



CHANGING VENDORS

Streamlined Onboarding:
How product owners can
prepare for a smooth transition

01

MadeCurious.

The need for change

Change is never easy. But when you're armed with the right knowledge, the path to achieving change can be smooth and predictable.

A change in vendors can take place for many reasons. The department might need more resources than the vendor can supply, or a change in strategy may require new capabilities. There might be a lack of alignment or you're unsatisfied with the level of service and commitment.

If you're considering a change of vendor, first define what you want and need from an alternative.

It's helpful to consider these in the form of measurable outcomes:

- Improving the values match with your department
- Improving contractals (simplicity, precision, effectiveness)
- Improving (or matching) performance to the levels and areas you need it
- Improving the ability to measure the service, e.g. using a service level agreement (SLA)
- Higher levels of support

Preparation for transition/onboarding

You may already be under pressure to start the process, but it's important to resist the urge to dive in. Instead, take time to plan and prepare for onboarding your new vendor.

The benefits of preparation include:

1. Increasing the speed and efficacy of onboarding - thinking now pays back later.
2. Spotting potential problems early reduces risk - collaborate on what-if scenarios.
3. Setting the scene before any undesirable defaults have crept in - start as you mean to go on.

Setting expectations

Clarity is key. Set out your expectations for your new vendor in the clearest possible terms, for example:

- Key timeframes - and context for why these are important.
- Go-live dates - so all parties involved know their part for key releases.
- Expected levels of support and service including SLA terms and conditions.
- Lead by example - model the behaviour you want before undesirable defaults have crept in.

TIP: If an SLA is being contractually enforced, consider giving your vendor three or four weeks grace to familiarise with systems and execute handover with their predecessor.

If your expectations are clearly understood, you can begin handover preparation with your new vendor. Remember to make sure that any relevant subject matter experts (SMEs) can be available for consultation with your new vendor before you begin.

Finally, and most importantly, invest some time in helping the new vendor understand your context, goals and constraints and if you have one, your long term roadmap. A vendor that better understands your world is better equipped to provide the right service and care.

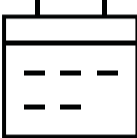

Handover Plan

Collaborate with the new vendor to build a clear picture of what should happen over a defined timeframe e.g. 4 weeks and detail key dates, key personnel, a schedule of actions and activities.

It is also useful to include incident expectations for events during the handover period and consider relaxing them until the new vendor is up to speed.

The handover plan might detail the help desk process for your own team, especially if it is different from the previous arrangement so everyone is clear on what to do and when.

Example schedule included in the handover plan:

	
WEEK 1	<ul style="list-style-type: none"> • System access to relevant systems and accounts i.e. Silverstripe CMS • Analysis & Discovery • Agency staff start logging new tickets • Liaise with Jane Doe and Joe Bloggs to resolve tickets • For any level 1 & 2 tickets that are dependent on the incumbent to get to resolution, the team should email or call Jane to coordinate. jane.doe@gov.co.nz or 021 0213 333
WEEK 2	<ul style="list-style-type: none"> • Analysis & Discovery • Familiarisation efforts/investigation • New vendor to collate a list of questions/requirements that remain unanswered.
WEEK 3	<ul style="list-style-type: none"> • Testing of work on the WIP tickets for familiarisation (not formal testing). • Provide a definitive list of all outstanding handover requirements.
WEEK 4	<ul style="list-style-type: none"> • Product owner will review the remaining requirements list and work with the incumbent to obtain the relevant information. • Product owner to provide all (or as much of) the information requested and flag any gaps.
WEEK 4	<ul style="list-style-type: none"> • BAU from 30th July onwards - normal support resumes.

Prepare information

Organise what you know

Is there an existing backlog of features or known issues?

If so, start collecting and collating as much information as you can find on the backlog(s) for the new vendor to receive. Information about the status i.e. unresolved, in progress, resolved. Share backlogs' early, especially known issues so that the team can prepare and learn about the typical type of support required.

If there is in progress development or remediation work with the current vendor (the incumbent) prepare for each ticket/feature/fix:

- any information specific to deployment (e.g. pre-deploy tasks) or configuration to happen post deployment (e.g. site config)
- instructions for how to smoke test this feature/fix
- information/updates for teams or users (e.g. if specific people need to know that a feature they were waiting for is now available)

If there are tensions or relationship difficulties with your incumbent vendor, make sure the new vendor is aware so they can conduct onboarding in a sensitive way and follow your guidance on how to engage with the incumbent (in some cases you may not want the new vendor directly communicating with the incumbent).

Support history - context is gold

Providing a history of the support and maintenance of the system is incredibly helpful to the new vendor, particularly during contract negotiations. Without any understanding of the amount, type and frequency of support and maintenance tickets submitted over the last year (or more), the new vendor will have to best guess estimate the right level of support allocation which is inherently prone to issues i.e. the monthly managed services charge is higher than necessary because the vendor estimated higher volumes of support. Examples of typical support requests can be really helpful to the team so have a think about what sorts of things have cropped up in the past and how much support has generally been needed?

In alignment with the outcomes you are looking to improve with a new vendor, it is helpful to consider what parts of support you would like to see better handled and any pain points you would like to resolve.

To make your expected level of service explicit and as measure as possible going forward, it is therefore important to provide information on:

- Number / Frequency of support tickets
- Type / Range of support requests
- Average times to fix
- Averages for the number of P1, P2, P3 tickets
- Examples of support requests

Armed with this information the new vendor can model the right balance of support and ensure the right resources are aligned.



Documentation

Is there any existing documentation about the use or purpose of the system, its features or configuration? Be clear on who needs to maintain the relevant documentation and consider providing the vendor with access to the source if they are to maintain documentation.

If there is no documentation available, think about what information you can provide so the vendor can create their own project documentation. This goes for both technical documentation as well as end-user guides.

TIP: Never underestimate the power of a diagram. If you don't already have one, draw a picture or diagram overview of how you see everything hanging together, and walk your new vendor through this diagram together, noting questions or gaps to come back to. This diagram will be particularly helpful if there are external touch-points, stakeholders or integrations.

Services & Integrations

What services does the system comprise of, and what else does it integrate with?

The new vendor will need access to most of them. Consider access for code repository (repo), hosting, database access, deployment processes, any 3rd party / external services, monitoring/logging, and analytics.

Confirm that the incumbent will agree to an approach for sharing environments and deployments.

Testing

- In what ways is it okay for the new vendor to use the production system for testing (if at all), e.g. can they have dummy users to test the functionality?
- Are there any limitations to testing in UAT e.g. can it send emails to real users from that environment?
- Confirmation of any smoke testing steps you want added to the standard release process.



Security

What are the security practices and policies the new vendor will have to align with? Information pertaining to the following will be very helpful:

- InfoSec
- Code of conduct
- Privacy
- System Access
- Downloading Data

Data

It is crucial to share information on your preferences and requirements for data i.e. data retention timeframes.

- Privacy requirements
- Size of data and application to be migrated
- Restore frequency
- Sensitivity
- Analytics

Historical Change

Consider whether there have been any kind of historical changes in the system - this might be large feature additions or data migrations for example.

Other pertinent historical information might be:

- Any adverse events
- Security breaches
- Privacy issues
- Significant system failures

Context and insights that have both positive and negative impacts for the system or its users will help your new vendor understand sensitives and preferences.

Armed with knowledge about where extra caution should be applied or improvements that were a great success for example, will have a bigger impact than you might expect.

Certification & Accreditation

Certification and Accreditation (C&A) is growing in importance, and if you haven't yet already been involved, you very likely will soon.

Certification and Accreditation is a fundamental governance and assurance process, designed to provide the Board, Chief Executive and senior executives of government agencies confidence that:

- Information and its associated technology are well-managed
- Risks are properly identified and mitigated
- Governance responsibilities can demonstrably be met

If the product or system you manage has already started C&A, then you will already have documentation and information which will be helpful to share with your new vendor. This might also include information the incumbent vendor was previously asked to provide in the form of evidence or reports. Of particular importance is the product's Assurance Plan, which details the relevant activities that need to happen across the year to maintain accreditation.

It would be very helpful to share the product's Assurance Plan as early as possible with the new vendor to allow them to plan and prepare for their contribution to this important security governance.

If no previous C&A has been completed on the product, but you are aware that it will start in the near future; advise the new vendor and provide any/all the information you can, especially timeframes for requirements. Here is a useful guide to C&A we have written that may help prepare your new vendor.

TIP: If you're still selecting a new vendor, consider that a sign of quality is a vendor that already has experience and knowledge of C&A and can help guide you through it as part of their services.

Your Team

Provide details of the team involved including roles/responsibilities and what should and should not be communicated to those people.

Roles and responsibilities are especially important to help our teams understand who to turn to for information or decision making.

Tasks the new vendor will need to prepare:

Shared Knowledge

Create the Project Summary in your teams shared workspace (we use confluence) with a general overview of project, including important people, and definitions of important domain specific language such as:

- Who are the users of the system and what are they called?
- What are their roles called?
- What unique words/terms do they use to describe the different aspects of the system?
- If any aren't clear, suggest clearer terms to aid communication

Code repository

A code repository, or repo, is a centralised digital storage that developers use to make and manage changes to an application's source code. Developers have to store and share folders, text files, and other types of documents when developing software. Access the products repo and set it up locally. Work out if there's:

- existing git flow i.e. branch/PR conventions
- versioning convention,
 - ensuring it's documented in the readme

- add usual tools:
 - pull request template;
 - .editorconfig file;
 - linting;
 - .gitignore;
 - add a docker setup for local development

Deployment Instructions

Communicate any times of the year where production deploys are avoided i.e Christmas, Easter, possibly during elections.

Create a deployment release plan in a shared workspace and record:

- The deployment process, including how the state of features are tracked
- How versioning is done
- Deployment steps
- Record any restrictions on deploying to specific environments (e.g. prod deployment windows)
- Typical post-deploy smoke testing
- Who needs to approve deployments
- Who needs to be notified for specific events

Once a deployment page has been created, a new vendor should provide a copy and ask for your feedback.



Change management

Preparing your own teams for a change of vendor might entail some thoughtful change management if there is a difference in process to the incumbent. It is also the perfect opportunity to introduce desired change if improvements need to be made.

Identifying any pain points with this team before you switch, enables you to take the opportunity to implement change and process to relieve such pain. It is also advisable to work with your new vendor to create process training for the staff that will be submitting tickets that details the information required and any tool training.

If teams are likely to work together closely/interact a lot with the new vendor' then it will be beneficial to get them working together sooner than later - if they don't know each other, they are less likely to work well together off the bat. It could be helpful to arrange a meet and greet with some ice breakers.

CWP or Silverstripe

When then product is CWP or nz.silverstripe.cloud specific a couple of extra steps need to be taken:

- Need access to SS Cloud (deployments/management etc) which includes access to the gitlab repo.
- The different access roles are explained here - <https://servicedesk.nz.silverstripe.cloud/support/solutions/articles/75000042794-roles-and-permissions>
- Organise someone with Deployer permission (usually DevOps + lead developers).
- Ideally everyone will have Deployer permissions but if that is not possible, the Developer role permissions allows access to the code and to test/UAT environments but doesn't allow any access to Production (deploy, download snapshot etc).

Checklist

INFORMATION CHECKLIST

- Backlogs
- Support History
- Documentation
- Roles & Responsibilities
- Services & Integrations
- Testing
- Support
- Historical Changes/context
- Support
- Team
- Data
- Security
- Change Management Process

TECHNICAL CHECKLIST

- Repos
- Deployment process
- Accounts/Logins