprise integration where a single system is no longer the sole system of record for much of our data.



## CLOUD ELEMENTS SUPPORT FOR METADATA DISCOVERY

**Discovery Service** - Cloud Elements includes a comprehensive data discovery service that provides normalized metadata, such as the list of field names and types. Additional information, if available from an endpoint, may also be obtained such as: display name, read-only, etc. If an endpoint doesn't provide discovery service APIs, Cloud Elements will still provide a minimum set of metadata about the given resource (e.g., name and type). Cloud Elements also allows you to discover custom fields (as long as the values are not empty), by supplying an object ID when a native discovery service is not available. The Discovery Service is used along with the Transformation Service to normalize the responses across endpoints.

# BULK OPERATIONS

Bulk Upload and Download of data is useful for many applications, and where available we see users are keen to leverage this functionality. Yet only **42 percent** of APIs that Cloud Elements works with actually support bulk. This is a surprising figure, as almost all data sets in enterprise SaaS applications are substantial and require some form of bulk operations. Most enterprises struggle with the amount of data they have in the cloud. With 58 percent of APIs not offering Bulk support, SaaS applications make it hard to migrate data into and out of their systems when less than half offer programmatic methods. This situation acts as a barrier for migration, or synchronization between any new applications that want to seamlessly work with existing systems of record.

In cases where bulk operations are not available, another factor developers must deal with is around API rate limits. While necessary protections to maintain availability and performance for all users, rate limits can make it near impossible to deal with large volumes of data. For example, a popular e-commerce app limits applications to two calls per second, with a “leaky bucket” mechanism allowing up to **40 calls in backlog**. This means your application must deal with bulk data at rate of 2TPS which might result in hours or even days of continuous API calls.

“

Enterprise users are often dealing with vast data lakes, that need to be synchronized, analyzed, migrated or even shared. Application providers must think carefully about the functionality they offer to handle this scale of data.

ROSS GARRETT

# BULK OPERATIONS

There is no doubt that bulk data operations represent a unique challenge, computing resources can be tied up for unpredictable amounts of time and data loading can require complex field mapping. These challenges are worth the effort however, as many users today will be looking for easy ways to migrate between products - it may seem counter-intuitive to offer your customers an easy way to move off your product, but not offering migration capabilities may prevent a customer from choosing your platform in the first place.



## CLOUD ELEMENTS SUPPORT FOR **BULK**

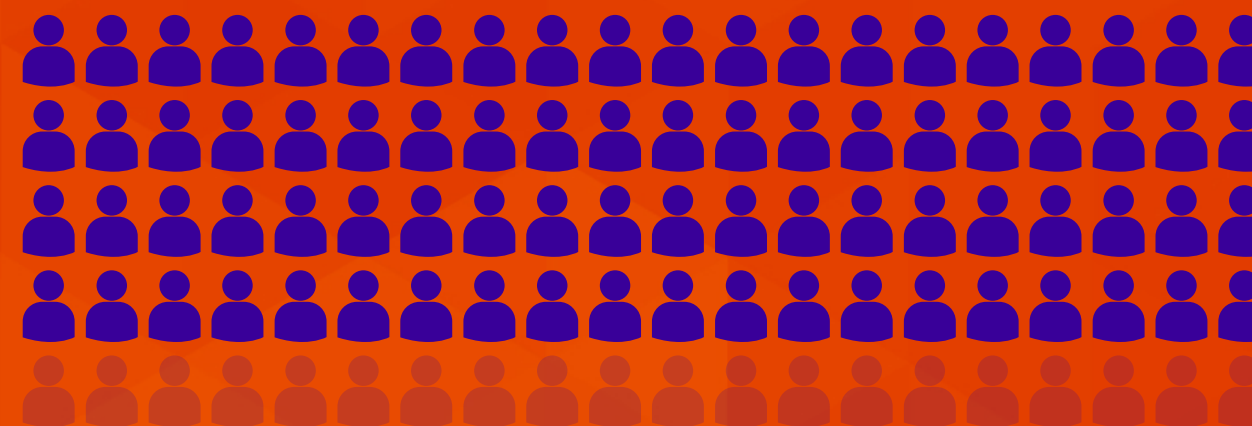
**Bulk** - Cloud Elements provides the ability to upload and download data in bulk from an endpoint in a normalized way. Cloud Elements will leverage the provider bulk endpoints whenever that is available. When there is no bulk available from the provider, Cloud Elements will provide a pseudo bulk service for uploading and downloading data from the endpoint. For uploads, we will accept a file and then create objects at the endpoint on a record by record basis. For downloads, we will execute a search API against the endpoint, and loop through all results until we have retrieved all the data. Cloud Elements will store these files (encrypted) in our platform for a maximum of three days. We're currently rolling out a Bulk service on an Element by Element basis.

*Let us know if you need Bulk APIs for a given Element by contacting us at [support@cloud-elements.com](mailto:support@cloud-elements.com).*

# EVENTS AND SYNCHRONIZATION

Business events are really the key piece of data that enterprises, applications and developers care about. In software terms, there are many ways of handling events - but webhooks is becoming the technology of choice amongst API providers and developers.

## DEVELOPER PREFERENCE



Eighty percent of developers prefer to use webhooks over other methods.

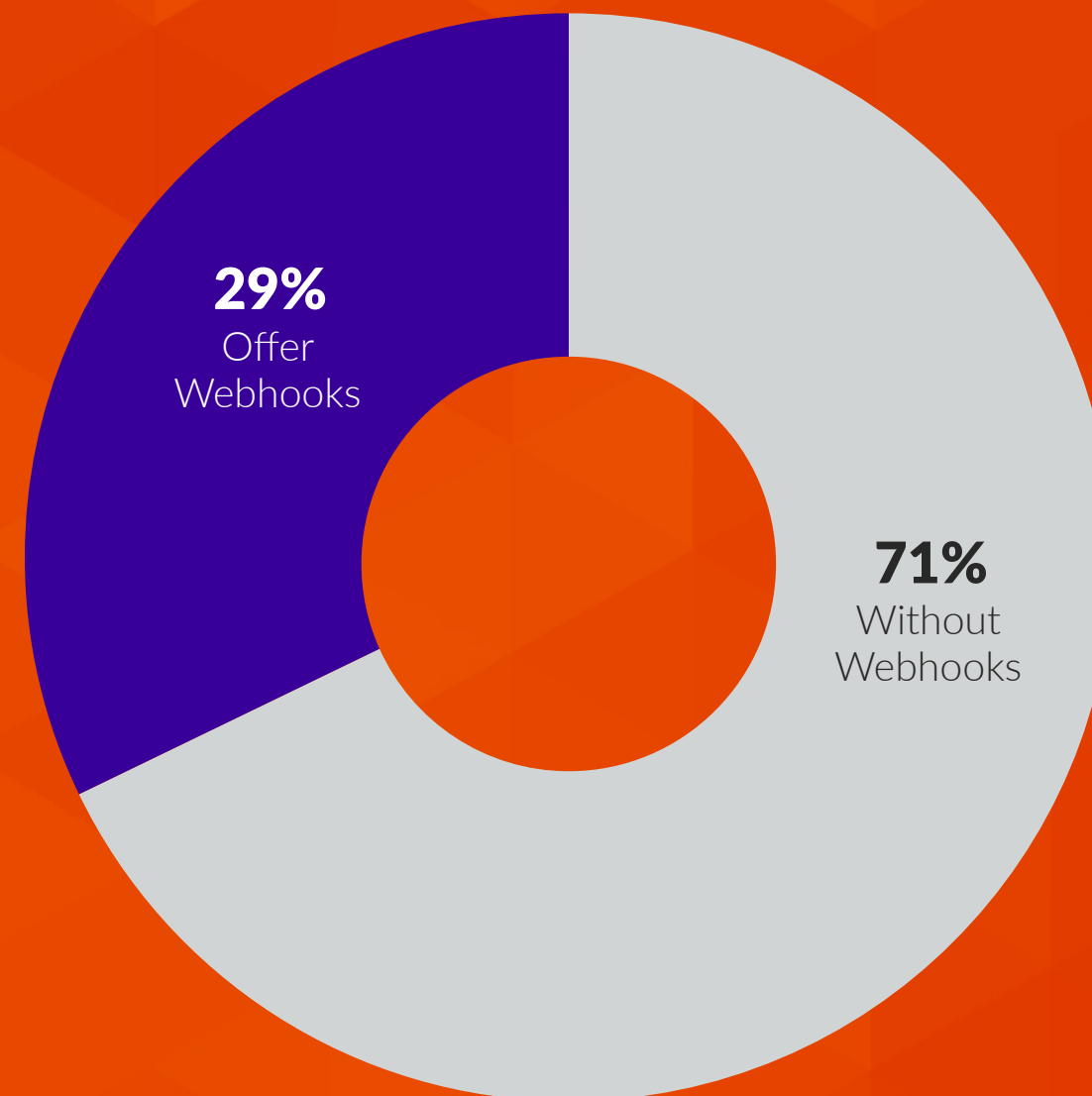
**Unfortunately, only 29 percent of APIs offer webhooks.**

Out of these APIs, 92 percent return JSON data, and 8 percent return SOAP or XML. Across the categories we profiled, Marketing Automation Software, Cloud Storage and e-Commerce offered the widest support for webhooks with 44 percent, 43 percent and 36 percent respectively.

# THE BENEFITS OF WEBHOOKS

Webhooks automatically post new event data to a user-defined URL monitored by the user's integrated applications. When a new event posts to the URL, the linked app updates to include this new data. The primary difference between webhooks and polling is that the hooks only update when there is new information available.

Webhooks work by automatically posting new event data to a user-defined URL which is monitored by the user's linked applications. Each time a new event is posted to the URL, the linked applications update to include the new data. Unlike polling which is successful on less than two percent of requests (according to research provided by Zapier), webhooks only update when new information is available. Because of the increased efficiency, over 82 percent of developers surveyed by Wufoo indicated that they would rather receive new data via webhooks instead of with polling. Despite this preference, only 29 percent of APIs currently support webhooks.



# EVENT AND SYNCRONIZATION

In most products without webhooks support, API integrations fall back to polling the application repeatedly, which is extremely wasteful. Approximately **98.5 percent of these polls** return no useable data. Another way to think about that: every 2 out of 100 times, you app's integration will receive no updated or changed data, when using a polling framework to manage synchronization.

Event-Driven Architecture is certainly not a new methodology, but it's enjoying renewed support in the world of integration because it reduces coupling between services and provides a far more efficient way of synchronizing and updating data.

If you have 3,000 applications interested in knowing when a new customer is created in your CRM, and if you are using traditional REST APIs, you will have 3,000 clients repeatedly asking the CRM platform "is there a new customer?", and most often getting the answer "no". An event-driven implementation reverses this integration model, where instead of clients asking if there is new data, the CRM platform tells each of the 3,000 applications when new data is available.

## POLLING RESULTS

**98.5% = FAIL**  
(No Usable Data)



## CLOUD ELEMENTS SUPPORT FOR EVENTS

**Event** - Cloud Elements can be configured to listen for and notify users of actions/changes (i.e., Creates, Updates) to endpoint resources in a normalized way. We leverage webhook events when available from an endpoint. If webhooks are not supported, we provide a polling service that polls for new or modified data at the endpoint. Whether we use webhooks or polling, we provide a consistent JSON payload of the action that occurred to a resource along with the object name and the object ID. For completeness we also include the vendor's raw payload of the data associated with that event. Note that in order to provide a polling event capability, an endpoint must provide the ability to search for resource updates. Polling also has an inherent limitation that cannot support deleted events.

---

## CHAPTER 4

---

# API CALCULATOR



# INTERACTIVE CALCULATOR TO SCORE YOUR API

To wrap things up, we wanted to give our readers a tool, mechanism, and calculator to gauge how your API currently stacks up against other APIs - focused on how easy it is for API consumers can integrate with your service. Here are two APIs we have scored based on the research and findings from this report. One good and one not so good -- we're not calling out products or vendors in particular, so we haven't listed the actual endpoints, but rather offer these examples to further quantify the current State of API Integration.



YOU CAN USE THE SCORECARD  
FOR YOURSELF TO DETERMINE  
YOUR **SCORE** OUT OF A  
POSSIBLE TOTAL OF 100.



**CLICK HERE TO FIND YOUR SCORE!**

<http://bit.ly/2od6cVJ>

# CRM ENDPOINT SCORECARD (BEST-IN-CLASS API)

CRITERIA		NOT APPLICABLE	POOR (1 POINT)	OK (4 POINTS)	GOOD (7 POINTS)	EXCELLENT (10 POINTS)	TOTAL (POINTS)
STYLE	We have ranked API style from Hypermedia (Excellent), REST (Good), SOAP (OK) & Other (Poor)					✓	10
AUTH	API Authentication models ranked OAuth 2.0 (Excellent), OAuth 1.0 (Good), API Key/Secret (OK), HTTP Basic Auth (Poor)					✓	10
API DOCUMENTATION & DATA TYPES	API Documentation options ranked from OAI/Swagger (Excellent), OData (Good), XML or Other (OK), None (Poor)					✓	10
API METADATA DISCOVERY	API Metadata Discovery capabilities ranked from Full Object & Type Discovery (Excellent), Objects only (Good), Type only (OK), None (Poor)					✓	10
EVENTS	API eventing support ranked from Pub/Sub (Excellent), Webhooks (Good), Comet (OK), None (Poor)				✓		7
BULK SUPPORT	Bulk upload & download support where applicable, is ranked Supported (Excellent) or Unsupported (Poor)					✓	10
PAGINATION	API Pagination support is ranked from Full Link Headers (Excellent) Paging with total count (Good), Paging without count (OK), Paging without page size config (Poor). Or not applicable where necessary.				✓		7
HTTP VERBS	Good RESTful design dictates that the appropriate HTTP Verb be used for each API action, and implementation is ranked from Complete (Excellent), Partial (Good), Inconsistent (OK), Everything is POST or GET (Poor)					✓	10
VERSION CONTROL	API Version control has been ranked according to developer experience from Default version (Excellent), URI version (Good), Content Negotiation or HTTP Header (OK), Inconsistent (Poor). Or if you haven't got to v2 yet then Not Applicable.				✓		7
ERROR HANDLING	Consistency and information provided across API error handling is ranked from Complete (Excellent), Correct categories with limited info (Good), Some Error Codes (OK), Inconsistent or Embedded (Poor)					✓	10
TOTAL SCORE:							91/100

# CRM ENDPOINT SCORECARD (WORST-IN-CLASS API)

CRITERIA		NOT APPLICABLE	POOR (1 POINT)	OK (4 POINTS)	GOOD (7 POINTS)	EXCELLENT (10 POINTS)	TOTAL POINTS	
STYLE	We have ranked API style from Hypermedia (Excellent), REST (Good), SOAP (OK) & Other (Poor)			✓			4	
AUTH	API Authentication models ranked OAuth 2.0 (Excellent), OAuth 1.0 (Good), API Key/Secret (OK), HTTP Basic Auth (Poor)		✓				1	
API DOCUMENTATION & DATA TYPES	API Documentation options ranked from OAI/Swagger (Excellent), OData (Good), XML or Other (OK), None (Poor)				✓		7	
API METADATA DISCOVERY	API Metadata Discovery capabilities ranked from Full Object & Type Discovery (Excellent), Objects only (Good), Type only (OK), None (Poor)				✓		7	
EVENTS	API eventing support ranked from Pub/Sub (Excellent), Webhooks (Good), Comet (OK), None (Poor)		✓				1	
BULK SUPPORT	Bulk upload & download support where applicable, is ranked Supported (Excellent) or Unsupported (Poor)						✓	10
PAGINATION	API Pagination support is ranked from Full Link Headers (Excellent) Paging with total count (Good), Paging without count (OK), Paging without page size config (Poor). Or not applicable where necessary.		✓				1	
HTTP VERBS	Good RESTful design dictates that the appropriate HTTP Verb be used for each API action, and implementation is ranked from Complete (Excellent), Partial (Good), Inconsistent (OK), Everything is POST or GET (Poor)		✓				1	
VERSION CONTROL	API Version control has been ranked according to developer experience from Default version (Excellent), URI version (Good), Content Negotiation or HTTP Header (OK), Inconsistent (Poor). Or if you haven't got to v2 yet then Not Applicable.		✓				1	
ERROR HANDLING	Consistency and information provided across API error handling is ranked from Complete (Excellent), Correct categories with limited info (Good), Some Error Codes (OK), Inconsistent or Embedded (Poor)		✓				1	
TOTAL SCORE:							34/100	

# THE STATE OF API INTEGRATIONS

# YOUR API SCORECARD

Here's a blank canvas, download it, fill it out, get your score and share it! We'd love to hear how your API measures up against the rest of the industry. Use the hashtag **#APIScorecard** and **#StateOfAPIIntegration** to help us track your data.

CRITERIA		NOT APPLICABLE	POOR (1 POINT)	OK (4 POINTS)	GOOD (7 POINTS)	EXCELLENT (10 POINTS)	TOTAL 4 POINTS	
STYLE	We have ranked API style from Hypermedia (Excellent), REST (Good), SOAP (OK) & Other (Poor)			✓				
AUTH	API Authentication models ranked OAuth 2.0 (Excellent), OAuth 1.0 (Good), API Key/Secret (OK), HTTP Basic Auth (Poor)		✓					
API DOCUMENTATION & DATA TYPES	API Documentation options ranked from OAI/Swagger (Excellent), OData (Good), XML or Other (OK), None (Poor)				✓			
API METADATA DISCOVERY	API Metadata Discovery capabilities ranked from Full Object & Type Discovery (Excellent), Objects only (Good), Type only (OK), None (Poor)				✓			
EVENTS	API eventing support ranked from Pub/Sub (Excellent), Webhooks (Good), Comet (OK), None (Poor)		✓					
BULK SUPPORT	Bulk upload & download support where applicable, is ranked Supported (Excellent) or Unsupported (Poor)						✓	
PAGINATION	API Pagination support is ranked from Full Link Headers (Excellent) Paging with total count (Good), Paging without count (OK), Paging without page size config (Poor). Or not applicable where necessary.		✓					
HTTP VERBS	Good RESTful design dictates that the appropriate HTTP Verb be used for each API action, and implementation is ranked from Complete (Excellent), Partial (Good), Inconsistent (OK), Everything is POST or GET (Poor)		✓					
VERSION CONTROL	API Version control has been ranked according to developer experience from Default version (Excellent), URI version (Good), Content Negotiation or HTTP Header (OK), Inconsistent (Poor). Or if you haven't got to v2 yet then Not Applicable.		✓					
ERROR HANDLING	Consistency and information provided across API error handling is ranked from Complete (Excellent), Correct categories with limited info (Good), Some Error Codes (OK), Inconsistent or Embedded (Poor)		✓					
YOUR SCORE:							___/100	

# INTERACTIVE CALCULATOR TO SCORE YOUR API

You can use the scorecard for yourself to determine your score out of a possible total of 100.



**CLICK HERE TO FIND YOUR SCORE!**

<http://bit.ly/2od6cVJ>



# CLOSING

And there you have it. We hope you enjoyed seeing the data and hearing from our contributors about the trends that are set to affect APIs and application integration. Don't forget to see how your API measures up against the rest of the industry - score your API and share it! Follow the conversation by using the hashtag **#APIScorecard** and **#StateOfAPIIntegration**.

Check out our [resource center](#) for other easy to follow information. If you have questions, or would like to learn more about Cloud Elements, [contact us today!](#)



TRY CLOUD ELEMENTS FOR FREE



# ABOUT CLOUD ELEMENTS

It's bigger than just a connection, integrating APIs is about moving data, powering transactions, connecting mobile apps and thousands of other functions.

Cloud Elements is a cloud API integration platform that enables developers to publish, integrate, aggregate and manage all of their APIs through a unified platform.

Using Cloud Elements, developers can quickly connect entire categories of cloud services (e.g. CRM, Documents, Finance) using uniform APIs or simply synchronize data between multiple cloud services (e.g. Salesforce, Zendesk, Quickbooks) using its innovative integration toolkit.

Founded in October 2012, Cloud Elements is purpose built for developers to help organize their world of APIs through a one-to-many approach. A 'Visionary' in API Management, according to Gartner Inc., Cloud Elements is headquartered in Denver, CO, but serves customers worldwide.



Learn more about us at:

**[WWW.CLOUD-ELEMENTS.COM](http://WWW.CLOUD-ELEMENTS.COM)**