

Case study

# Elements on Elements

An insight into how we have used Elements to help us grow and change our own Salesforce Org.



# Contents

<b>1. Exec summary</b>		<b>03</b>
<hr/>		
<b>2. Case study</b>		<b>05</b>
2.1	Why this app?	<b>05</b>
2.2	How we drive business agility and change	<b>07</b>
2.3	Building what users really need: ANALYZE	<b>11</b>
2.4	Build quickly, but build for long term: BUILD	<b>18</b>
2.5	Go-live: DELIVER	<b>22</b>
2.6	Running an agile business: OPERATE	<b>25</b>
2.7	What are the results?	<b>28</b>
<hr/>		
<b>3. About us</b>	3.1	<b>The founders</b>
		<b>33</b>



# Executive summary

## A history with Salesforce

The founders have been customers and partners of Salesforce since 2001 so understood the challenges of managing complex implementations. Seeing the gap in enterprise tooling Elements.cloud was launched in 2016. It is emerging as the de facto standard for Salesforce customers who want to stay in control of their Orgs. It supports the implementation lifecycle enabling releases to be delivered more quickly, with greater end user adoption, and the confidence that the Org will not break.

Our bold claim is that we have the best run Salesforce Org on the planet. If any customer can better us on any metric, then they can have Elements for free. So let's look at the numbers.

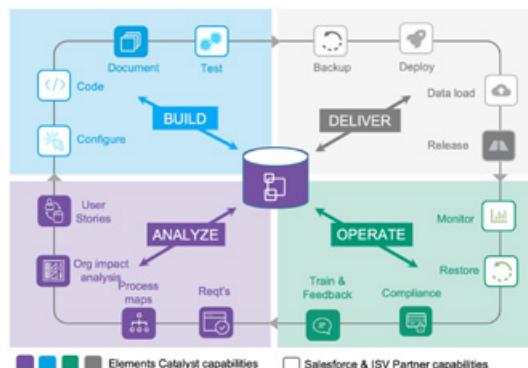
## Best run Org metrics

- On average, we are closing 40 change requests per month.
- User adoption has increased month-on-month for the last 18 months.
- We have such a well documented Org that we have only had 1 release rollback in the last 3 years, and this was due to a flow that included an external app integration.
- Training is at point-of-need, and the latest sales hire was trained in 1 week.
- Users report that in-app help, as well as uncluttered page layouts, speed up workflows considerably.
- The feedback buttons on objects and fields empower users to communicate directly with Admins and speed up change lifecycles.



## Drinking our own champagne

The best way to look at our approach is to dig into the implementation lifecycle, phase by phase. We use every aspect of our own platform, Elements. We are our best customer, but also our harshest critic.



### Analyze

Our aim is to build the right thing the first time, which gives better user adoption in the final, Operate phase. We take a lot of time to understand the business problem we are trying to solve. We create really crisp user stories based on clear requirements, looking at the operational process maps and the data model.

We then analyse the risk of every user story. If the release risk is low, then we can safely develop in Production. If the risk is high we put it through a multi-stage development process. This method ensures we maintain as much business agility as possible and match the development effort to the risk.

### Build

The user stories are sync'd to Jira. Then we decide on how new functionality will be built: code or declarative. As we build, we rely on the Elements Org Model, which is the metadata dictionary that is kept in sync with the Salesforce Production and Sandbox Orgs. It has automated impact analysis and documentation, and we can then add additional documentation about what we configured, and more importantly "why". This is crucial to improving the impact assessment of future changes and maintaining an agile Org.

### Deliver

We are rolling out Copado to drive the changes through our deployment pipeline and into Production more quickly. This eliminates needing to use change sets to migrate metadata, but also adds more control and reporting. At the same time as the Salesforce changes, we also need to deploy the updates to help/training which could be process diagrams, videos, and other training content.

### Operate

Operate is the ongoing monitoring of the system, tracking usage and capturing feedback. We also need to make sure that regulatory compliance is met. We use the Elements in-app help functionality to enable notes, documents, URL links and process diagrams to be linked to objects and fields, and process diagrams are embedded into record pages. There is a feedback button connected to every object and field on page layouts. Having a way for users to give feedback in the moment as they use Salesforce drives the next set of requirements and also increases user engagement.

2.1

## Why this app?



## 2.1 Why this app?

In this case study, we want to share how we "drink our own champagne" i.e. use Elements to help us grow and change our own Salesforce Org. This impacts all parts of the business so we've had input from the different teams: Jack Lavous (Business Excellence and Admin), Adrian King (COO/CTO), Ian Gotts (CEO), Rob Brown (Sales) and Alyssa Abbey (Marketing and Community).

The three founders of Elements.cloud: Ian Gotts, Richard Parker and Adrian King have been using Salesforce since 2001 with their previous company, Nimbus.

**"Elements.cloud has a single internal platform for all customer and partner engagement within salesforce.org"**

Think of it - there was no Dreamforce yet, and Ian remembers speaking at the first London Salesforce event with Marc and 120 delegates. The only standard objects then were Accounts, Contacts, Leads and Opportunities, so they custom built virtually everything the business needed: 285 custom objects covering HR, compliance, hosting, partners...and the list goes on. To enable Nimbus to fully understand how the Org reflected their processes, and what might break if they changed something, they built an Org model (Metadata Dictionary) by hand, but it was a nightmare to maintain. Hugely valuable, but time consuming.

Fast forward to 2016: Ian, Richard and Adrian started a new company, Elements.cloud.

And the product they decided to build was the one they wished they'd had back at Nimbus, and knew every Salesforce customer would need. It would do all the things they did to run Nimbus, with the app doing the heavy lifting.

They wanted Elements.cloud to have a single internal platform for all customer and partner engagement. Salesforce was the natural choice as it is the most customizable, scalable, user-friendly CRM on the market. Over the last 4 years the Elements.cloud Org has been heavily customized, as it supports every area of the business.

As COO, Adrian's goal is to get the most out of the Salesforce capabilities to support the business. With Salesforce, the possibilities for customization and agility are vast, but so is the possibility of technical debt, which kills agility. The challenge is to get the benefits and avoid the drawbacks. This is the story about how we manage our own people, process and technology.

### Org Zens

Let us share Org Zen #1 from [orgzens.com](http://orgzens.com), where we invite stories of well-run Orgs. It sums up what we do with Elements, and with our organization. It was written by our Salesforce Admin, Jack Lavous:

"Having a COO who understands the importance of bottoming out requirements and processes before I start work configuring Salesforce is so important. It means I have the confidence that what I am building will be adopted by the business. But it also means development and testing is faster because I don't have constant rework. BTW He also realises that I need time to document the configuration because it will save time doing impact analysis in future releases. My title is Business Excellence. Without a clear methodology and executive support my title could so easily be Captain Chaos."



2.2

## How we drive business agility and change

## 2.2 How we drive business agility and change

This chapter is a simple overview of the stages we go through when implementing changes.

In later chapters we get into more detail about how we enable our end users to stay agile.

### First, here's a high level view of our business challenges... which are not unique:

- ✓ We are a rapidly growing startup, so we need to ensure we maximize our investment in Salesforce
- ✓ We are applying best practice implementation approaches so we can scale the business without scaling headcount
- ✓ We are concerned about technical debt, since we don't want anything to slow our speed of growth or agility
- ✓ We still need to balance agility with confidence that rapid changes won't break the Org
- ✓ We have employees in 8 locations in 6 countries in 5 time zones, with the same need to train users and drive adoption
- ✓ We have customers and partners in every time zone
- ✓ We are have both low-touch customer engagement and a managed model for the enterprise customers
- ✓ We need to integrate with other systems such as accounting, as well as our own app Elements to manage trials and licensing
- ✓ We work with partners who resell and provide consulting
- ✓ We want Salesforce to be the single repository of all company data

See the Elements Implementation Lifecycle diagram in Figure 1. The four quadrants show the activities that need to take place to deliver adoptable changes at pace, and with confidence. Although this lifecycle is usually continuous, it starts with ANALYZE on the bottom left and continues through BUILD, DELIVER and OPERATE.

We use Salesforce, Elements and complementary ISV apps to deliver the lifecycle. The list of activities below looks huge, but every one of the bullet points is required no matter how large or small the change. It appears time consuming, but it actually saves time because we seldom have to rework releases. We build what users actually need, and we don't break the Org.

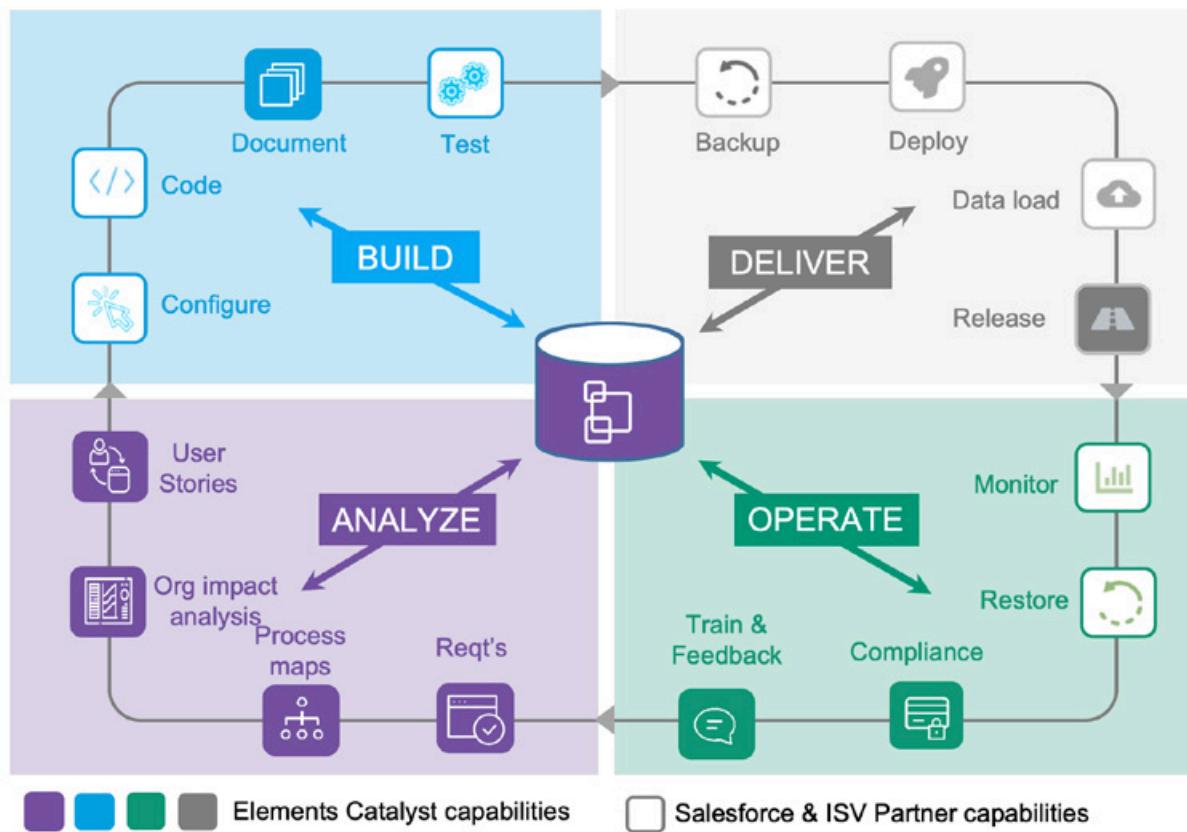


Fig. 1 - Salesforce Implementation Lifecycle

## 01. Analyze

- What are the needs and changes that are required?
- What are the business processes we are trying to support?
- What changes are needed to the underlying data model/ERD?
- What are the implications on the current Org and how risky is the planned change?

## 02. Build

- How are we going to build the functionality - clicks vs code vs external app?
- How are we testing the changes against user requirements?
- Why were changes made and where is the documentation that describes why and how we changed Salesforce?

## 03. Deliver

- What approach are we using to deliver the changes; straight into Production or through deployment pipeline?
- Do we need to upload or update data?
- What needs to be released alongside the Salesforce Org changes?

## 04. Operate

- How are users trained, then supported to increase adoption?
- How are we gathering feedback and new requirements?
- How often do we back up and what do we back up?
- What do we measure in the business?

Figure 2 is a process map of how we make changes in our Org to reflect the evolution of the business. The process is split into four phases which aligns with Elements implementation lifecycle diagram. Every activity box has lower level diagrams; 24 in total.

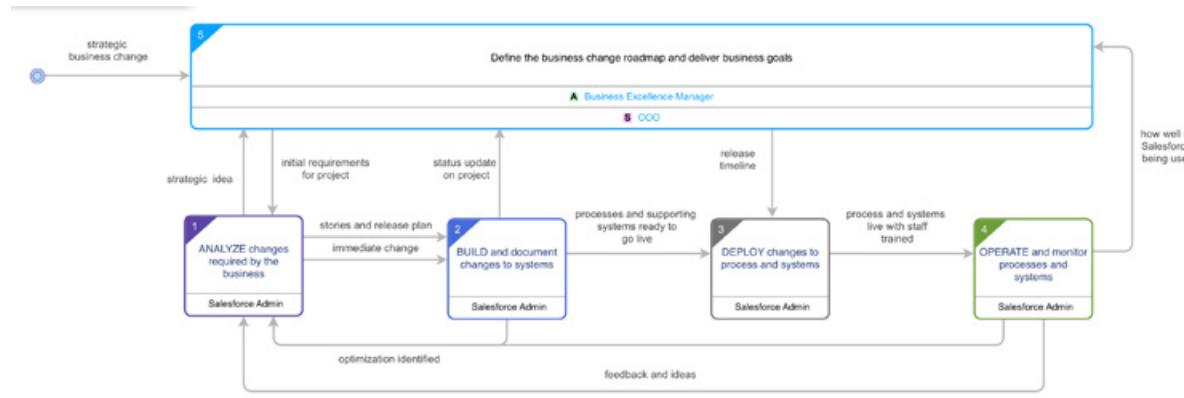


Fig. 2 - Top level diagram: Drive Operational Improvements

## How to read an Elements.cloud process map

- Elements.cloud process maps have just one shape: square activity boxes. A decision point is shown by the different flow lines.
- The lines between the boxes are just as important as the boxes. They indicate the 'why' or 'when' the activity happens and the text shows the handoffs.
- Resources are listed at the bottom of the boxes. They can be human or technical.
- A shaded corner in the top left means there is a 'child' process map with more detail at a lower level. This way, each diagram can be simpler and more readable.
- Paperclips in the top right corner mean attachments, which can be simple notes, to any URL link, pictures, related requirements, user stories, or any Org metadata item.

2.3

## Building what users really need: ANALYZE



## 2.3 Building what users really need: ANALYZE

Taking time to work out what users really want and need is critical for successful business change and Org implementation. It is also the best way to drive up adoption. Yes, it's tempting to just start building, but we are super careful not to succumb to that temptation. Have you ever started painting a room without taking the time to tape the woodwork? You probably regretted that, and ended up spending more time fixing the paint mistakes. Building your Org is no different. If you build the right thing the first time, you'll have much better user adoption.

### ANALYSE process overview

Figure 3. shows the steps in the ANALYSE phase. Each of the five activity boxes in this diagram has further detail, denoted by the dark triangles in the upper left corner of the boxes.

According to Adrian, the single most important thing is to understand the business problem you are trying to solve. This might take some digging, interviewing and thinking. The trouble with simply gathering requirements is that people usually have 'wishes' rather than 'requirements' and they can be at very different levels of detail - "We should use Einstein" vs "I want a new picklist item".

"It takes discipline and patience to not dive in to build the solution, but to go work out what solution you really need to build."

 Adrian King

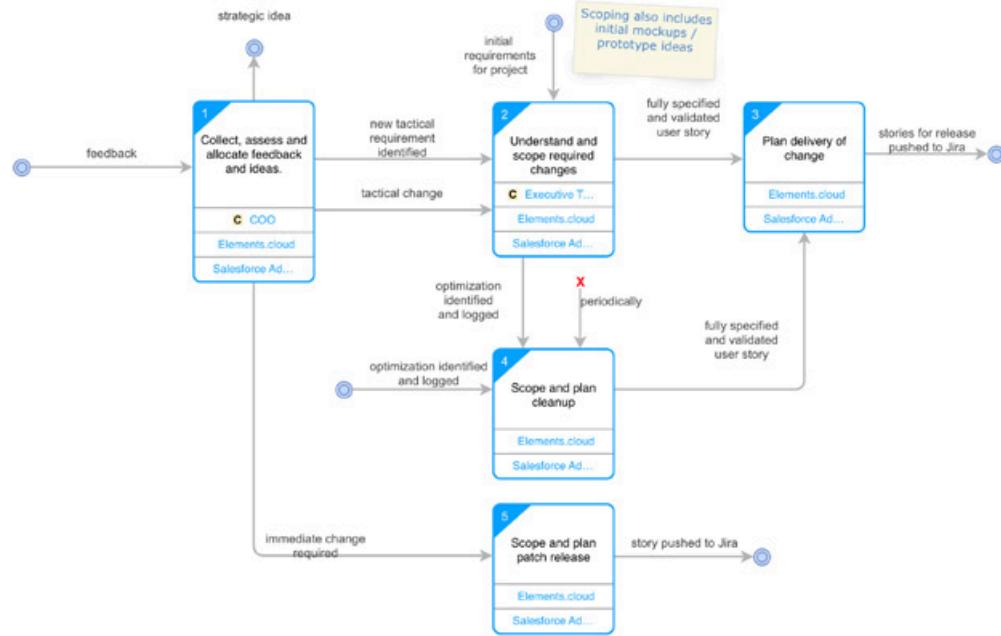


Fig. 3 - Drill down of the Analyze stage, Box 2 from Fig 2

Figure 4 shows the further drill down from Box 1 "Collect, assess and allocate feedback and ideas" in Figure 3. It includes engaging with the user who gave the feedback, then triaging all that feedback. We capture and manage all the requirements in Element.

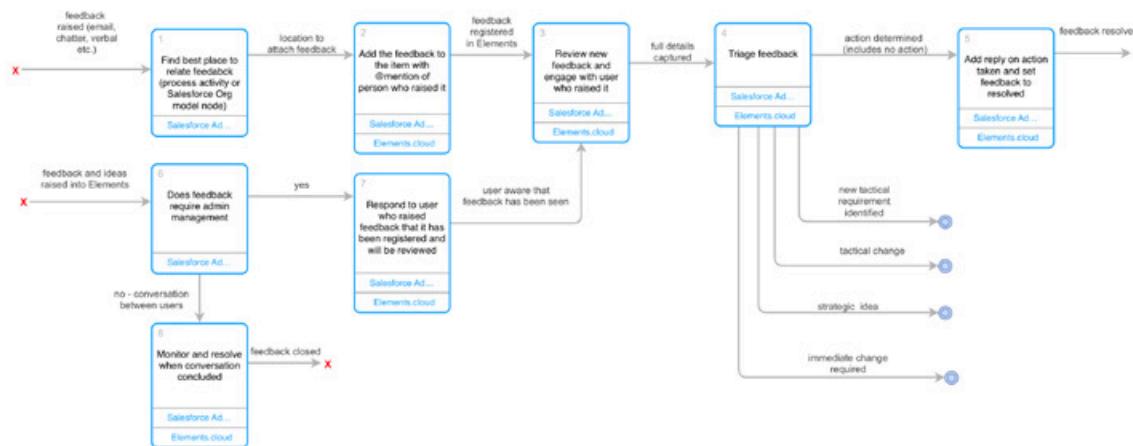


Fig. 4 - Collect assets, and allocate feedback ideas in depth

Then we move on to Box 2 of Figure 3, and the detail of this is shown in Figure 5. This is where the real analysis work takes place. Here we create really crisp user stories based on clear requirements, looking at the operational process maps and the data model.

When we look at process maps, we also update the ERD to confirm the data model changes needed to support the new processes. We find it easier to have a separate diagram with just the impacted entities (objects) than to look at the Salesforce Schema Builder, which is a little overwhelming. A little-known Elements feature is that you can draw ERDs with the process mapping functionality as well as business process diagrams.

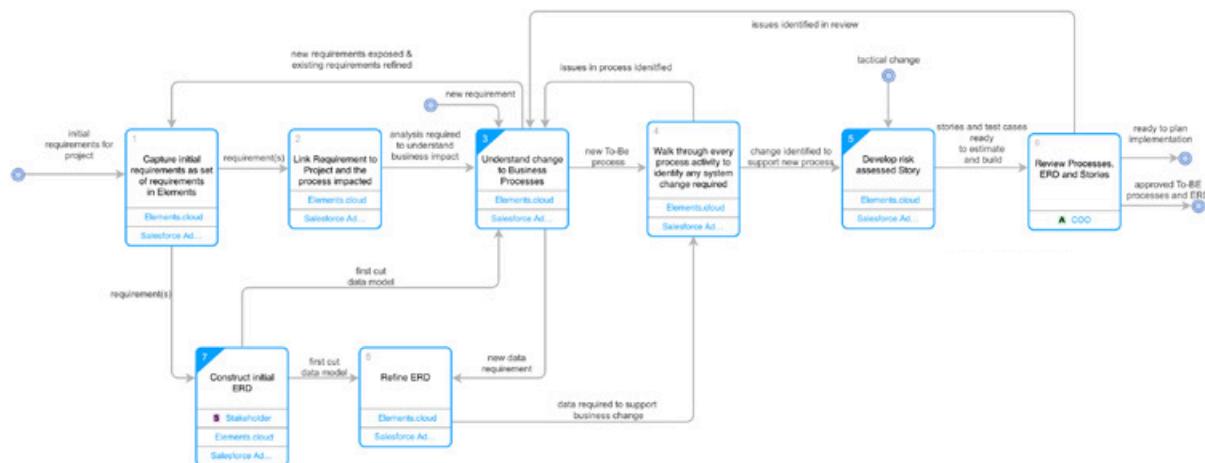


Fig. 5 - Drill down of Box 2 from Fig 3: Understand and Scope Changes

## OPS MANUAL AS PROCESS MAP

Our entire operations manual is a hierarchical process map; the activities have 'resources', and we use the process map as training. The Operations Manual process map starts with the top level diagram that describes the company end-to-end, which is developed and signed off by the exec team. Every company operational process is a lower level (drill down) diagram somewhere in the hierarchical structure. We love the Elements process mapping functionality. It's simple, intuitive and is perfect for virtual workshops.

### Creating 'Usable' User Stories

With Elements User Stories functionality, we can create user stories that clarify the requirements and processes. The user stories are in the standard format "As a \_\_\_, I want to \_\_\_, So that I can \_\_\_".

The output of the Analyze phase is user stories that can be developed and implemented. We need to get around the Implementation Lifecycle with the fewest unintended consequences, but as quickly as possible. The faster the changes get into production, the more effective the business is.

We know that Admins are told not to develop in Production, but it is the fastest and quickest way to drive changes.

We analyse the risk of every user story. If the release risk

is low, then we can safely develop in Production. If the risk is high we put it through a multi-stage development process. We build in a development sandbox, then put it into an integration test sandbox where we test against real data. Once it works, we deploy into Production.

This ensures we maintain as much business agility as possible. Let me just repeat that: This ensures we maintain as much business agility as possible. And that is what we are striving for. Many of us opted for Salesforce because of its agility. Risk assessment is at the heart of maintaining that agility. For example, we recently released two sets of feedback into Production in less than two hours.



### Assess user stories risks

We analyze every user story to assess the risk. We do this by understanding the impact of changes in terms of breaking the Org, the business, and any regulatory requirements. We then look at the complexity of the changes to the Org. The combination of the impact and complexity gives us the user story risk, as seen in Figure 6. The release risk is the highest level risk of any of the user stories in the release. From the release risk level, we decide the development path, shown in Figure 7.



Fig. 6 - User story risk

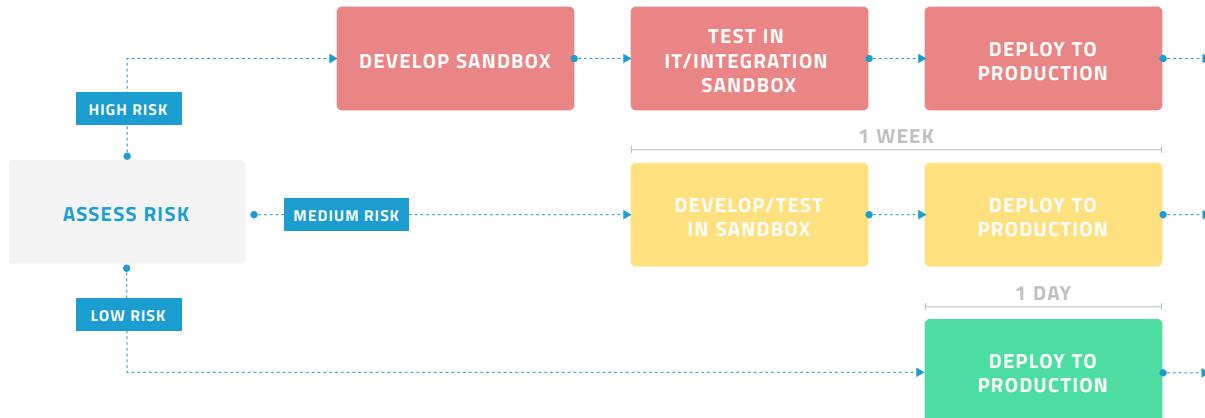


Fig. 7 - Deciding the development path

Our risk based approach would have been impossible without Elements Org Impact Analysis. This shows us the implications of making the changes to the Org. Often what looks like a simple change can have far reaching implications. The Elements Usage analysis in Figure 8 and Dependency Tree in Figure 9 are lifesavers. The Elements Org Model is a sync of the Org metadata with automated documentation and impact analysis. Our own Salesforce Admin, Jack Lavous, says;

**"The power of the Elements analysis before making any change to your Org is a vital part of any change project and should not be underestimated."**

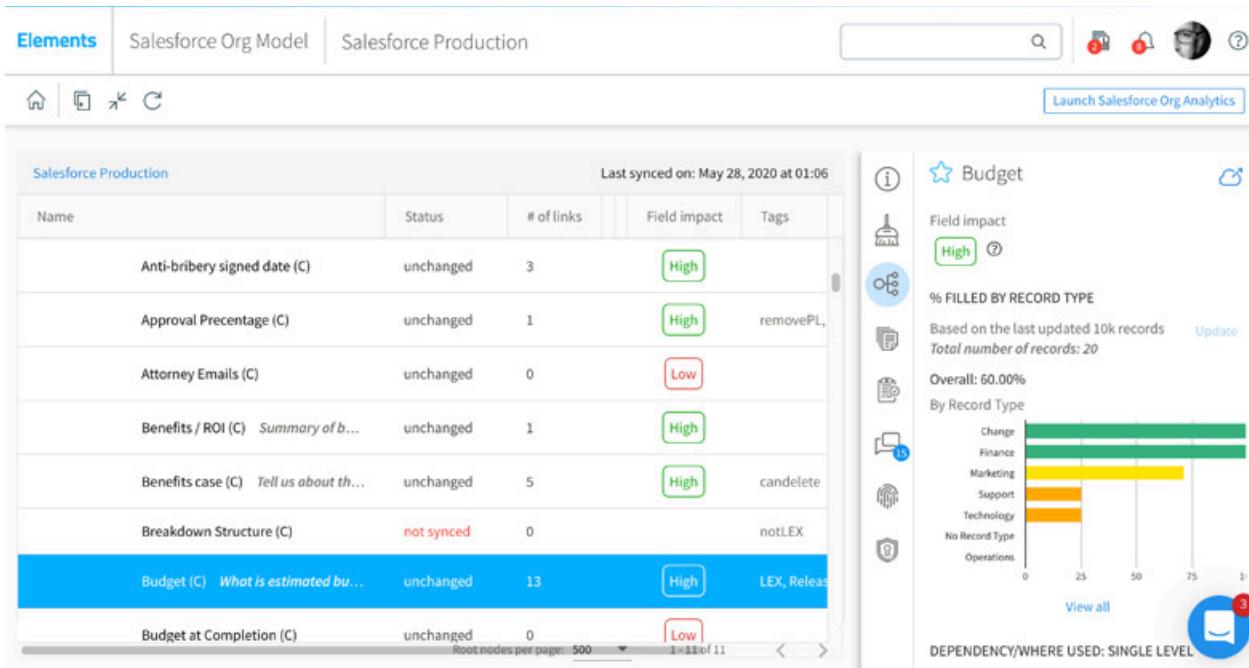


Fig. 8 - Elements Org Model showing Field Impact Analysis, Field Population and Where Used

"The biggest challenge for an Admin is to hear 'Can you just...?'. It's difficult saying no to someone senior, plus you do want to help, and the truth is, yes you can. But should you? Asking why, and why, and why again gets to the heart of the issue."

 **Jack Lavous**

Recently he was going to make changes to a single custom field which looked simple. But he ran the Dependency Tree for that field and this is what he discovered - Figure 9. The Dependency Tree is a deep dive into Field Usage so it shows you all connected Org items, and therefore the implications of making a change. His reaction, once he saw the tree  
“Thank goodness I stopped. Imagine the carnage I could’ve caused.”

## Dependency Tree for a field

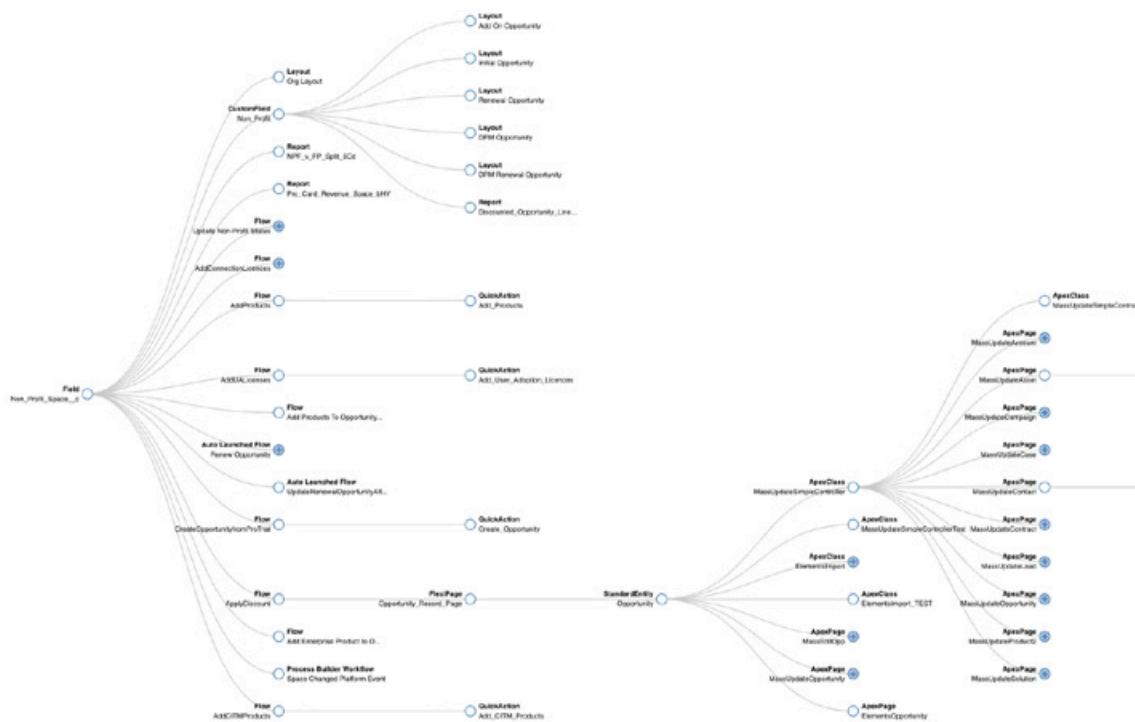


Fig. 9 - Field Usage Dependency Tree in Elements Org Model

Alongside the Dependency Tree, we also look at the documentation linked to the metadata. An implementation with no clear documentation of what has been configured – and most importantly why – means that it is very difficult, if not impossible, to support, maintain and improve. We'll look at how we document changes in the next Chapter.

“Remember that nearly every time, a change will be bigger and take longer than you initially anticipate, since the changes will interact with everything else in the Org. The Dependency Tree is a great way to push back when business users don’t understand the complexity of what they are asking for.”



Adrian King

2.4

## Build quickly, but build for long term: BUILD



## 2.4 Build quickly, but build for long term: BUILD

Remember that every Org is constantly evolving. This can be incremental changes to tweak and improve the user experience, all the way up to major change projects to implement new functionality, such as Lightning, CPQ or Einstein. All these changes need to build on the existing Org. So the first thing is to decide on how the new functionality will be built: code or declarative, what flows, objects etc.

We always create a design document so we understand fully what we are going to build. This is added as a note on the user story, and describes as succinctly as possible what is to be built. There is no template, since changes all have different aspects to them. An example of a design document can be seen in Figure 10. And not unusually, we are often running more than one release, of different risk levels, at any one time.

### An example of a design document

#### Page Layout Section - Go-To Market Strategy

- General description of competitor (150 words or less)
- Target industries (specific industries the competitor is targeting)
- Value propositions (what are the key/fundamental "values" that they are selling? could be a free text field)
- Pricing / licensing information -> free text field
- Channels of distribution: do they sell directly or do they have apps/add-ons available through appexchange, app store, google store, chrome store, etc. ; do they support cloud, desktop, and/or mobile versions of the product?

#### Page Layout Section - Updates & News

- URL field that could point to their website/ Twitter page

#### Page Layout Section - Differentiators

Would need fields (free text fields) to capture

- Product strengths (could use help text: What do customers value about the product?)
- Product weaknesses, (could use help text: What do customers dislike about the product?)
- Differentiators (could use help text: What is the difference in customer experience and offered value between you and the competitor?)

#### Page Layout Section - Customer Personas

It should be a multi-select option to associate:

- Salesforce Admin / Business Analyst
- Salesforce consultant
- Salesforce developer
- Salesforce ISV
- CIO/CTOs Enterprise companies
- Training officers / HR managers
- Process consultants

#### Need to decide where these fields go

- Quick review of fundamentals
- Key value adds
- Additional information - NOT NEEDED

#### Flows: N/A

#### Process builder: N/A

Fig. 10 - Example design document

Then it is time to build, configure and test, and our process is seen in Fig 11. This is the drill down diagram on Box 3 from Figure 2: "Build and Document Changes to System". You can see that there are different paths based on the risk level of the release.

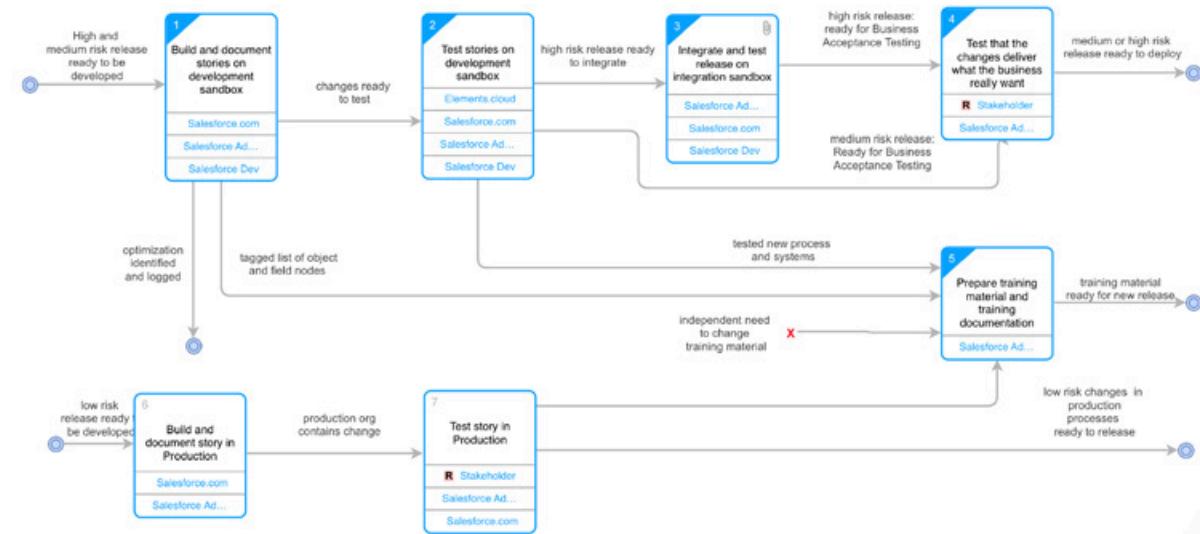


Fig. 11 - Detail of Box3 (from fig 2: 'Build and Document changes to System')

## Build configure and test headline

As we said in Chapter 3, if the release is high risk, we use a multi-stage development process. We build in a development sandbox, then put it into an integration test sandbox where we test against real data. Jack will always test on multiple profiles, something he finds invaluable. He says, "If you test on the Sys Admin profile, you rarely get test fails because you have access to everything. If I'm building something for the Product team, I log into the Product profile in the Sandbox so I can make sure I am testing correctly."

User Acceptance Testing is critically important for high risk releases. If what we have built does not support the end users' business process we will have adoption issues. The good news is we use the process diagrams we built in

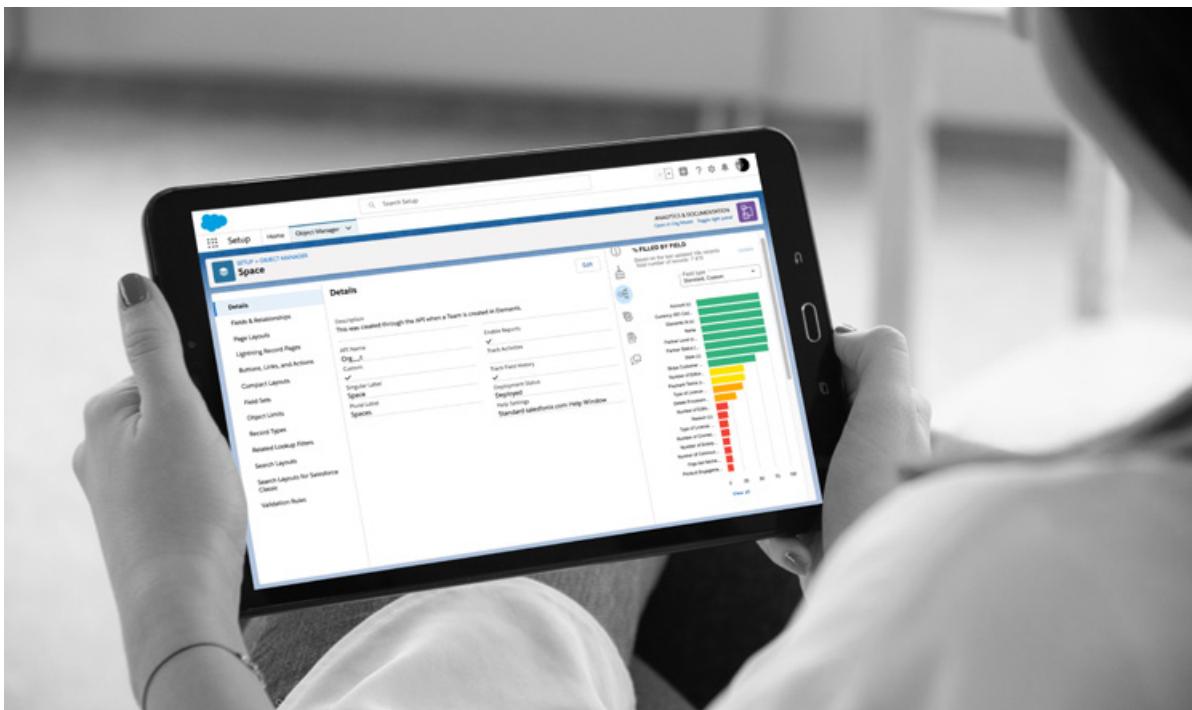
the Analyze phase for UAT. This is done in the UAT Sandbox. Once we have completed testing in integration, we deploy into Production from the initial development sandbox so we only have to create one change set.

If the release is medium risk, we have a single-stage development process, where we build in a sandbox environment, but we don't need to put it through an integration sandbox.

If the release is low risk, we build and test straight into Production. That might seem heretical, but when you can have confidence that the risk of a release is low, you can take advantage of the quicker and easier release, saving time and money. The real benefit is the end users get updates quicker.

Jack builds according to the design plan, and documents as he goes. This is something you really, really (seriously) need to do. It's easy to think you'll do it 'later', but there is no 'later.' Documentation could just be a short sentence, a photo of a whiteboard or screenshot. Or it could be a more detailed document. What is important is linking documentation to the metadata items in your metadata dictionary. We do that in the Elements Org Model which sync's all the metadata from the Salesforce Org.

An important part of documentation is to update the in-app help. The Elements Org Model allows any or all documentation attached to objects, record types and fields to be made in-app pop-up help inside record pages for end users. A neat Chrome extension extends the Salesforce help icon. So we get a double benefit. Documentation is useful for the Admin and the end user. More on this in Chapter 6.



"I see Org development as a continuous process, and I work hard to cement the habit to document as I'm building and testing. Right then, in the moment. If you've got it, attach it"

 **Jack Lavous**

2.5

## Go-live: DELIVER



## 2.5 Go-live: DELIVER

Now it's time to deploy, at last. Jack migrates what he tested in the UAT Sandbox into Production, and adds the user stories into the release notes. The new process diagrams are also released, so the new versions are published. How we do this is in Figures 12 (below) and 13 (overleaf) which make up the drill downs on Box 4, Fig 2: "Deploy Changes to Process and Systems".

After changes are deployed to Production, Jack continues to run testing against acceptance criteria on stories - according to the BDD Methodology. As an example, the following is stored in the Acceptance Criteria column of our user stories in the Elements app:

### User Story #506

#### **Test Scenario 1: Creating an affiliation from a contact record to another account.**

**Given:** A contact has a primary account they are related to.

**When:** They have an affiliation with a different account to their primary account.

**Then:** The Contact will have an affiliation created with another account with a role, current status and date assigned. The affiliation record will then show on the related list on the contact record.

#### **Test Scenario 2: Creating an affiliation from an account record to a contact outside of that domain.**

**Given:** An account has one or many contacts related  
**When:** Has an affiliation with another contact

**Then:** The account will have an affiliation record created and assigned to its related list. The account, chosen contact who is affiliated with the account will be on the record.

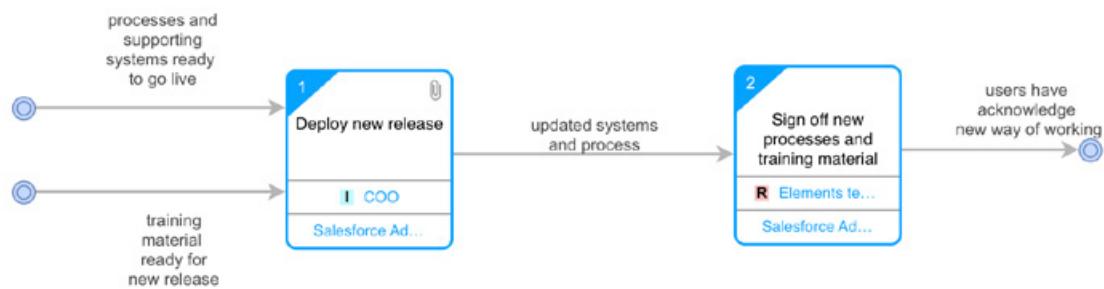


Fig. 12 - Drill down of Box 4 Fig 2

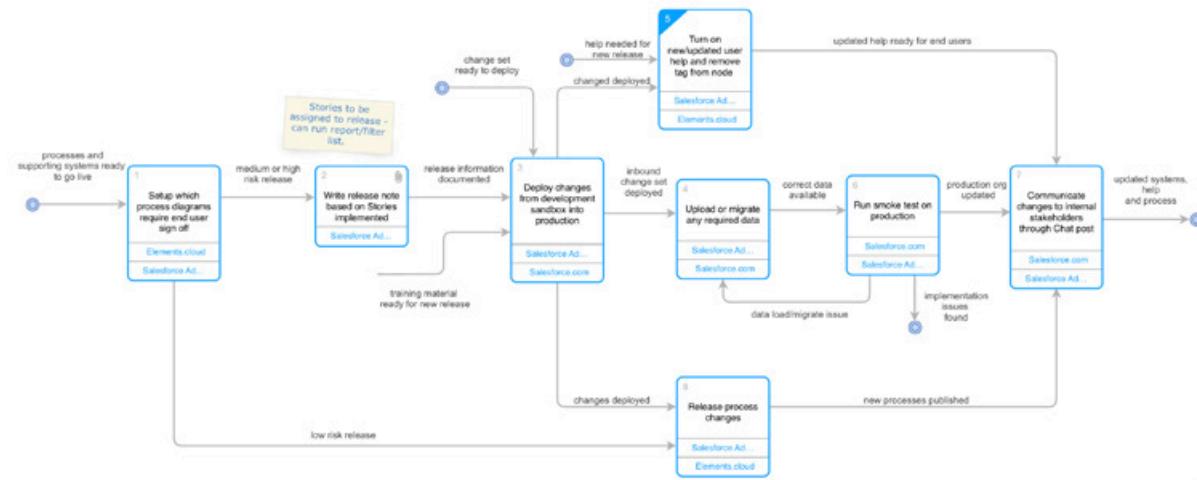


Fig. 13 - Drill down of Fig 10: Deploy New Release

2.6

## Running an agile business: OPERATE



## 2.6 Running an agile business: OPERATE

The final phase of the implementation lifecycle is to drive adoption and gather user feedback. When configuring in the Build phase, Jack creates training content that is attached to objects, record types and fields. When the process diagrams are released there may be some processes that need to be signed off by users so they're aware of the new features. The training requests are created automatically and the users are notified and tracked.

We monitor adoption and if people aren't using the new feature, we'll assess what's needed: more training, or that the change hasn't met user requirements. We also encourage end users to use the feedback button connected to every object and field on page layouts. That way the comments are associated with the related metadata item. And by the way, having a way for users to give feedback in the moment as they use Salesforce drives the next set of requirements.

We use the Elements in-app help functionality, enabling notes, URL links and process diagrams to be seen as in-app help. We follow a strict set of best practice guidelines so the help looks the same for every page layout, so a familiar pattern makes it easier for end users.

An example of our in-app help is shown in Figure 14. This example has a drop-down box on the Opportunity object with several different help topics. The object level help can be record-type specific, so users don't have to wade through long picklists of irrelevant material. The help can also be for any field - custom or standard. We've had great success with help usage. Since putting it in place in our Org toward the end of 2019, we've had a 6-fold increase in usage, and we know that because it is tracked and our team can rate each help topic.

The screenshot shows the Salesforce Opportunity page for 'zenalpha.com(ZenAlpha Operational Excellence) - Initial'. The top navigation bar includes 'Opportunity' and 'zenalpha.com(ZenAlpha Operational Excellence) - Initial'. Below the header, there are fields for 'Account Name' (zenalpha.com), 'Close Date' (10/1/2020), 'Amount' (USD 23,500.00), and 'Opportunity Owner' (Ian Gotts). A progress bar at the top indicates the opportunity is 'SDR Qualified'. Below the main content, there is a 'Contact Roles (0)' section with 'Add Contact Roles' and 'Edit Contact Roles' buttons. The main content area is divided into 'Details', 'Chatter', and 'Process' tabs. The 'Details' tab is active, showing a dropdown menu for 'Primary Opp' with three items: 'Opportunity Name', 'Account Name' (zenalpha.com), and 'Space' (ZenAlpha Operational Excellence). The 'Process' tab contains a process diagram with nodes like 'SDR Quali...', 'Nurture', 'Sales Qua...', 'Sales Eng...', 'Enterprise...', 'Triangulat...', and 'Pro...'. The 'Chatter' tab is empty. On the right side of the page, there are sections for 'Stage' (SDR Qualified), 'Close Date' (10/1/2020), and 'Probability (%)' (20%). A note in the process diagram says 'USAGE: account dashboard'.

Fig 14 - Page layout of Opportunity page

An important part of our Operate phase is that we periodically select a part of the Org to review and capture items that need cleaning up. Every Org accumulates technical debt, and this is another thing that kills agility. The Elements Optimize capability is a place to tag and track clean up items. We can report on them and group them

together logically, and create one or more user stories to do the cleanup project. Optimizations can be identified at any time; feedback from end users, identified whilst developing a user story, or noticed during the Build phase.



## ISO27001

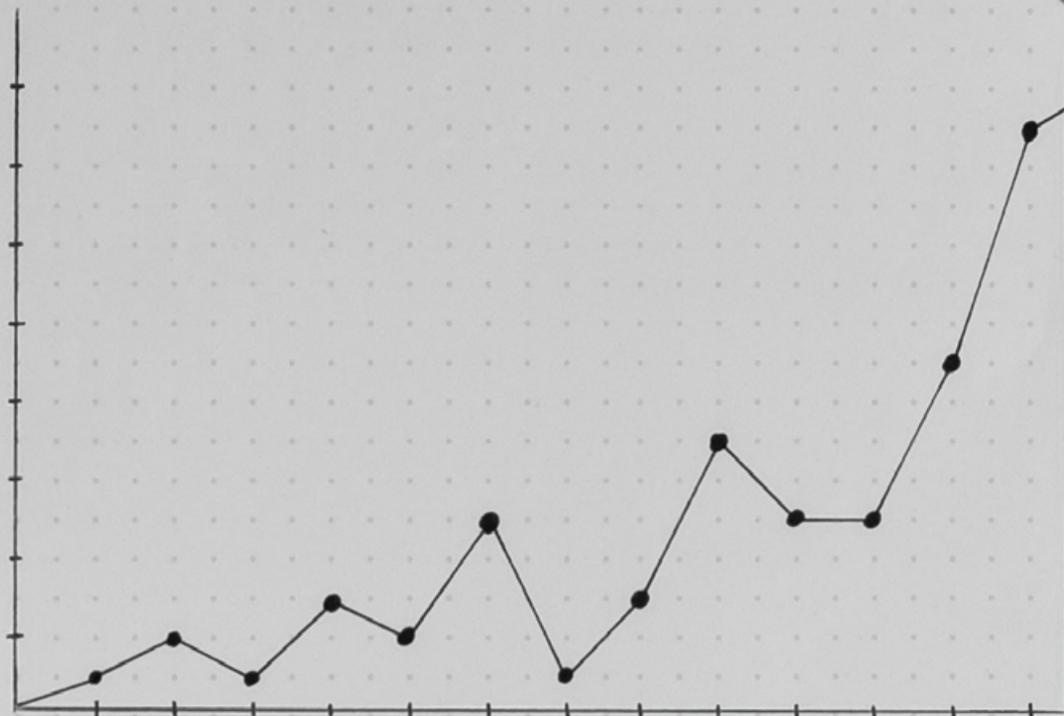
ISO 27001 is an information security standard. It has requirements for keeping info security under management control. To be accredited requires a strict audit, as Adrian King says;

"Being ISO 27001 certified means that we have to demonstrate that we have control of our software development process. Using Elements means that we not only do we understand what we do, but we are also constantly refining and improving the process and when it comes to our audits the auditors are always really impressed with the way we are clearly in control"



2.7

## What are the results?



## 2.7 What are the results?

The measures of success are speed of the implementation lifecycle, and user adoption. We need the different areas of the business - marketing, sales, customer success, finance - to be agile and responsive. On average, we are now closing 40 change requests per month. The reason for that is over 50% are low risk, so Jack feels he's keeping up with the rapid changes happening in the business.

We have been seeing greater and greater adoption of Salesforce. With Elements, feedback is easy, so users tend to give more feedback, and since we have a good system for implementing that feedback, users increasingly believe that feedback will create better processes. This is a virtuous cycle.

*"Now, more than any time in my business career, we need to be agile and able to respond to changes. We are in a great position as we have a solid foundation of business processes, a well documented and architected Salesforce Org, and a world class implementation methodology."*



Ian Gotts, CEO

*"When I arrived, we didn't have an adequate sales process for selling into the large accounts. We had to set up that infrastructure in Salesforce, as well as a complex workflow through which we manage opportunities. The faster we could get that process implemented, the faster we could start creating campaigns. Creating those is not a one-off, though, and ongoing agility is vital. Basically, changing the engine while in flight. We involve Jack in sales meetings so he can truly understand our needs, in order to build the right things - not just build them fast. Plus adoption of fast lifecycles needs in-app help and documentation. There's no time for instructor-led training. Success = time to value + agility + adoption. I've been working with Salesforce since 2002, and seen at least 40 Salesforce Orgs close up. There is no Org on the planet that is run better than ours."*



Rob Brown, VP Strategic Accounts

"One of the great things about using Elements as part of how we run the business is that we can use process maps as an Operations Manual. It's a dream to capture processes as we invent them - then not constantly reinvent the wheel. I can change and add to them, and when they are released everyone will see the latest version."



Alyssa Abbey, VP Community

"Shortly after I started, very green and still doing basic Trailhead modules, I was introduced to Elements Org Impact Analytics report which has a set of metrics. I started a cleanup project on our own Org. I did one hour per day for the first four weeks. Basically it was one bite at a time. Watching the metrics in that report turn green was very satisfying. It seemed like such a big job at first, but once you lay the foundation, keeping on top of it is not that difficult. Normally, documentation and clean up is at the bottom of people's priority lists, but they don't understand the pain of not doing it."



Jack Lavous, Business Excellence

"The excitement of joining a new company is rarely on par with learning the corporate processes and completing training. Having lived through several blackhole cloud data experiences, onboarding with Elements was a tutorial of simplicity being the ultimate in sophistication. Succinct, intuitive, and available ON THE FIELD in Salesforce.com. I was trained in the first week."



Conrad Aach, Account Exec

"When I first started working in our Salesforce Org, the in-app help was a lifeline. Having some resources and instruction definitely sped up my workflow. Both that, and the uncluttered page layouts has made finding the information I need at any given moment incredibly easy."



Solly Breakwell, Finance Assistant

"I've worked with scores of Salesforce Orgs over the years, and what I like about our Org is its closed loop mechanism that reinforces an agile methodology. We move from Feedback to Requirements, Stories and Process Flows - then to changes in Salesforce with corresponding contextually sensitive documentation for developers and end users."



Ed Schlesinger, VP HigherEd

"As a marketer, the cleaner the data in Salesforce, the more value I can add to the company. I can pull up any contact or account and have vital information available as help, which gives me context before I run campaigns. Furthermore, the feedback functionality means that marketing can be in constant contact with admins, execs, sales and product - eradicating silos."



Nathan Hale, Marketing Associate

The final word needs to go to Adrian King who in his two roles - COO & CTO - is both customer and supplier.

"I have a vision of one set of data that we run our whole business around, and Salesforce is the platform we are using to achieve this. To deliver on this vision, however, requires clarity on what the business actually does, its processes.

1. how changes impact what is done today
2. how they interrelate and are coupled
3. how and why we have configured Salesforce to date, with insights on the dependencies;
4. and really importantly, supporting the users in actually getting the value from the new capabilities we introduce.

Elements is enabling us to do this by giving a 360 degree views to all stakeholders."



Adrian King, COO & CTO

**3.1**

# About us



## About us

Elements exists to make work faster, easier and more valuable for millions of people. As a team, we are focused on providing the approach and access to robust, elegant, affordable software, delivered in the cloud at scale.

### The founders

The founders worked together as part of the executive team at Nimbus which we grew from a startup in a bedroom in a small fishing village in the UK in 1997 to a global software company that had an enviable client list including 10% of the Fortune 500. Clients such as Nestle, Sara Lee, BP, Chevron, Barclays, Northern Trust, AstraZeneca, Novartis, and local and central Government. In 2011 Nimbus was acquired by TIBCO and the executive team disbanded, however in 2015 the 'band got back together' and formed Elements.



**Ian Gotts**

As CEO of Elements.cloud, Ian is passionate about providing the tools and training to help customers reduce the risk and improve the adoption of their Salesforce implementations. Ian is the author of 10 books and a prolific blogger with a rare ability to make the complex seem simple. This makes him a popular and entertaining speaker who challenges his audience to see things differently.



**Adrian King**

With a background in R&D, Adrian is an experienced software entrepreneur, strategy consultant and interim CEO. He provides strategy consulting to a number of businesses with an emphasis on leveraging IT and optimizing operations.



**Richard Parker**

Richard has led Sales and Consulting globally with some of the largest enterprises in the world, including Nestlé, Orange, Chevron, AstraZeneca, Novartis, Toyota and Central Government. Richard is an author of Common Approach, Uncommon Results and is an experienced conference speaker.



We'd love to hear from you.

[elements.cloud](mailto:elements.cloud) | [success@elements.cloud](mailto:success@elements.cloud)

**North America**

[San Francisco, US \(HQ\)](#)

[New York](#)

[Toronto](#)

**EMEA**

[Petersfield, UK](#)

[Geneva, Switzerland](#)

[Maastricht, The Netherlands](#)

[Lviv, Ukraine](#)