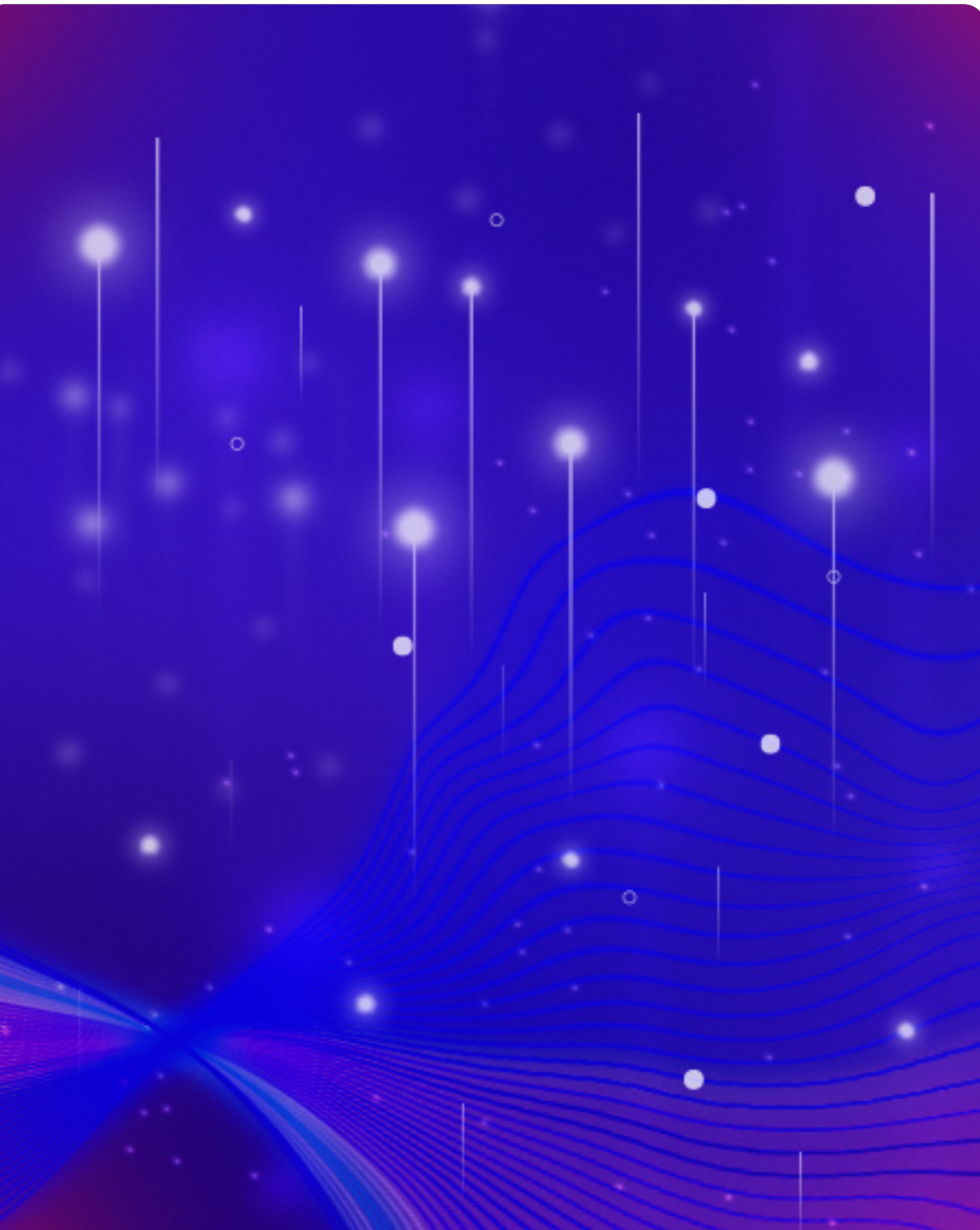


WHITE PAPER



The ultimate event tech stack:

Getting your data architecture right



Grip.events

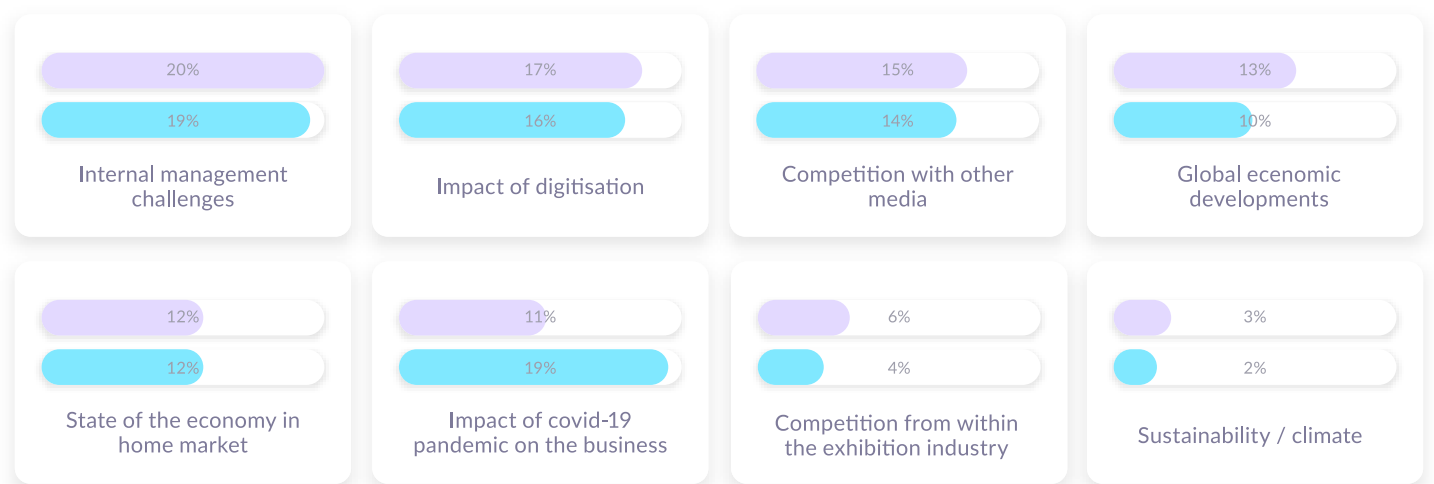
hello@grip.events

Contents

- 3 Introduction
- 4 Data architecture fundamentals
- 5 Why is data architecture important?
- 6 Data architecture best practices
- 7 Five data architectures maps for commercial organizers
- 8 How to read the event data architecture diagrams
- 10 Diagram example
- 11 Data architecture examples
- 18 Missing systems
- 19 Next steps to better understand your own data architecture
- 21 Appendix

Introduction

The recent [UFI Exhibition Barometer](#) shows digitization is proving to be a major challenge for conference, events and exhibition organizers.



It's clear that all types, and sizes, of organizers are using various technologies to run events, or series of them, which is delivering a fragmented user experience. A multiple number of log-ins and an increasingly complex data architecture is resulting in a sub-optimal experience for organizers, exhibitors and visitors alike.

When it comes to simplifying things, organizers often find it difficult to know whether their particular data architecture is suitable for their size and shape and if the challenges they face are unique or widely shared.

This white paper provides clarity on what we believe is the best possible structure for organizers of different sizes and shapes. This will hopefully help structure your technology stack, the various data architectures that are possible, and where you might be able to streamline your processes.

While the examples are based on learnings about hundreds of different data architectures across the events industry, they are generic, and there will be nuances in the structure, based on the unique objectives of your business.



WHITE PAPER

Data architecture fundamentals

The first hit on Google links to a [fabulous article](#) on data architecture which defines it as follows:

“Data architecture translates business needs into data and system requirements and seeks to manage data and its flow through the enterprise.”

This is a great way of thinking about data architecture, it is there to serve the needs of the business. This also means that as the needs of the business change, so does its data architecture, which is a situation that many

event organizers find themselves in today. Data architecture is the foundation for effectively managing data.

In order to understand and make decisions about it, organizations must be able to represent their organizational information at different levels of abstraction so that users can comprehend its complexity while still being capable enough in making informed choices with what they find through analytics tools or business intelligence reports for example.

Why is data architecture important?

There is a range of reasons why a strong data architecture is important for event organizers, they include:



- > **Operational efficiency:** Organizations end up wasting a lot of time bringing data from one system to another. As labour becomes increasingly expensive, and event teams are required to do more events (virtual and in-person) with less people, a streamlined way of creating and managing events will become even more important.
- > **Cost reduction:** With the looming recession across the western world, organizers will have to review their systems and whether there is room for consolidation, a strong understanding of the needs of the business and whether existing or new suppliers can take on the function of various providers is an obvious way to cut costs.
- > **Customer experience:** Every organizer will know about an exhibitor complaining about having to upload the same information multiple times and having to remember multiple login details. Strong data architecture and understanding of the customer journey will result in a superior customer experience.
- > **Competitive advantage:** Data gives a competitive advantage and can lead to a greater understanding of customers and their behaviours resulting in better business strategy and customer experience, as described in this [Harvard Business Review article](#). The potential for so-called 'data network effects' is also described in our white paper [The New Competitive Advantages of Event Organizers](#).

Data architecture best practices

When creating an efficient data architecture, you should follow the below best practices, while they all seem obvious, very few organizations actually follow any of them, let alone all of them:

- > **Data should not exist in silos:** The goal of creating a data architecture is to ensure that datasets are in a central repository. There should also be a flow of information between these datasets.
- > **Standardize data entity creation:** Follow the highest standards when creating data entities. For instance, apply constraints like primary keys and null allowances for relational datasets.
- > **Use entity-relationship diagrams (ERDs):** They help you create and understand the relationships between data entities (such as tables in a database). Thus, they should be a part of the standard procedure for datasets in relational databases.
- > **Update data architecture and ERDs:** As you create data entities, update the existing data architecture and ERDs.
- > **Create a data architecture document:** As stated earlier, create a data architecture document. Furthermore, a compliance team should review this document regularly to keep it up-to-date.
- > **Make data structures consistent:** The data structure in the repository should be **consistent with data visualization** and reporting requirements. Also, the data architect needs to ensure that there is consistency in the different data flows and architecture for the various business products and services used within the organization.
- > **Automate the Extract, Transform and Load (ETL) process of data between systems:** Data is automatically loaded to various systems when you automate the ETL process.

Five data architectures maps for commercial organizers

As mentioned in the introduction, every organizer has different needs depending on the type, shape and size of their organization. Simple questions such as, who you consider a customer and what is the monetary value of a single customer, have a profound impact on your data architecture.

Therefore, we have created five data architecture maps for exhibitor, conference and hosted buyer event organizers, then segmented these into small/medium and large organizers.

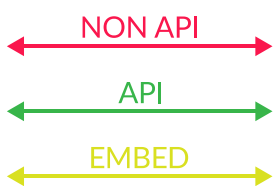
Customer type	Large organizer	Small / medium organizer
Exhibition organizer	<ul style="list-style-type: none"> > £10M+ in revenue > 80%+ revenue from booth sales > 100+ exhibitors per event 	<ul style="list-style-type: none"> > Less than £10M in revenue > 80%+ revenue from Booth sales > Less than 50 exhibitors per event
Conference organizer	<ul style="list-style-type: none"> > £10M+ in revenue > 40%+ non-booth revenue (content, tickets, meetings, sponsorship) 	<ul style="list-style-type: none"> > Less than £10M in revenue > 40%+ non-booth revenue (content, tickets, meetings, sponsorship)
Hosted buyer	<ul style="list-style-type: none"> > 70%+ of revenue coming from scheduled meetings > Less than 500 participants per event 	



WHITE PAPER

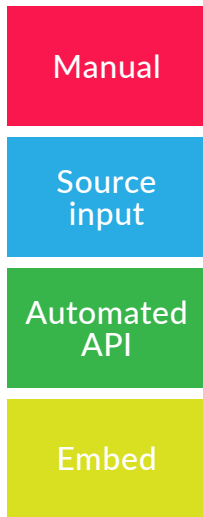
How to read the event data architecture diagrams

To read the diagrams you'll see three types of entities:



Lines: Defines how data is 'transferred' between systems, there are three options:

- > **Non-API:** e.g. through a CSV uploader or with a genuine manual, line-by-line input process.
- > **API:** this can be a push, pull or webhook API. We won't go into the benefits of webhooks or the difference between JSON or XML but if everything is 'linked' that's a great start.
- > **Embed:** the best way to streamline data is by avoiding the need for integration, embedding information in another system can be an effective way to avoid the cost of a complex integration especially for smaller organizers.



Sqaure rectangle: Defines the data stored in a system, the colour defines how data has entered into the system. There are four ways data can end up in the system.

- > **Manual:** Through a CSV uploader or the manual copying of information.
- > **Source input:** There needs to be an initial system where the data is entered, this is important because the limitations of this system will dictate all future steps.
- > **Automated API:** The data entered the system through an API integration.
- > **Embed:** The data is available within a system but because it's embedded, meaning it's not actually stored within the particular system and therefore no integration was needed.

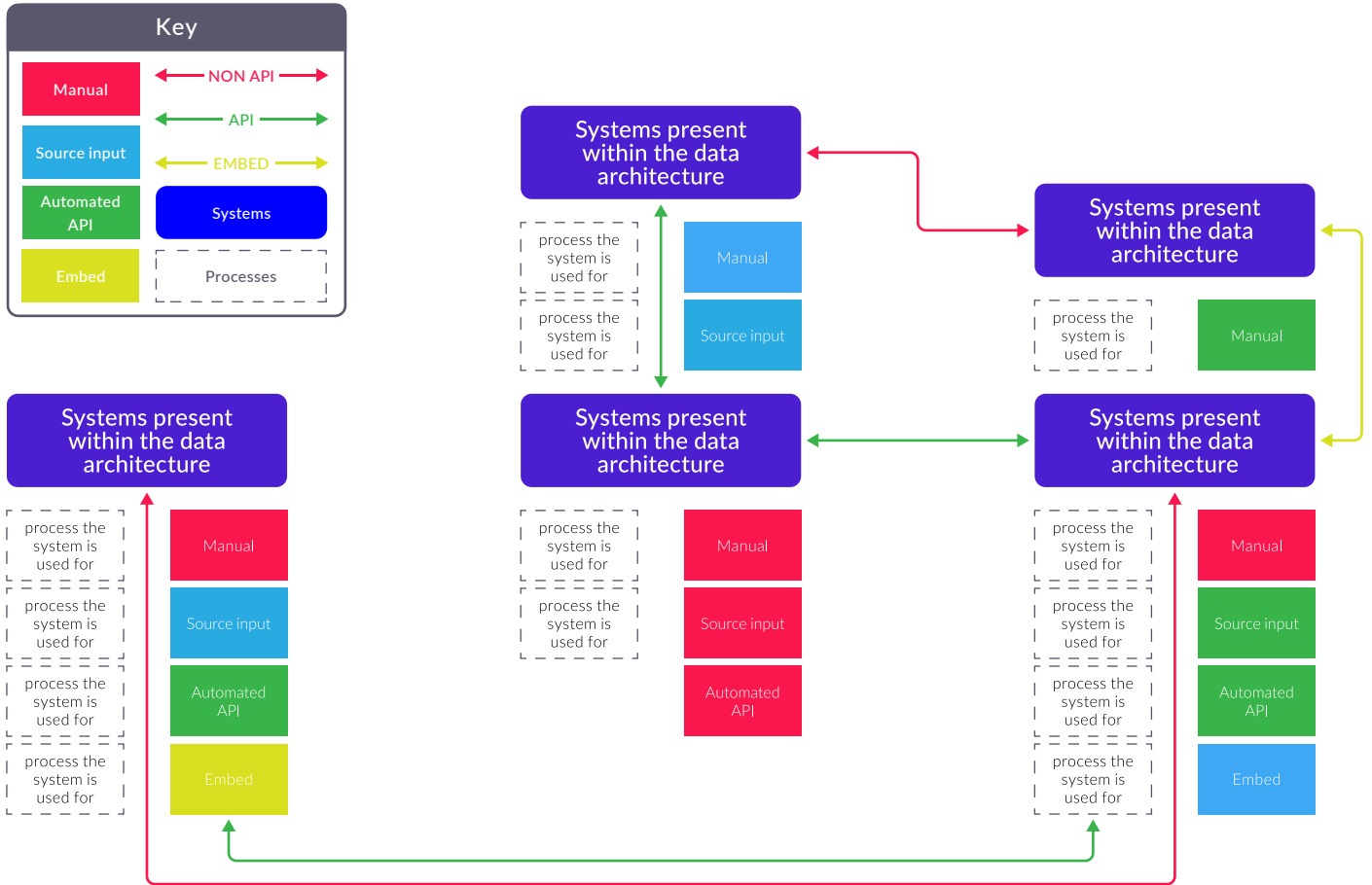


Rounded Rectangle: define the systems present within the data architecture. There are no differences between the rectangles.



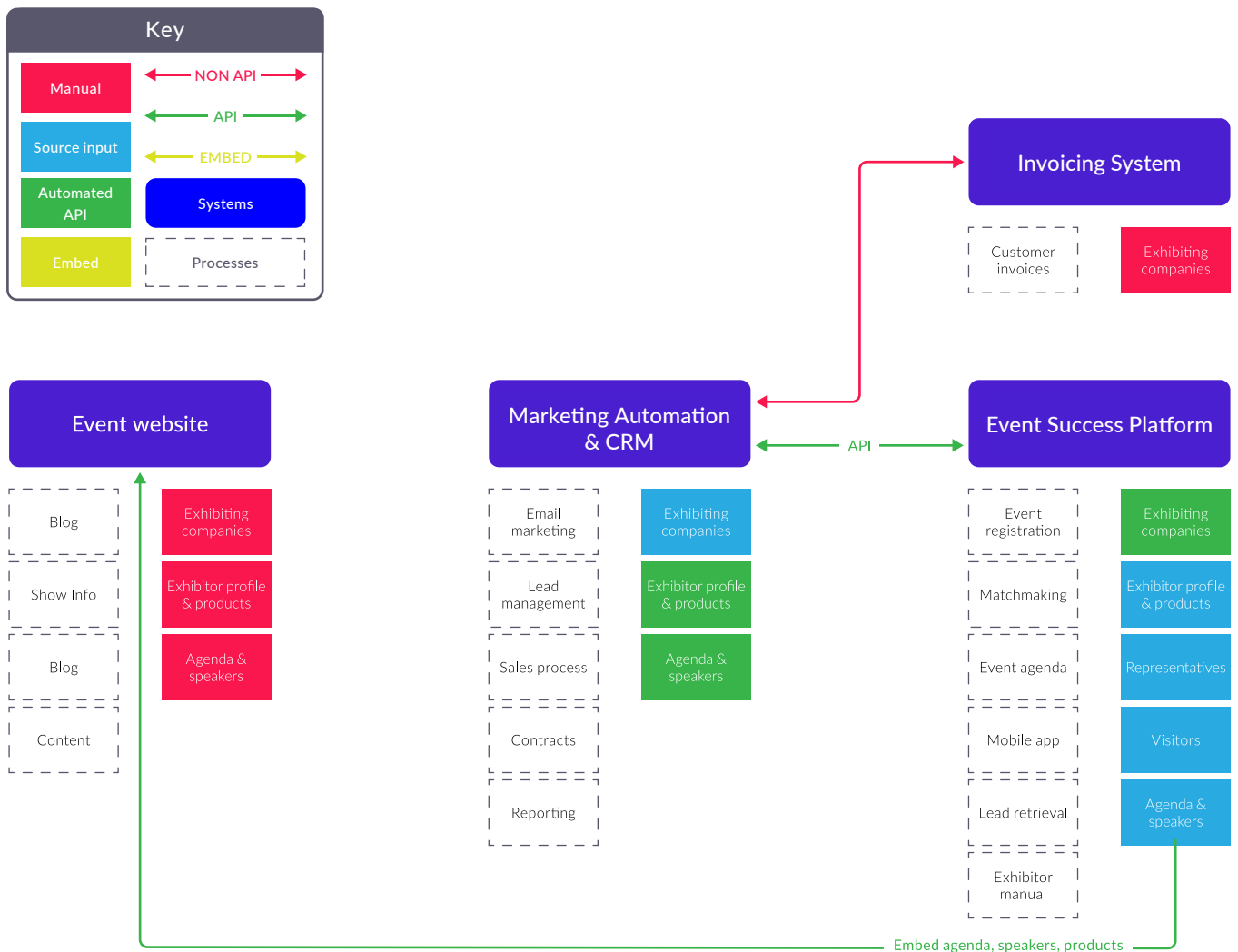
Dashed rectangles: define the process the system is used for. You will notice that while systems might be the same, the process they are used for might change. This means that the business needs might exceed what can be expected from a typical system in this category.

Example



Data architecture examples

Small / medium conference organizer



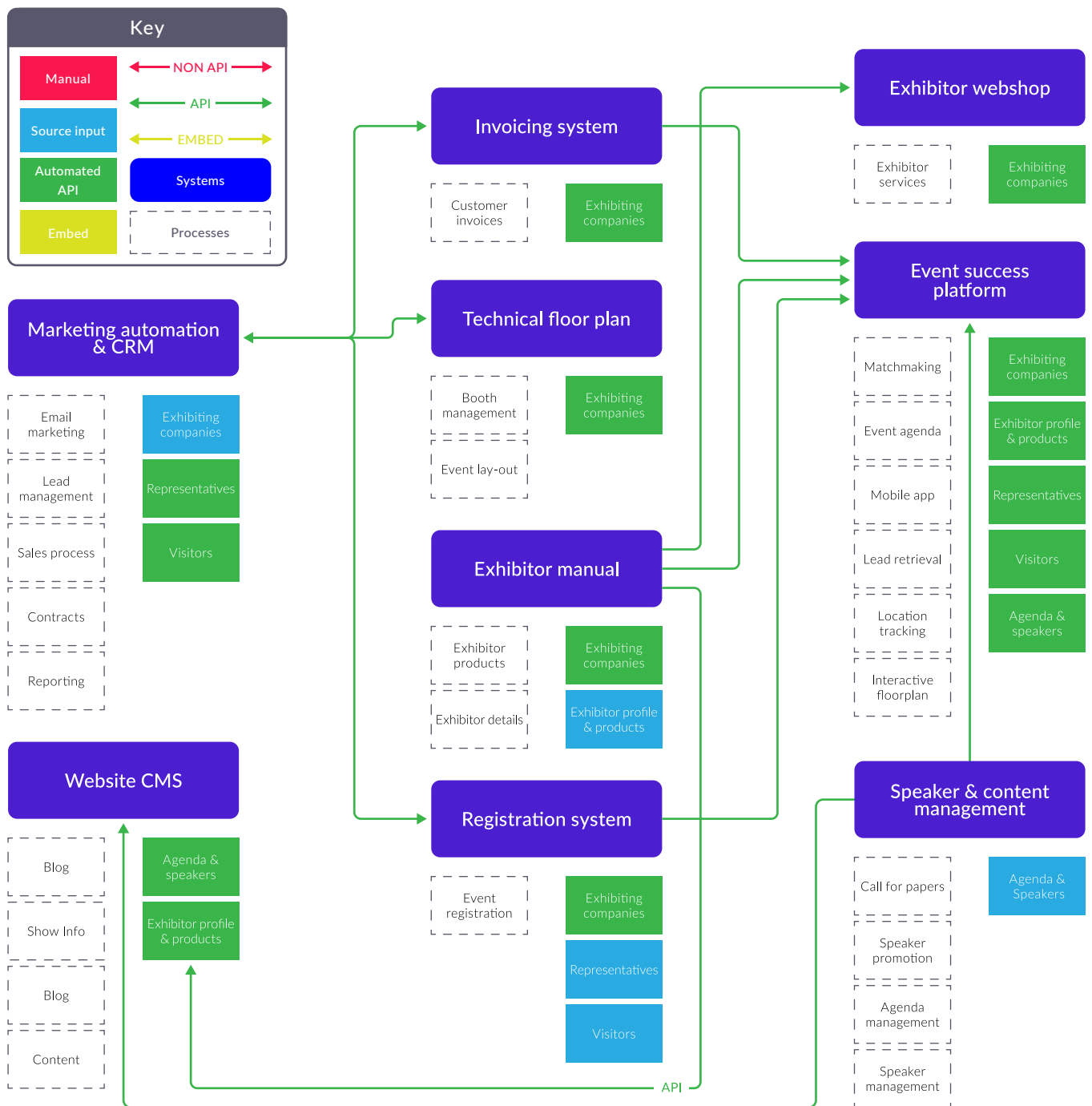
Small teams with limited budget and high pressure result in small to medium-sized conference organizers often ending up with many poorly integrated systems. Our recommendation is to keep it simple.

With just three main systems, and only one integration you'll be able to keep data flowing seamlessly between your CRM / marketing automation and event platform.

The reason we recommend a separate event website is because of the need for a nimble and powerful CMS that is able to stay ahead of the competition. Webflow and other no-code CMS systems are a good option for this.

While there is the temptation to introduce more systems, keeping it simple and minimising the amount of data to be synced will result in the best user experience whether that be for you, the organizer or your visitor, exhibitor, sponsor and speakers.

Large exhibition organizers



Large exhibition organizers diagram 1 of 2

By far the most complex architecture is for large-scale exhibition organizers; the plethora of different events with different needs means that every major business process ends up requiring a dedicated enterprise system.

Here, there is often a lot of focus spent on exhibitors, where centralised operational teams want an exhibitor manual and technical floorplan solution that can be used across all shows. The event platform of choice needs to be flexible depending on each event's specific needs.



As content teams typically sit within show teams for exhibition organizers, a key piece of technology that is underutilised is a speaker and content management solution. Content is increasingly important and management of the standardization of these processes will result in greater insight, feedback and scale advantages over time.

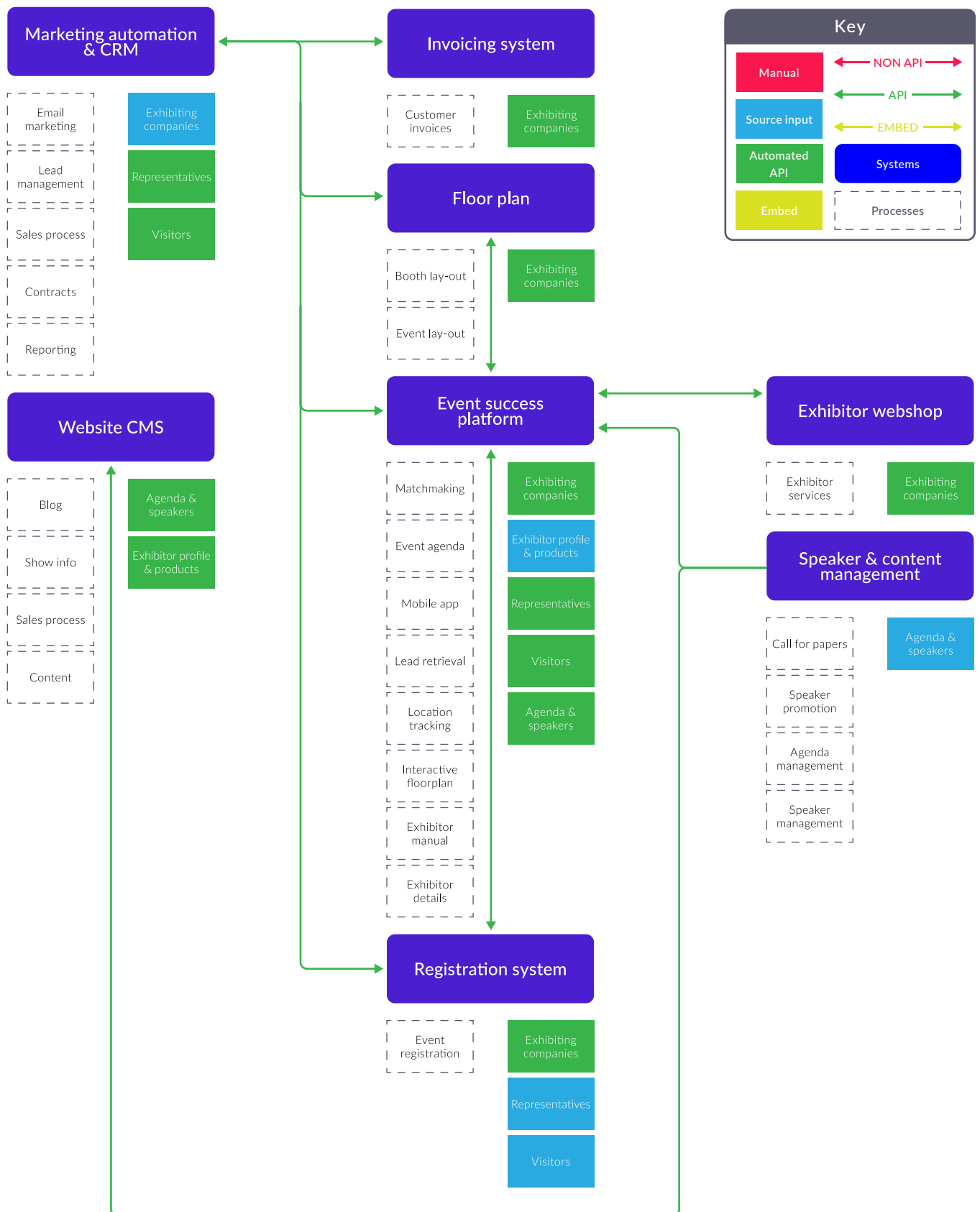
Due to the event data architecture being complex, there are often gaps in the data available in various systems making it more difficult to innovate and roll out new ideas. It also results in it costing more to introduce new systems, for example, the introduction of a mobile app could easily result in three separate integrations (speaker management, registration,

exhibitor manual) while this would often only be one integration for a small conference organizer. That's a significant difference and makes innovation harder and slower.

The event tech stack of exhibition organizers is quickly evolving as event platforms take an increasingly central role.

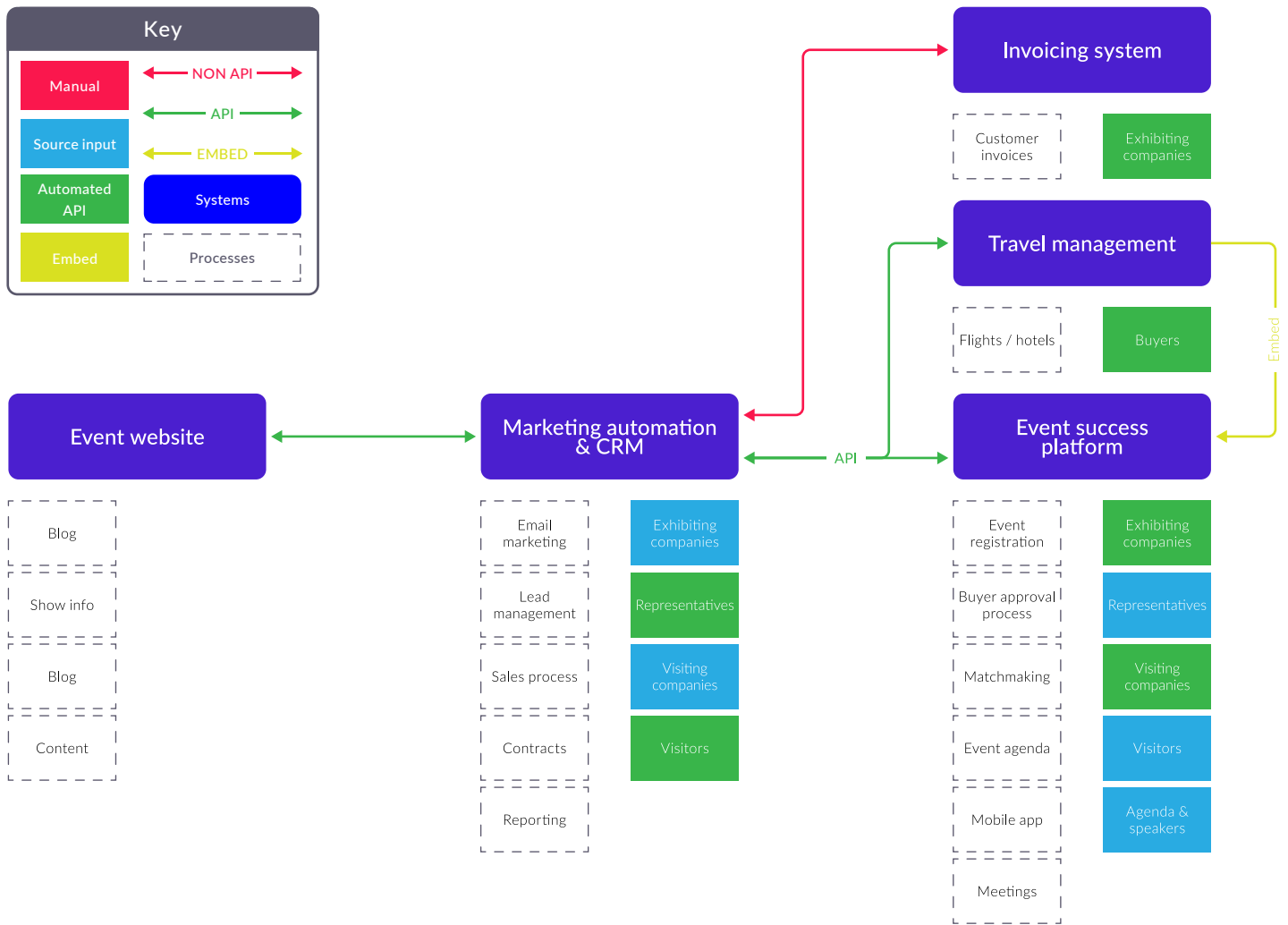
The main impact this will have is the consolidation and streamlining of the experience and the functionality of online exhibitor manuals. For example, the exhibitor manuals will need to be integrated into the event platform as it's a critical component of the exhibitor success journey.

Large exhibition organizers continued



Large exhibition organizers diagram 2 of 2

Hosted buyer event organizers



Hosted buyer event organizers typically have an approval stage within their registration process meaning they collect buyers that are 'interested' or are being 'invited' and then they make a selection of who is 'approved' to register.

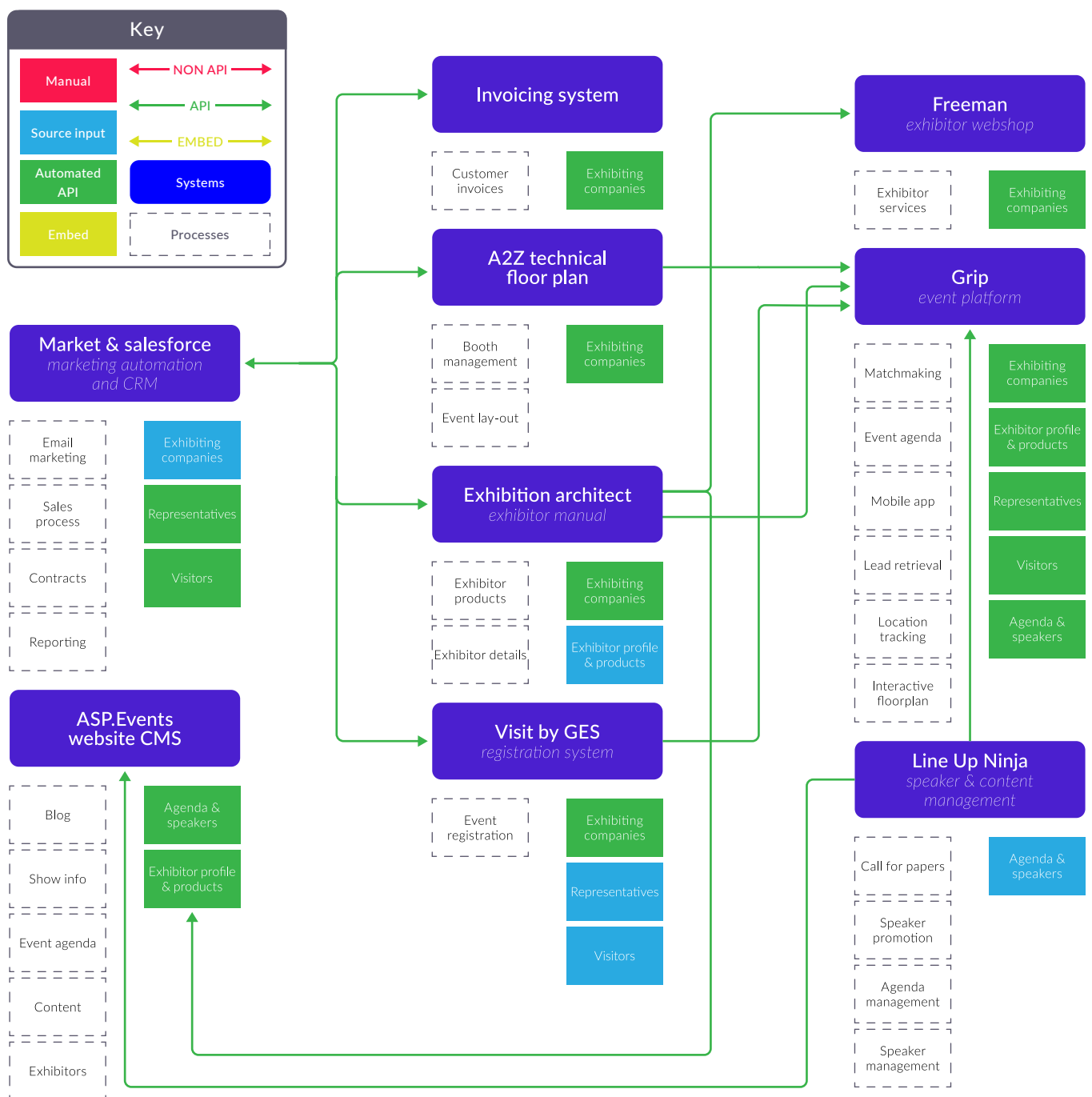
This requires some specific functionality in either the event registration system or within the CRM or marketing automation platform. There are pros and cons to both approaches and it depends on the level of sophistication of both systems as well as the depth of integration between the CRM and the event platform.

Our recommendation is for a deep integration between the CRM and the event platform that is two-way and synchronizes information such as meetings, meeting ratings, feedback and other vital information back from the event platform into the CRM.

This results in the sales team having a singular view of the success of a seller (or supplier) within the CRM system which helps them with their renewal conversations as well as understanding which sellers are 'at risk'. It also helps buyer teams to understand if there are any trends in which buyers are rated poorly across a larger number of events.

The last component that makes the technology stack of hosted buyer event organizers different is the need for a travel management technology in which the flights, hotels and personal details (passports, etc.) of buyers can be managed.

We recommend keeping this in a separate system for compliance reasons (highly sensitive personal information) but also because it's only relevant to a small segment of events. Meaning that even if an event platform does this as part of its solution, it's probably not going to be up to the level that you need. Having a separate well-integrated travel management system is a better solution in our opinion.

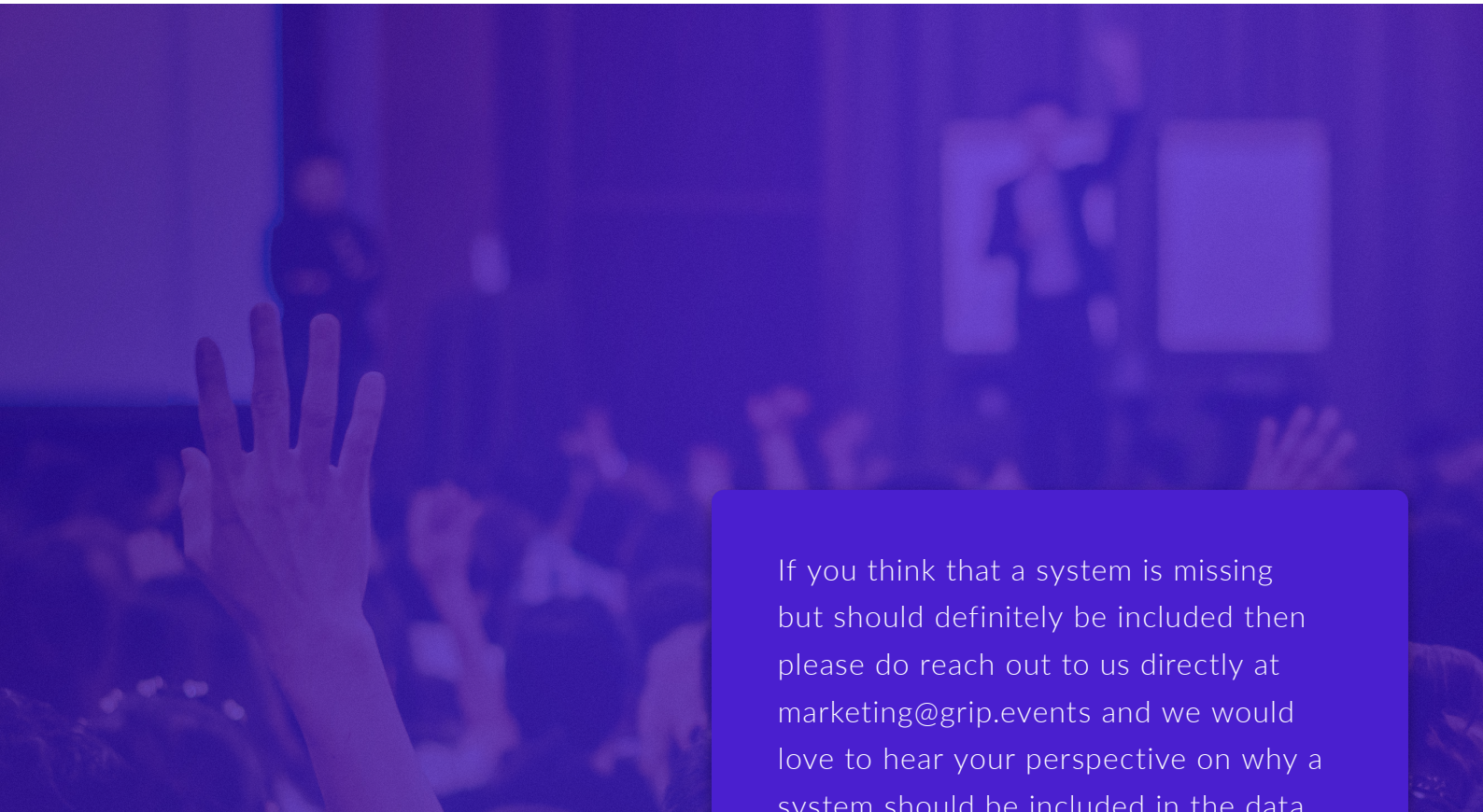


Missing systems

For all diagrams, while the main systems are included there are others that are not included such as audience interaction, community amplification, streaming solutions and many others.

We've either excluded these as there is limited to no data that is stored within them or they are a sub-process within another system. Community amplification, as an example, while a valuable step in

the customer journey, is typically tightly integrated with the registration system and therefore requires minimal management from a Head of Digital or CIO at a large conference, event or exhibition organizer.



If you think that a system is missing but should definitely be included then please do reach out to us directly at marketing@grip.events and we would love to hear your perspective on why a system should be included in the data architecture overview.

Next steps to better understand your own data architecture

Below are the steps you can take to better understand your current state and what you can do to improve your event tech stack and data processes.

Key stakeholders involvement

There is a good chance this exercise will require several stakeholders already within your organization, ensuring you are all on the same page in terms of what data sits where and why this process is required.

Create a CRUD Matrix

The process of listing every place where data is Created, Read, Updated, and Deleted (CRUD) can provide insight and surprises. A CRUD matrix can show, for example, how an address change is entered but subsequently overwritten by someone else, or how and where multiple people are creating the same data.

This process can also uncover critical data entered manually and stored on spreadsheets. Often overlooked, ignoring this step can lead to more serious problems down the line, especially when the data is not synchronized across multiple platforms.

Create your own 'data architecture diagram'

Platforms such as Miro are great for creating our event data architecture, but starting out with PowerPoint can be just as easy. There are many templates out there but consider just using the four types of lines, rectangles and barrels, as we used in the section 'How to read event tech architecture diagrams.'

Future event tech partners

Creating your own data architecture will help streamline conversations about integrations with new technology providers or partners. Being able to show a diagram of what technology systems you use, how they are integrated and what they are used for will make the process of selecting new technologies or partners significantly easier for you.

Talk to us

We love working with organizers who are invested in evolving their business needs and those wanting to understand how technology could grow with them. We would be happy to talk to you about your challenges and what possible solutions are for your business.



Tim Groot

Founder & CEO - Grip



tim@grip.events

Appendix

1. Glossary

Entity-relationship diagrams (ERD)

An entity relationship diagram (ERD) is a graphical representation of the relationships between entities (or tables) in a database. ERD can also be referred to as an Entity Relationship Model (ERM) and can be used to model data in a relational database.

Extract, transform, load (ETL)

ETL can be used to transform data from one format to another, or to merge data from multiple sources into a single target database.

No-code content management system (CMS)

A no-code content management system (CMS) is a content management system that does not require coding skills to use. No-code CMSs are often visual, drag-and-drop systems that allow users to create and edit web content without needing technical skills.

Enterprise resource planning (ERP)

An Enterprise Resource Planning (ERP) system is a software solution that helps businesses manage their core operations, such as accounting, HR, and supply chain management which can improve efficiency and data accuracy.

Customer relationship management (CRM)

A customer relationship management (CRM) system is a software solution that helps businesses manage their customer data. CRM systems can be used to track customer interactions, sales, and marketing activities. CRM systems can also help businesses automate their customer service and support processes.

Travel management system

Travel management systems helps travel managers book, track and analyse business travel typically used for hosted buyers in the events industry. This can include travel policies, reporting and overall management of the individuals travel requirements.

Created, Read, Updated, and Deleted (CRUD) matrix

A CRUD matrix is a tool used to map out the relationships between data, operations, and user interface elements in a software applications. The matrix can be used to help design or redesign the UI, or to understand how changes to the data or operations will impact the UI.

Appendix

2. Platforms mentioned

Webflow <https://webflow.com/>

Sitecore <https://www.sitecore.com/>

Miro <https://miro.com/>

Grip

Mindspace, 9 Appold St
EC2A 2AP, London
United Kingdom

hello@grip.events

