

Approaching a User Experience & Unified Interface Transition

Applies to:

Dynamics 365 Sales
Dynamics 365 Service

Whitepaper

Summary: This business-oriented white paper outlines the planning, governance, and management principles to consider when approaching a user experience change within a custom business application on the Dynamics platform. This white paper focuses specifically on Dynamics 365 applications adopting the Unified Interface, but many topics are applicable to any user experience update.

Writer: Hayley Bass (Microsoft)

Content Reviewers & Contributors: Matt Menzer, Roger Gilchrist, Rick Prologo, Amy Taricco, Henry Jammes, Janelle Aberle, Adrian Orth, Praveen Kumar Singh, Lance Delano, Filip Karadzic, Christopher Moncayo, Mark Spilde, David Yack

Published: June 2021

Copyright

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2021 Microsoft. All rights reserved.

Microsoft, Microsoft Dynamics 365, PowerApps, are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

Contents

Introduction	5
Purpose of this white paper	5
Scope of this white paper	5
How can I get started?	5
Positioning the Unified Interface	7
Work wherever you are	7
What is the value of Unified Interface to the business?	8
Design approach best practices	9
Importance of simplicity	9
Agree on common guidelines	10
Design for the user not the manager	10
Value-based design	10
Sweat the small stuff!	10
Understand our UX and align to supportability	11
Document as you go	11
Designing role focused experiences	11
Design for the specialist role	12
Design for the business user	12
Understand your business user	12
Power of observation	13
Limit distractions	14
Business process design	14
Efficiency: Focus on insights not data	15
Measure outcomes not actions	16
Adoption	17
Data	20
Governance and guidelines	21
Communication	21
Delivery model	21
Setting guidelines	22
Embracing change in the right way	22
Measuring success	22
When should I be thinking about measurement?	23
Internal/External facing metrics	23
Dynamics 365 – UX usage recommendations	25
Navigation and dashboards	26
Interactive dashboards	28
Form design	29

Review your data structure.....	30
Multiple forms.....	31
Reflow	31
Tabs	31
Optimize for create and consume experiences	32
Quick view form.....	33
Capitalization.....	34
Field type selection: Option sets vs lookup fields	34
Timeline Wall control.....	35
Grid selection	36
Editable grids	36
Subgrids and associated grids	36
Utilizing views.....	37
Composing Applications.....	37
Handling multiple apps.....	37
Business process	38
Quick create	39
Dialog deprecation	41
Custom icons (grid and entity level)	42
Handling web resources and iframes in Unified Interface	43
Client-side scripting.....	43
Custom theming	44
Custom Help and Guided Tasks	45
Design example	46
<hr/>	
Summary.....	49
<hr/>	
Appendix.....	50
<hr/>	
Sample Employee Satisfaction Survey Questions	50

Introduction

The evolution of the Microsoft Dynamics 365 applications brings substantial architectural and functional changes to enable digital transformation. The front end for this new user experience, [Unified Interface](#), is built on responsive web design principles, which provide optimal viewing and interaction for any screen size, device, or browser orientation. With a move to make Unified Interface the default experience for end users of Dynamics 365 apps, many customers are taking the opportunity to review their current implementations and optimize workflows to reflect changes in the business.

Purpose of this white paper

This white paper is for the business administrator/sponsor who is responsible for the planning and execution of front-end changes in the end user experience and how they are implemented for different roles within the organization. The goal of this white paper is to provide some practical steps for planning a move to the Unified Interface and where possible, we will help you to achieve best practice and added value for your users to ensure successful deployments and high productivity.

The key objectives therefore are as follows:

- Key requirements and actions to take when planning a move to the Unified Interface
- Guidance on how to approach user experience changes and successfully land them within a business

The guidelines within are also more broadly applicable for organizations looking to perform a design refresh or approaching a project for the first time. This document isn't intended to explain the technical details of the Unified Interface components. Instead, it's a collection of best practices based on real life customer scenarios. Further information about the Unified Interface can be found on our Documents site located [here](#).

Scope of this white paper

Unless specifically noted, all features mentioned in this white paper are available as of June 2021. Business guidance and best practices are based on real life customer scenarios and situations. It should be used as a starting point to adapt to your own organisation.

The following topics are out of scope:

- Technical deep dive into the Unified Interface components
- Technical performance tuning of the application
- UX feature guidance on applications outside of Microsoft Dynamics 365 apps

How can I get started?

We recommend you read this document in its entirety to help plan for a design refresh or new implementation. However, if you are looking for just a few quick pointers on transitioning to the Unified Interface specifically, here's a basic snapshot of the key areas that should be planned and executed.

Initial Review	<p>Take some time to gain a good level of understanding on what is currently in place from a technical and business perspective.</p> <ul style="list-style-type: none"> • Spend some time reviewing the goals for the system and the metrics agreed to validate successful adoption. • Get familiar with the roles in the system. • Stakeholder reviewers: Identify a group of users within your organization that can help with guidance as you move through the transition. • Run Solution Checker, using the Power Apps checker PowerShell module, against your deployment to understand any potential compatibility issues that could impact a move to Unified Interface.
Start now	<p>Get started right away with a test environment or app showcasing your current deployment running in Unified Interface. This will help you to quickly identify the areas that require further assessment.</p>
Analysis	<p>Review the test environment:</p> <ul style="list-style-type: none"> • Test for errors (automated/manual) to spot problem areas that may need re-work to function correctly. • Identify error patterns to help provide the bigger picture of areas that need work. • Identify if there are any known parity or functional gaps that could be blockers to adoption. Think about new guidelines or workarounds to be messaged. • Review migration challenges that need further work, such as the dialog process which is deprecated in the Unified Interface. • Review the user roles against the updated screens to see if they still function as designed or have become sub-optimal for Unified Interface. • Consider refactoring into multiple apps where appropriate - this is a good time to do the review of what should be an app.
Re-work	<ul style="list-style-type: none"> • Remove areas that you don't need • Tackle anything that is required for compatibility. If there are parts of the system which simply do not function in the new interface, you should target these first to ensure you have a working environment. • Once compatible, take the opportunity to review and update any legacy design decisions or layouts that are suboptimal for Unified Interface. This includes potential display changes within dashboards. • Take advantage of any technology updates and switch on if they will provide positive usability and productivity. • Take time to review how your application looks across devices. • Have you provided the optimum layout for re-flow?
Monitor	<ul style="list-style-type: none"> • Monitor the results of any changes you make and provide tangible metrics on the business impact. • Get reviewers to look at the changes and provide feedback.

	<ul style="list-style-type: none"> This is a good opportunity to identify any areas that may require deeper re-work to reflect changes in business process or a change in customer behaviour.
Re-imagine	<p>This area could be combined with re-work depending on your project scope. A move to the Unified Interface is a great catalyst to engage with the business and refresh key areas of the system to reflect any changing work patterns.</p> <p>Taking advantage of new capabilities like Role-Based Views (App Modules), AI, and other functionality to make a positive user impact. Any changes should be quantified and measured.</p> <p>Consider a pilot with a small set of users who provide feedback as the re-imagined experience is put together</p>
Release management (Initial, iteration and evolution)	<p>Understand how and when you want to introduce the Unified Interface to your business audience and build a plan.</p> <p>Consider:</p> <ul style="list-style-type: none"> Releasing by role to enable teams to get on board faster Providing access after basic re-work to gain feedback and iterate with strong feedback channels Remove the legacy web client access once trained. Unified Interface applications can run in parallel to Web Client for comparison, but all new applications are Unified Interface only. It's advisable to get users acclimated to Unified Interface and remove Web Client access. Providing ongoing updates to innovate and refine the deployment to align with our release roadmap and your business.

Positioning the Unified Interface

Organizations using Dynamics 365 applications already have access to Unified Interface. To provide background, the term Microsoft Dynamics 365 refers to a grouping of business applications including Dynamics 365 Sales and Dynamics 365 Customer Service. Any new applications will be utilizing the Unified Interface, and this is the strategic direction.

These business applications are built natively on Microsoft's Dynamics 365 customer engagement platform. They utilize a shared Common Data Service for data storage and core platform services. Unified Interface fits into this as the design and interaction client across all access points of the applications including browser, tablet or phone. The underlying design is across all our first party applications such as Sales and Service and individual and enterprise custom applications we call model-driven apps. See illustration for a summary:

Work wherever you are

Expectations for engaging with work life have changed. Individuals now expect to have the flexibility to conduct work and personal business anywhere, regardless of device, time, and environment. That could be a laptop at their desk, a tablet/hybrid device in a meeting room, or using a phone while on the train.

Natural transitions between devices is very important. Unified Interface forms are fully responsive and re-factor depending on how the user chooses to view the application.

Full details about the Unified Interface can be found [here](#). You will find any new applications have Unified Interface automatically enabled. For an existing implementation using the web client we have published instructions on how to [enable across your existing implementation](#). Take time to review the changes within a test application prior to full production enablement.

What is the value of Unified Interface to the business?

Making a change to anything the user interacts with is often perceived to be a large effort that may cause friction to a business. Often, we see initial reluctance to get started. Finding the value in making this change is therefore critical to receive user buy-in and evangelism. There are, however, many reasons a user should be delighted to move to the Unified Interface. The fresh new look is not only very usable and natural in terms of navigation, it is also designed with accessibility front of mind. Users should see speed improvements across the system and be able to quickly get up and running regardless of access point with the common design pattern reducing the need for lots of training.

As well as design there are many productivity enhancements introduced that a user could take advantage of. Some examples include:

Timeline control - The Timeline wall helps users collaborate with their team by tracking customer communication in a record on a single page in an easy to read view. See everything from posts and voice attachments, to emails and notes. It provides a quick way to see the entire communication thread of an email and reduces clicks when looking at a snapshot of recent activity. For a business user this is likely the first place they will go to see what has happened to a specific record of an activity.

Business process enhancement – Business process flows are a great way to guide a user through getting their work done, providing a visual indicator of the steps they need to complete. The business process flow component has been improved by the docking mechanism to the right-hand side of the screen. End users can now dock the business process stage on the screen to help them stay focused on the task at hand. This is especially useful when a stage in a process includes complex steps.

Reference panel - Use the reference panel for apps built on Unified Interface like Dynamics 365 Customer Service. The reference panel is a great way to get work done without clicking away from the screen. Users can look up things like Knowledge Base articles within the context of the record they are viewing.

Navigation - The new menu options let you swiftly navigate between the different apps in the system. Users get quick access to recently viewed records and pinned favorites. Reviewing the right navigation options for your user coupled with providing focused role experiences allows the user to only see the information they need and can eliminate confusion.

Reflow - The app also scales by reflowing the components on the screen. The responsive design adapts to your environment based on screen size, so the more available space that you have the more information can be displayed. This is great for business users who may switch between devices or an organization that has users covering a span of different screen resolutions.

Controls - Introduction of a variety of different dials and visualizations help transform forms, dashboards, views, and homepage grids. Although this is a technical feature it has a positive business impact. These can also be set for browser, phone, or tablet to provide the optimum experience for the user and a touch-based alternative. An example could be a slider to set a monetary amount on an opportunity screen. More details can be found [here](#).

Focused Role Experiences - Administrators, makers, and end users are always looking for ways to simplify systems. Reducing ‘noise’ from other roles/departments allows a user to focus on the key elements specific to their job role. Unified Interface experiences are based on model-driven apps, which means the application is generated automatically from the database. You can scope the user experience based on the tasks a particular set of users need to accomplish and have single or multiple business apps based on their requirements. Users can just see the information they need to see for any given role and can easily switch between apps as needed.

To achieve real business value these capabilities should be showcased in context with the business needs and requirements. Spending some time with the users to understand your deployment today and where improvements could be made in parallel with a transition to the Unified Interface is where the true value lies. However, this should not be a blocker to get started quickly. Some of our best deployments have got it in the hands of the business as soon as possible to gauge feedback and then worked in quick update cycles to make changes to optimize for the way Unified Interface is designed coupled with the business change demands. Take advantage of the ability to [create custom business apps](#) by role and move your users over in logical groups. Often the momentum will take hold the moment one group starts using Unified Interface and evangelizes to others. We will cover some tips and guidance in approaching a full user experience transition as part of this white paper.

Design approach best practices

Dynamics 365 application projects can follow any one of the many methodologies available. There are multiple variables that affect the impact and success of an implementation. Irrespective of the methodology followed or the differences in play, there are some common traits across most successful deployments where user experience is at the center. In this section, these general best practices for approaching a start to an implementation or a re-design will be covered. This is particularly important if you are planning a transition to Unified Interface.

Importance of simplicity

Dynamics 365 apps contains a set of standard forms designed to give organizations a starting point on how the application could potentially be used. The temptation is to jump straight in to change and add new components on top of the standard design. We often see challenges in this approach when the deployment finally gets in front of the business users. Expectations versus reality can be misaligned.

Keeping to a simple design ethos not only helps with growing the system alongside our product roadmap to reduce custom work but also has a significant impact on user adoption. A complex screen without user value is likely to have poor data quality and often lead to the user filling in the ‘management minimum’. All organizations have nuances which may require configuration or customization but before delivering these, pause and ask the following questions:

- Could the standard functionality achieve the required outcome for the user?
- Could you conduct an early feedback session with business users to evaluate how close you could keep to standard?
- Are there any elements of the standard or if you are already live, the adjusted form that do not provide tangible business value to the user and customer?
- Can you take a business requirement and break it up into deliverables that can be measured before increasing functionality?
- Do you have a design principle agreed with stakeholders to align to our capabilities and roadmap where possible over custom design?

Deciding what to remove is as important as agreeing what functionality to include. Review the forms and functionality with simplicity in mind. This might mean adding role-specific applications, forms, and quick creates to strip unneeded data for each role. This is far better than holding back on role-based applications in favour of creating a simple maintenance experience for the Admin/Maker.

Equally, don't be afraid to take away unnecessary data. For example, the standard experience is designed to give organizations a perspective on how the system could be used. That could provide too much functionality for a user depending on role. Review this alongside your business requirements and if in doubt, remove the component and monitor the impact.

Agree on common guidelines

If you have a team of individuals working on the same system whether they are internal or external there are often nuances between how someone may interpret a specification and produce the feature. Differences could include the way a form layout is approached, theming, column widths, etc. to the type of field or graphic visualization selected. Spending time to gain agreement on a common set of rules across the system will pay dividends when users look to adopt the solution. Adding new areas will then not seem such of a leap for a user if they understand the basic construct of how to get around the system and enter data. Of course, you can have applications for different roles with different forms, views, and dashboards but to reduce maintenance and confusion for individuals that may move across roles it is sensible to have a common baseline in place. This is also referred to in our governance section as good practice.

Design for the user not the manager

Look at your existing screen designs and divide them up between the content that is valuable for a user in their role and the data that needs collating for management. If the balance is more in the favor of management then it is a clear indication that re-work is required. Ensure the system doesn't feel like a tax and instead delivers useful insight and optimizes for input as well as analysis. Always question any new field, features, and sections added to your application – is it going to help improve your customer's experience or the user's efficiency and insight?

Value-based design

How is this new capability delivering operational excellence or improving my customer's goals? This links strongly to the above point about putting the user and customer in focus. Users should feel driven to log into the system because it is telling them something they may not have known about their accounts and contacts. Value could be different depending on role and so be careful not to assume one size fits all. Salespeople look for competitive insights, trends to help close deals faster, and insights into how to build new relationships. However, a Customer Service agent may look to understand common issue threads and troubleshooting information to serve a customer quicker on a call. Fitting value around a persona will help focus the design – see [Designing focused role experiences](#) for more detail.

Sweat the small stuff!

Often it is the small areas which cause the most unrest amongst a user community and can also be easiest to resolve. When you conduct feedback reviews you will often hear about the small design changes that a user is focused on. Plan those in and deliver quickly if they make sense. User advocates are extremely important when introducing a new design or piece of functionality. Showing you have listened and understand the value of a collection of small elements which address user concerns will help build trust for when larger items need addressing.

Understand our UX and align to supportability

All too often we see organizations commission a specialist UX design agency that doesn't have knowledge of the Dynamics 365 platform to come up with a fresh new design for implementation. The designs are often detailed and cover every custom-made element a user may wish for in a system for that moment and then it gets handed over to delivery to execute. This is fraught with danger as one of two outcomes occur:

- Follow the design to the letter and build and customize a custom-made system which looks exactly as requested. However, this likely incurs a tax of heavy customization which needs on-going maintenance and could cause a potential lack of flexibility should the business decide to make a change or adopt new standard innovations following platform updates.
- Disregard the UX design and provide a system that doesn't meet expectations and therefore could cause poor adoption rates.

Consider instead:

- Combining UX design alongside experts in the technology to ensure what is proposed is optimum to maintain and provides a solution to meet business needs today and in the future.
- Setting expectations correctly with the business on the benefits of moving to a supported platform. Custom-made is possible but the benefits of adopting a standards-based approach allows for faster change, easier maintenance and future-proof innovations.
- Keeping in mind the purpose of the user experience. Often needs can be met as effectively using the product versus a completely tailored experience that was conceived due to lack of awareness of how the product could achieve it.

Have someone on the delivery team always looking at the next innovation to showcase to the business. This will help users realize the benefits of moving to a supported platform and still get their business requirements met.

Document as you go

This doesn't have to be a 500-page manual. With the rate of changes within a business that would be impossible to maintain. It is important to ensure you have captured detail on a new process or business-related feature and its intended usage. This will undoubtedly help training teams when they have to formulate materials but also to keep track of the reasoning behind the feature, how it relates to business value, the metrics supporting it, and the business team sign-off. This is particularly important if you have transient resources on a project where information could be lost. Commenting within the code to capture a summary on the technical behaviour is also sensible so that should any issues arise later it is quick to troubleshoot with the context on what it was supposed to do.

Designing role focused experiences

With the exciting introduction of the [application designer](#) for Dynamics 365 applications, it is now very straightforward to create a separate application for a group of business users to support a function or role without having to write code. Consider it to be simply a filter on the full client targeting just the areas a specific group of people need to interact with thus reducing noise and navigation confusion.

Before jumping in to create a new application or deciding whether to just use the out of the box applications, it is important to understand the user personas, focusing on their goals and motivations. Take time to identify the processes and insights that will be key to each persona and the flow they would typically work through to get their job done. The most successful implementations we see look at this aspect in detail before deciding how many role/persona-based applications they may create. Think about the following design decisions.

Design for the specialist role

Most implementations have specialist and generalist roles. Good UX design is seldom generic. It's advisable to design experiences optimized for each specialist role. Optimizing experiences for the generalist role makes it hard for the specialists to use the system, thus affecting adoption cycles. It's important to check for this pitfall right after the solution design phase but also acknowledge there is a fine balance to be met.

Creating multiple applications is simple to execute from an IT perspective but there should also be a business discussion on whether similar personas should consolidate into using common ways of working to be able to report in alignment. The decision will ultimately come down to the amount of compromises each role will need to take if they did merge into one application. It is important to have these discussions upfront and gain business stakeholder agreement. We have seen these conversations occur with organizations that have acquired other companies and now have a group of Sales and Service roles that have different processes for legacy reasons. It is a good time to review and refresh those before applying that design to applications.

Design for the business user

When you are looking to refresh an existing deployment or build an area from scratch it is important to always consider the business user and the impact to their daily flow of activities. A common challenge is the difference in how advanced developers or testers use the system and how business users (the primary target user for the system) use the system. Business users and technical users have different goals and ways of working, which often lead to very different UX optimizations. The business user is the person the system should be optimized for to ensure maximum efficiency. One way to mitigate risks is to make low cost prototypes and test them on real business users before finalizing designs. This can help reveal usability issues early and build trust.

Understand your business user

There are many ways to build up a good picture of your business users and how they operate. Building a persona document, journey map which augments a series of user goals and actions into a visual story timeline, and day in the life scenario document will help to identify the touchpoints they need to go through to serve their customers and surface the business data they need to know. Persona documents and journey maps are constantly evolving so it is sensible to have a regular cadence in place to review and update these as the business changes. These should then be reflected back into your design to ensure your application is relevant and useful for the teams.

Persona documents typically contain the following information:

- **Role summary** Who are they, what do they do?
- **KPIs** What do they care about/paid on?
- **Needs** What are the must haves in the system to help them do their job?
- **Motivations** What makes the user tick? What is important to them?
- **Expectations of system** for example, it must be available offline
- **Limitations** Pressure points for the user, such as time
- **Pain points** What do they struggle with inside the system?
- **Context** Do they have awareness of ask/customer or would it be new each time?

When looking to build out a journey map look at the end-to-end processes and break them down into typical milestones. Ensure that you review this through the lens of both the internal business user and the customer. A journey map typically contains:

- **Journey description** (e.g., Lifecycle of a customer)
- **Stages and substages** (e.g., Acquisition and prospect discovery)
- **Touchpoints** (Where the customer and internal business user interact through the process)
- **Feedback from staff** What you heard from them in an interview (Likes/ Dislikes/Wishes by stage)
- **Review of emotional journey** How do they feel currently about that specific area (e.g., unsupported, self-driven). These are triggers that could cause negative adoption if not addressed.
- **High-level pain points per stage** Summary of biggest areas of concern in this stage
- **Data and system:** What key data must they capture and in what source?

Day in the Life

Scenario documents are then the next level of detail. Take a typical day plan from the user and document the following:

- **Action type** e.g., Preparing to onboard a new client
- **Details on what they are doing** e.g., Research, creating a new record, review research materials
- **Systems** Where they go to execute the detail
- **Pain points**
- **Frequency** of how often they carry out the task
- **Typical location** e.g., Internal/external on phone/tablet/laptop/other

Review your deployment against this information to help refine the flow and enable the right data to be captured and support the user.

Power of observation

This is particularly important if you are looking to update your system and try out new designs. Interviewing a user will reveal good insight into their wants and needs based on knowledge of the business roles they work in. What it may not reveal is how they execute that within the system. Physically sitting with individuals to observe their behavior will help to spot gaps in the UX design and allow you to identify features that may excite them. This could even build to a small prototype which allows you to make changes based on your observations and get feedback before evolving to a broader group. Observation is a very powerful way to see through the lens of the user. It will help to identify design patterns which are successful, and which need refinement to suit what the user is trying to achieve. What you may think as easy or obvious may not be for the user. Decisions can then be made on whether to provide additional training or whether there should be a change in the design.

Many end users guide design based on their experiences of previous systems, not on full understanding of the art of the possible with a new system. Observing their experience adapting to new processes can reveal insight that may not have been realized during initial design as they discover new ways of working.

It is also worth noting that often good insight will combine observations with a structured interview after the observation exercise has finished to gain further insight into what is happening and why.

Limit distractions

As you start to map the information you have identified about your business users into the system try to limit the amount of data they need to see. Too much information can lead to user confusion and cause unnecessary form hopping and scrolling. It is always best to remove from the display anything you are not sure about and then evaluate with the users if that piece was critical to their success with an intent to add it later.

There is typically a fear in initial design of the consequences of leaving something out, leading to large datasets to capture. That fear often leads to an overburdened design for fields that may never or rarely be used. It can be useful to identify edge cases and design for them separately, so they do not distract from more common usage patterns such as only making them visible as needed.

As mentioned, simple design equals greater efficiency and ultimately satisfied users and customers.

Business process design

Business process flows allow a user to follow a pre-defined guidance path to ensure data is captured in the right way. It can force a user to follow linear steps or allow them flexibility to jump from stage to stage without losing focus on the overall flow. They can also help a user navigate a specific set of actions without having to know which area of the system they should be in, thus reducing training.

Business process flows can also automate actions, move across entities, trigger follow ups and much more without technical assistance. When reviewing your system design, consider adding in a business process to help your users move through their daily activities with ease. Some typical examples for business process flows could include:

- Managing a sales opportunity to help capture qualification data, manage price approvals, and conduct service aftercare
- Opening a new business account within the banking sector
- Managing an event from creation to launch within Microsoft Dynamics 365 Marketing
- Handling a complaint procedure against a case
- Reviewing a patient care plan within healthcare
- And many more...

Business process flows can span multiple areas of the system, for example accounts, contacts, and opportunities (entities). You can run more than one process against any record in the system. This is particularly useful if you have different users handling the same record for different tasks. An account manager, for example, may need an account review process to help guide through a yearly account plan and a compliance user may require a process on the same company to ensure they are adhering to any compliance regulations. Both processes can run on a single account record and can also be deployed just to certain role-based applications for specific users to see. If you have included them already within your deployment, make sure they are reviewed at regular intervals to ensure they align with any changes with the business. Some key questions and comments to think about if you are looking to review or build new process flows:

- What do you have documented already on your business processes?
- Do they still reflect the current process today?
- Is there a process you need a business process flow for?

-
- Are there any gaps which cause inefficiency for the customer and the internal users? This could be down to key information not being captured at the right point in the customer cycle. Equally we see this when a customer has an action which internally requires multiple department interactions and the flow between departments is not handled well in the system.
 - Assess the gaps and the pains points. What is the business impact if the process was not altered? What are the benefits of making a change? (Suggestion here is to look at deals which have been lost due to a gap in the process – due to competitive engagement, lack of business unit visibility, team collaboration.
 - Can any elements be simplified or automated?
 - If you are using multiple processes for different teams, for example a sales process for Product Team X versus another sales process for product Team Y, look for common areas, naming conventions, and terminology. This can help with overall reporting later.
 - How would the teams like to change the process? Observe and interview your staff to make sure the application is providing value to the end users.
 - Define which users need to see which processes and apply the correct access. This will reduce confusion on which process a user needs to follow.
 - Ensure you understand the guardrails of business process flows to decide if they are a true fit to your requirement.

Define a true vision for what ‘better’ might look like. The temptation may be to jump immediately into redesign. But we suggest that you take the time to document your existing process flows first alongside the business. There are two reasons to do this task. First, if your team represents multiple functional business areas, it is important for them to develop a solid understanding of the parts of the process that are outside their department to agree on common areas. Second, to take the time to surface the problems in your current process.

Technology won’t correct fundamental flaws if there are basic problems with the way you are interacting today. You need to ensure that issues are identified and corrected in your redesign efforts instead of being inadvertently repeated. Get some testing in place once improvements are mapped into the system to get feedback before a broader rollout.

For additional Information:

- Business process flows [overview video](#) (Notice that the screenshots in the video are using the legacy web client but the content is relevant today)
- Documentation: [create a Business process flow](#)

Efficiency: Focus on insights not data

We have talked a lot in this white paper about the importance of having an efficient system for the user. All features and data capture areas should be intuitive to find and follow a pattern to make future areas feel familiar to navigate. Have common design guidelines to support this effort (see [Governance and guidelines](#) for more detail).

Efficiency is also about the ability to gain insight quickly to enable faster decisions and actions. In recent years, data has been seen as an ultimate solution to problems that businesses face. Terms like “big data,” “data driven decision making,” and “artificial Intelligence” are used often in business discussions to help gain competitive

advantage and enable user efficiency. A business might be sitting on a treasure trove of data but unless they can figure out what the questions are that they want answered, the data is useless.

The most successful business application designs focus more on the insights and on the actions based on insights, rather than showing raw data. Some examples of insights follow.

What help can we drive?	What goes wrong?	How can the user be more effective?	How to ease discovery?
<ul style="list-style-type: none">•What do users not know about their customers or actions?•What would change their interaction with customers?•How can I understand who else knows my customer?	<ul style="list-style-type: none">•What is frequently done in error?•What causes negative results, such as customer satisfaction or cost?	<ul style="list-style-type: none">•Where does time or efficiency get lost?•What behaviour can we drive to get to the right outcome first time?	<ul style="list-style-type: none">•How to formalize organizational memory?•How to discover what others are doing to avoid duplication?

Let's look at an example – “What do users not know about their customers or actions?”

Consider a retail bank named Contoso Bank and assume that financial fraud is the largest business concern. Now consider the situation when a customer calls the support desk about a wrong entry on their credit card statement. The wrong data to show to the support agent would be the number of support tickets raised by this customer in the last 30 days. Or the customer's last 10 transactions. If the key objective is to identify suspected financial fraud, the data to be shown here is all cases related to wrongful credit card charges in the last month from the customer's household. If this number is greater than a specified threshold, the system should issue an alert for suspicious activity. As you can see here, we are moving from static status data about the last five cases to data that is more likely to be actionable. Asking the right question helps to ensure that the system is designed to address the key organizational objectives.

Measure outcomes not actions

When redesigning areas of the system, the focus is often put on reducing individual actions, rather than measuring the efficiency of the overall outcome for the user and customer. This can lead to reduced actions for the user. However, this isn't guaranteed to drive efficient outcomes. It's important to keep this in mind while you decide what to measure, which will determine if an implementation is successful.

Calculating the number of clicks to achieve a task is a measure of navigation efficiency, not a measure of user experience. This isn't to say that the number of clicks isn't important. However, it is important to look at the big picture when making decisions on user experience.

For example:

- If you could increase first call resolution for service calls by 50%, would it be worth adding two more clicks to the process?

-
- If you could increase your opportunity win rate by 10% by adding four extra clicks, would it be worth adding them?

In both cases, if your main decision point had been based on a measure of number of clicks to achieve a task – an action – your answer would be no. However, if you're measuring the effect on business outcome, it would be a resounding yes. Take the service call scenario, each *initial* call would be two clicks more. However, the *overall* number of clicks, when considering the complete lifecycle, would be lower as it reduces the amount of time you need to call back the customer for follow up. This broader view, not only of the complete scenario but also of the overall business benefit, is important.

Adoption

You can have the best software in the world, with the most sophisticated features, analytics and integration, but if people don't use it, it isn't going to add value. Often planned at the last point within a deployment lifecycle, good adoption is crucial to see how successful your changes have been. Although a move to the Unified Interface is not a vast difference to the legacy Web Client, it is definitely a change and therefore an adoption plan should be formulated to make sure the transition is as smooth as possible, especially if you have taken the opportunity to refresh forms, processes, and flows within the application. Remember, features are rarely the driving force behind successful user adoption. There must be tangible business value to land a new user experience well with your staff.

Adoption should never be seen as a one-time goal at the beginning of a launch. Adoption is something that must be monitored and tracked constantly so that changes can be made as needed.

Here are some of the common adoption concerns we hear that all should be addressed as part of a typical project. Starting and continuing to address these concerns with users will prevent negative business impact.

Project

No clear ongoing executive sponsorship = why should the user care?

Lack of clear expectations on data entry, system scope, goal

Training time and ramp up

Resistance to change

Feedback never implemented

Slow to see improvements

Perception

Seen as System of Record

Command and Control mechanism for management reporting

System is a user tax

Lack of value to user

Doesn't match real world

Business Impact

Low quality data as users are filling in the absolute minimum to drive reporting

Can't drive business change

Low ROI

Lower on-going project investment

Low personal value = Low adoption

Potential missed opportunities and revenue

Poor data means falling behind on digital transformation plans and technology innovations like artificial intelligence

Successful adoption is achieved when you start to put the user and customer at the center of the system. Prioritise any improvements to the solution only if they can be linked to a tangible customer experience or business user benefit. Management functions should be secondary. Inevitably adoption will go up if the user spots value by making them more efficient. The quality of data can then drive better business decisions, ROI, and future investment in the solution.

It is an excellent idea to get a pulse on user opinion of your existing deployment before you start to work out what changes you are going to make and how to launch those into your ecosystem. Conducting some focus groups or even a survey will help you understand the top areas of concern and may also help you prioritise which parts of your solution you may want to refresh first for greater adoption. For some employee satisfaction example questions, see the appendix.

Here is a collection of common adoption ideas seen in our top deployments.

Content availability	Delivery	Refresh	Reinforce
<ul style="list-style-type: none"> •Doing something face to face often received well in first instance •Think about e-learning library of key processes so users can watch back (keep short!) •Can you devise one pagers on key processes? Could they be broken down by skill level? Quick steps/Mid-level detail/Complete description •Think about embedding within the product (KB or custom Help with interactive pointers and guided tasks) 	<ul style="list-style-type: none"> •Make the content available in manageable pieces – Attention spans •Content could differ by New Starter/Existing User •Can users/ Peer Champions and Executive deliver any content rather than trainers? Shows in it together. 	<ul style="list-style-type: none"> •Users are likely to forget something they do not do often. Remember after any formal training they need to put it into practice. •Provide materials to support ‘reminders’ on key business processes (Business & Technology) •Think about a ‘did you know’ tips section within a newsletter or other format to keep them learning •Those users struggling with IT changes may appreciate some 'How do I' sessions to ask questions. These are sometimes called drop in clinics and could be calls/face to face •Don't forget onboarding materials – often forgotten but very important to set new people on right path 	<ul style="list-style-type: none"> •For each role ensure they have Unified Interface only accessible rather than option to switch back to the legacy web client. Any issues should be logged rather than having them switch from one experience to another causing confusion •Straight after any formal training set scenario walkthroughs prior to go live to give them a safe environment to practice what they learned! This is important if you have a gap between go live and training •Utilize managers and Peer champions to showcase best practice usage •Ensure management conducts team reporting and recognition via the system where possible

Projects like this are not an IT function. This needs to be a partnership with the business to ensure the right changes are prioritized. Adoption programs should also be a combined effort:

Champions	Community	Feedback
<ul style="list-style-type: none"> • Business and Management Champions who are ‘go to’ people within the organization and are part of not only the ongoing process but also provide feedback through development cycle • Must be subject matter expert of the business and how it maps to Customer Engagement • Management equally important – champions need to be at all levels • Suggest a pre-day of training for managers to secure sponsorship but include in general training in addition 	<ul style="list-style-type: none"> • Build an ecosystem of people who can answer questions throughout the business on the system and the process • Think about internal community sites like Yammer/Teams to manage queries and reward those users who answer questions for others without the need for support (Get the management teams to also participate) • Think about user led videos/snippets that showcase best practice • Is there anything you can add to gamify and make adoption fun? 	<ul style="list-style-type: none"> • Engage the users to help improve the business processes within the application • Reward ideas that have been utilized • Think about business change groups to reflect changes in real world. The application should not stand still! • Be visible on improvements made based upon user feedback – show you are listening!

As a key takeaway: It’s very important to focus on how all elements of the application supports the customer and business users’ day-to-day activities to improve efficiency and deliver value. If this shared purpose isn’t established early and the focus isn’t enforced through design and implementation decisions, poor adoption and overall project failure will likely follow.

Data

We should never underestimate the value of quality data. When approaching a review of your system, include an evaluation of the reliability of your data. Poor data quality has a significant business cost and will also have a considerable impact on the user’s productivity and ultimate trust of information. If you also have a desire to adopt the latest technology innovations such as artificial intelligence, high quality data is fundamental to see trusted results and allow reliable decision making. Better data also enables more accurate targeting and communications within the marketing department, provides reliable reports using Power BI, and ensures you are on the right side of any compliance regulations where data accuracy and respecting user preferences is key.

Data impact can certainly be easily measured as part of your user experience project. Building a governance process to manage and maintain the data is an important step forward. There are also sensible steps you can take within your application to also ensure data integrity.

- Look to provide business guidance within your adoption plan on how information should be entered into the system.
- Think about where you can help the user by supplying consistent option sets within a form to reduce the need for manual entry. This will also speed up data entry which is great for a user.
- Constantly review and fix duplicates and poor data. This is something that needs constant monitoring to keep confidence levels high and should have sponsorship from your business to ensure everyone takes the right level of ownership to keep records clean.
- Look at some of the form recommendations later in this white paper as quality data is often linked to form efficiency. Remember unless a user sees value, they will likely fill in the minimum amount of information and always opt for default options on fields they don’t understand.
- Communicate ways that quality data delivers benefit back to the end user. If they can see the direct benefit, it can be a motivation to keep the date accurate. For example, having a phone number in the

system makes outbound dialling from synced contacts easier, having case/opportunity details means managers can check on the state of a customer directly in the system reducing out of hours calls for urgent status updates.

Governance and guidelines

A successful deployment must provide the functionality that the organization's business requires in a way that is manageable, supportable, and sustainable by the organization's delivery team. It is an ongoing dialog and partnership between business stakeholders, end users, IT deployment program teams, and executives to ensure the business and customer needs are met by utilizing the strengths of the platform. To keep this balance working effectively it is important to invest in defining the right guidelines to keep all parties with the right expectations.

Communication

A cornerstone of any governance model is a communications plan to your business and stakeholders on the project. A feeling on inclusion will help adoption and keep expectations grounded. Our most successful projects have had a strong communication pillar across every delivery milestone and feature enhancement.

The communication method can vary depending on the audience, but it should follow a series of templates to keep continuity and familiarity to the recipient.

Some common communication templates:

- Program strategy & boundaries are articulated by executive leadership and communicated to business users to set expectations on value of the project and visibility.
- Launch communications – whenever new functionality is introduced it should be communicated to the business. Content could include: Details on the problem it is trying to solve, the solution how to guide and the expected benefit. Multi-media communications with quick to watch walkthrough videos, images or business stakeholder endorsement content are always a good way to grab attention.
- Provide communication on agreed delivery plans. If you are planning a regular cadence for updates, make sure the business is aware when they are due and the impact it may have on them. It is then very important that you stick to them to build trust and communicate to the group when they are coming and once again when they have launched.
- Business workshop meetings – whenever the business meets with delivery teams it's important to document the findings of that meeting for the group to agree and sign off. This reduces confusion later and starts to build a partnership contract on any changes with a deployment.
- Have a template for capturing tips/tricks guidance. This then should be slotted into existing communications channels to keep the deployment front of mind.
- Communicate your change guidelines. For guidance, see [Setting Guidelines](#).

Delivery model

All communications about enhancements to the business should be underpinned with a predictable and reliable delivery model. No business user would be pleased to hear about a new feature coming and then find it was not delivered within the promised timeframe. To build trust, ensure you have a regular release strategy and stick to it. If features slip from a release just communicate that to the business as soon as possible and slot it into the next planned window. Through experience with other customers, we have seen that releasing on time in a regular pattern is much better than holding back a release for an item of missed functionality which could cause uncertainty with release patterns. The user should then see that releasing updates is a predictable activity and likely won't have to wait long for the next update to come through.

Setting guidelines

To ensure consistency across a deployment all delivery and change management should have a set of guidelines to follow which are clear and visible.

UX Design Guidelines are typically useful especially for disparate delivery teams to ensure everyone is following the same pattern for delivery of new features. Guidelines examples are an agreed set of opinions and could include when to use an option set over a look up, theming, Form Layout principals, and column widths in a list view. This will really help when it comes to adoption and training as there will be a consistent design theme running throughout a deployment which a user should be able to recognize and ultimately find familiar to navigate.

Change Management Guidelines should also be agreed with the business. Setting expectations on how the team will approach exploring a business problem and the expectation that the delivery team is there to find the best approach for the requirement and the technology. Content here could include articulating to the business the strategy on aligning to the Microsoft roadmap where possible over customization, the change process steps on how to capture and work on a change, how feedback is taken, measurement of value impact, and iteration model for updates.

Data Management Guidelines are also useful to set expectations on responsibilities for maintaining data quality. Consider inclusion of business process expectations on what data should be filled in, the format it should be in, and how to resolve any issues. Ensure there is an agreed balance between IT and Business ownership here.

Embracing change in the right way

Here are some other rules to keep in mind when you are governing your deployment:

- The faster business value can be deployed the more satisfied the users are even if fit is not 100%. Often getting something out there for feedback is better than spending time building something that you believe to be a complete fit. Inevitably you may then find the business requires re-work once they start getting their hands on the feature.
- Set expectations of phased approach and take feedback before adding more capability. Avoiding re-work is always preferable.
- Ensure business changes are evaluated and solutions identified in alignment with the technology available. This may not mean the solution delivered is what the business was imagining but the regular communications and feedback loops should ensure this is not a surprise and met poorly.
- Showing innovation and deployment speed should help secure future project work and engagement is then made easier to justify and stop risk worry.
- Revisit investments and priorities constantly with the business. What is right today may not be tomorrow. Just as Microsoft provides updates to the application to meet with changing business needs the same should also be executed between the system functionality and internal business process changes. If gaps between how a user works and the system show, work quickly to resolve them before adoption rates plummet.

Measuring success

Linked very much to governance it is important to be able to articulate the value of every change you make in real terms to the business. We have seen many projects lose resources and funding when they don't have evidence of improvement.

When should I be thinking about measurement?

- At the start of the project/initiative, work with the business owners to determine the outcomes and behaviours that they are trying to drive. We know some delivery teams that refuse to implement anything within the system without agreed measures in place and the business confirming the improvement assumptions. If you cannot measure why adding something new will be beneficial to a Customer, Strategy, User then it should not be included within the solution.
- Define measurable and realistic metrics around these outcomes that relate back to organization key KPIs and strategy (get business agreement).
- Get started by looking at the situation before any change. Measure the current process against the measures to capture the current state (get business agreement again that this is an accurate representation of current state and the impact it has to customers and users)
- A few months after delivery, re-measure. Review with the business successes/failures and plan any changes if required. This is a constant process as business needs do alter and conversations with the users will help to prioritize what is needed. The ongoing partnership also fosters trust and visibility that your teams are aligned.
- If you are down the track and have functionality within the system where success measures are not in place, review with the business to decide on whether to keep it. If in any doubt, consider disabling the functionality to see the impact.

Internal/External facing metrics

It is good idea to have metrics across your project. Not all of these may be externally facing to the broader organization, but they will keep your delivery team in check with how the system is performing and any issues that need addressing.

- Internal measures are valuable for a technology team to validate data creation and general usage, and are critical to see that there aren't training gaps. These are not a measure to be used to validate a successful customer engagement but more to help understand where the potential gaps are.
- Keep the results to showcase technology adoption to the project team and sponsors. Many companies also take top results and reward the top performers to drive others to use the system.
- One measure to think carefully about here is proving a reduction on number of clicks to get a piece of work done. Outcomes should be the priority and not number of clicks. This is discussed more within the design principles.
- A few of many example measures:

Internal Measure Examples

- Number of logins
- How many records created
- How many records modified
- How many stagnant records
- Frequency of access to system
- Quality of data input
- Average time spent in solution
- Point score of number of times a user completes a key area in the system in any given month

External metrics are where the business teams get more involved to help quantify what impact new capabilities will have aligned to strategic KPIs and other metrics. If you are a business that has multiple initiatives targeting a specific area, it may be difficult to apportion what part the system played in making the difference. This is where getting the business to agree some upfront calculated assumptions will help post launch. Some external metric categories could include:

- **Financial** - ROI, increased sales, reduced costs, money-related measurements
- **Customer** - level of satisfaction, acquisition ratios, retention
- **Business user** - user satisfaction, user involvement, efficiency
- **Standards** - project related, delivered on time, to budget
- **Insight measures** (this is more about providing insight back to business rather than measurement of success) – Most successful deals have X touchpoints, most engaged customers have at least X active stakeholder contacts. This is one of the most important areas to think about – information and insight is very valuable.
- **Quality control** - consistency of process, streamlined processes, quality of data

When you are looking at the current state before making changes, take time to get some data around impact for not making a change. For example, if we look at the impact poor data quality has:

- **X missed opportunities:** If your competitors are gaining more insights from data than you are, they will have insights you don't. That might mean a company misses a critical opportunity for new product development or customer need that a competitor with a more mature understanding of data may capitalize upon. Companies should treat data as an asset and manage it to maintain quality to derive insights that can lead to competitive advantage.
- **Lost revenue:** Value of a missed deal and what that means over X time.
- **Reputational damage:** Misspelling of a customer's name, sending communications to the incorrect person.
- **Undermined confidence:** Lack of trust in value of application can create obstacles in gaining buy-in for future projects and dampen enthusiasm.

Tips for success

-
- Start with a small set of external metrics, then expand over time (get momentum first before you start measuring success!)
 - Analyze metrics by varying timeframes; the overall metric may look good but look at interval times.
 - Combination of metrics and KPIs to address the different business audiences. Executives speak to Business KPIs and Team leads potentially speak more to specific metrics rolled up to KPIs.
 - Remember it is often important to look at current state to identify impact to business in not making a change: can you quantify in terms of revenue/lost deals/customer churn/call abandon rates?
 - Look at how metrics can provide insight back to the business. A truly valuable project should be able to inform the business on areas of opportunity. If you can see trends that will help to make the business more successful, improve customer experience and help user efficiency then this is valuable to the business and also in itself showcases the value.
 - Don't just focus on the metric measuring the quantity of data entered, look at quality. Ask the question does the business care about this metric? If no, then consider dropping or keeping internal.
 - Metrics are an art and a science. Constantly review and iterate.
 - If there is low user adoption of an area review:
 - What is in it for the business user to utilize? (Does it relate to a KPI?)
 - Does it fit with how the user actually works? (Does it need re-design?)
 - Have you a robust user adoption plan to help train in the technology and process?
 - Business user feedback: What are your business users saying is the issue?
 - Think about experience outcomes – does it do what you and the business expected? Think about a survey to gather feedback.
 - If the area is no longer important – remove and simplify!
 - Once you have established your measures get them mutually agreed across the business and IT.
 - Ensure the Business and Executives 'sign up' to these metrics as a true view of what success looks like. We call this **Business Value Contracts**.
 - Getting a Business Value Contract in place early ensures the business and IT are agreed on common goals rather than disagreement/blame later in deployment.

Dynamics 365 – UX usage recommendations

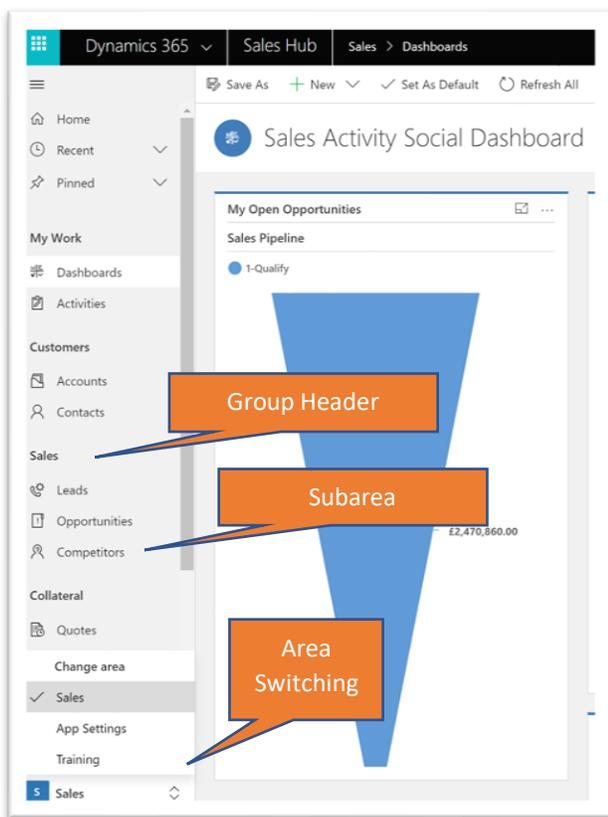
When a user of Dynamics 365 applications looks to complete a specific task, they could be presented with a number of different ways to achieve this by using functionality inside the app itself. It is the admin/maker's responsibility to ensure that the most efficient option is configured and recommended for usage. In this section, we cover a selection of UX components in Dynamics 365. This is based on the original intentions at time of design of these components, observed usage patterns on successful implementations, and typical use case scenarios. There will always be edge cases that warrant a different approach from what is recommended. Hence it is important to understand the key principles and then apply those as appropriate to particular scenarios and

try to avoid shipping your first thought without a review period with the business. It is much easier to change design before it goes live than after.

Navigation and dashboards

One of the key principles of the user experience is that it provides an intuitive flow through the application regardless of device. The intention is that a user should be able to naturally follow a business process, with the system assisting them in achieving their goal. It should then provide value and added intelligent insight to ensure the user has the best possible information at their fingertips to drive success with their customer.

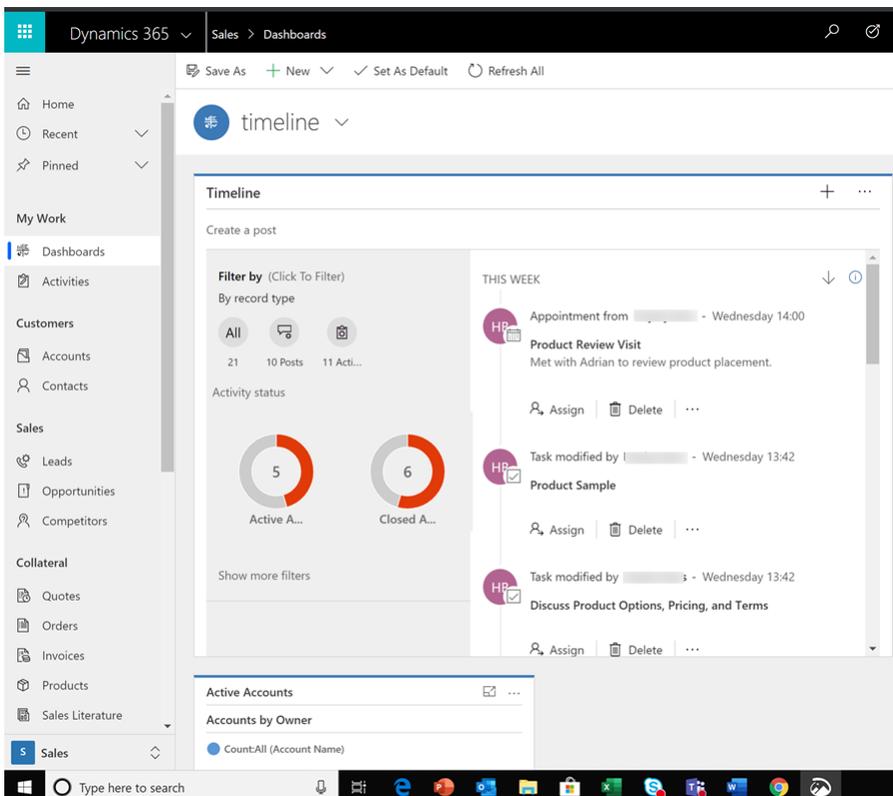
When looking to design the navigation bar always keep in mind that its primary task is to help the user context switch. This is often misunderstood. The navigation bar isn't intended to be a part of a contextual business flow it is just a quick visual way to jump to a category of information which can then be drilled down to record level. The options can be grouped which should help the user identify where to look for something, but it could take multiple clicks and visual searching to find the right area depending on how complex your site map is. The recent changes to allow creation of a role-based applications allows you to set a personalized site map for that role, which would reduce noise and help the user to find a specific working area. It is a good idea to really question your application if it has a lot of switching between areas. This could be an opportunity for a separate application depending on the user role it supports. It is a good idea if you are adjusting the site map to a specific role to think about the business language to use for area, group, and subarea. 'Group' specifically will help a user visually spot which section they should be looking in but keep text succinct to under 18 characters to avoid truncation of information.



Navigation example with Dynamics 365 Sales

Using the navigational bar makes sense for data exploration. For example, if the user is looking for a set of records via a view for a rarely used entity or if they cover multiple roles and change their focus during a working day within the same application. We also see context switching if you are a developer or customizer looking to review any changes on the front end of the application. However, this should not be the primary way a business user should get to the information they need to serve a customer. There are better ways to assist with the presentation of information in context and intuitive capture of information. Recognizing and adjusting the design of an application to use navigation more naturally through the dashboards, forms, and views themselves and only using the navigation menu for context switches gives a much more natural experience for the user.

Dashboards are a key component to the refreshed user experience. Role-based dashboards ideally should be designed for each role, answering key questions. These dashboards should be the home page from where 80% to 90% of day-to-day tasks for most roles should start. Ideally, all or most business flows should start from the dashboard. **This was the key intended use for dashboards: to act as the initial gateway into a user’s primary work.** Consider specific roles individually and provide the information and insight vital to their role, rather than have a generic dashboard for all. If you have dashboards in place already and move to the Unified Interface take some time to review any changes to layout and content. What’s new for example would now be replaced with the Timeline control so this could require a rework within a template.



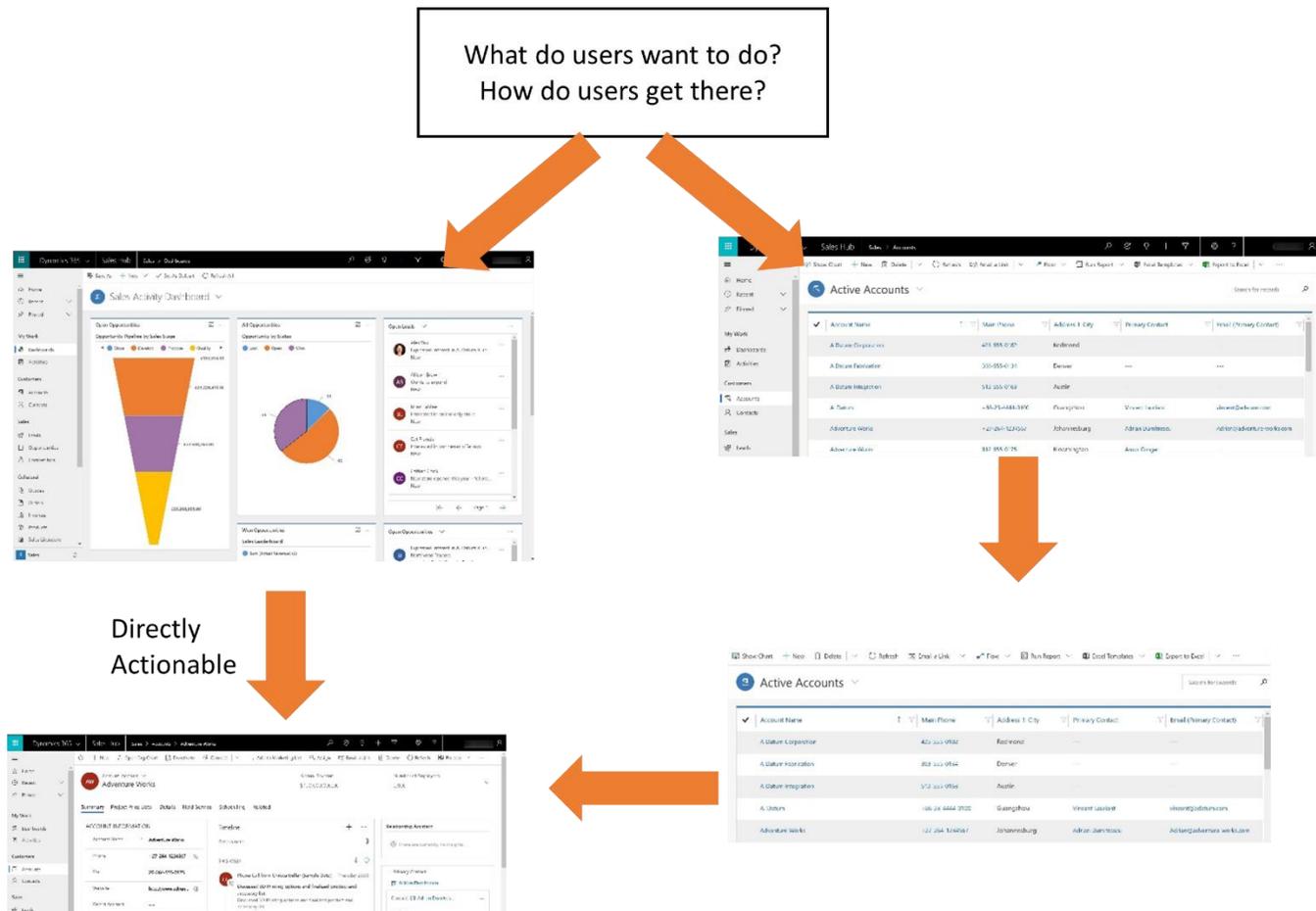
Timeline control added to a dashboard

The following table can help you decide what information to show on a role-based dashboard.

Question	Visualization
----------	---------------

What do I need to do to understand the state of business and my work?	Chart of important trends
What are the top tasks for the day?	Queue of activities that need to be done, My Activities and tasks
Who do I care about?	View of key accounts and contacts
How am I doing?	Performance
What information would be valuable to help me to be proactive?	Artificial Intelligence

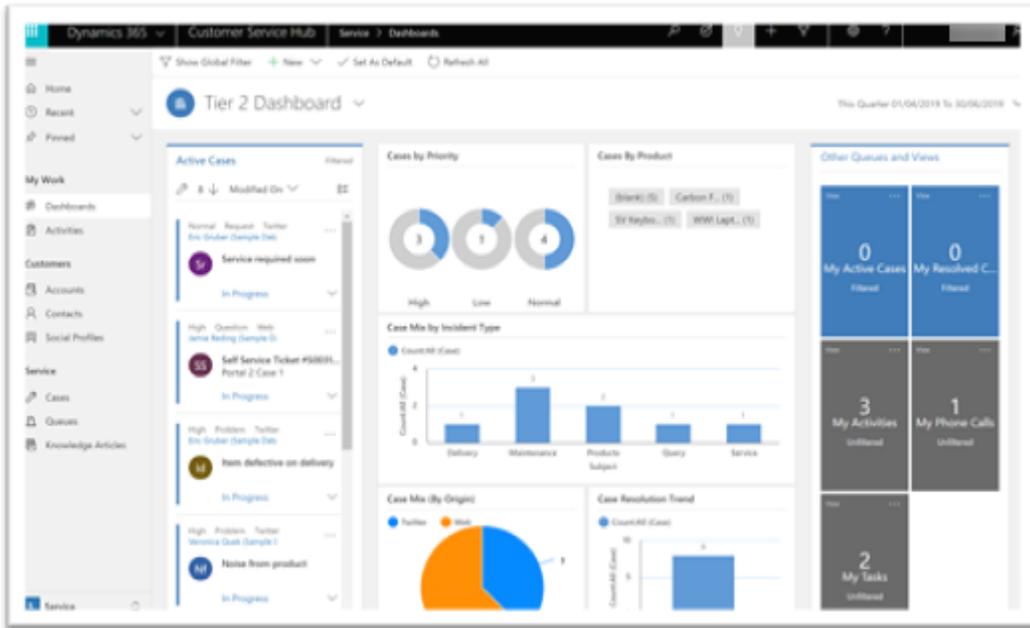
As you can see from the illustration below, driving your working day via a dashboard is a faster way to get to the target customer record. Care should be taken to make sure the dashboard is relevant and provides value to the user looking at it. Content within doesn't all have to come from the application but could be a collection of key information from internal and external sources and provide intelligent insight to enable the user to be proactive.



Interactive dashboards

Introduced originally for cases and now available more broadly, [interactive experience dashboards](#) provide the next level of interaction to the user. They can see, filter, and then drill into streams of data represented in

charts, data grids, and summary tiles. Filters can reflect across the whole dashboard and be easily reset back to the default.



This should be the primary location for a business user to start their working day. It is worth spending time to review dashboard ideas with the business champions to keep them insightful and constantly adjust the content to make sure it stays relevant to the business processes.

Form design

How a user interacts and consumes data for a particular record will affect the optimum form design. It is difficult to design a single form which caters for all roles within an organization. As a user, I would likely want the form to allow for quick data update and surface the tasks needed to perform a specific business process. However, a team leader may wish to see more graphical information and consume summary data. A 'one size fits all' is rarely successful, so before designing forms it is a good idea to review requirements and agree how many forms are required to handle the different needs. When it comes to creating a role-based application you can then make the right design choices on whether to include all or specific forms within the app. Understanding usage will help firm up your thinking.

Before making changes to your form design to reflect a new optimized layout for Unified Interface, it is always a good idea to create a copy of the original form as this provide several benefits:

- No impact if OOB forms get upgrades.
- Possibility to fallback or compare in testing with the "old" version of the form. Once migrated, and the new display is accepted the old forms can be hidden or deleted later.

Keep in mind that with custom copied forms you may have to make any app changes yourself rather than have them automatically apply which is a flip side. Form requirements are also likely to alter as the business changes and you should question every component you add onto a form to confirm relevance back to the user and the end customer. As the forms evolve you will see a reduction of whitespace and greater flexibility on component and control placement. Think also about [utilizing the header](#) correctly and review the top 4 fields the user needs to constantly see on a record. This area is docked when you scroll down a screen and so should contain the most

important at a glance data and note this area can be read-only or edited. See more on this topic in the next section and remember less is more!

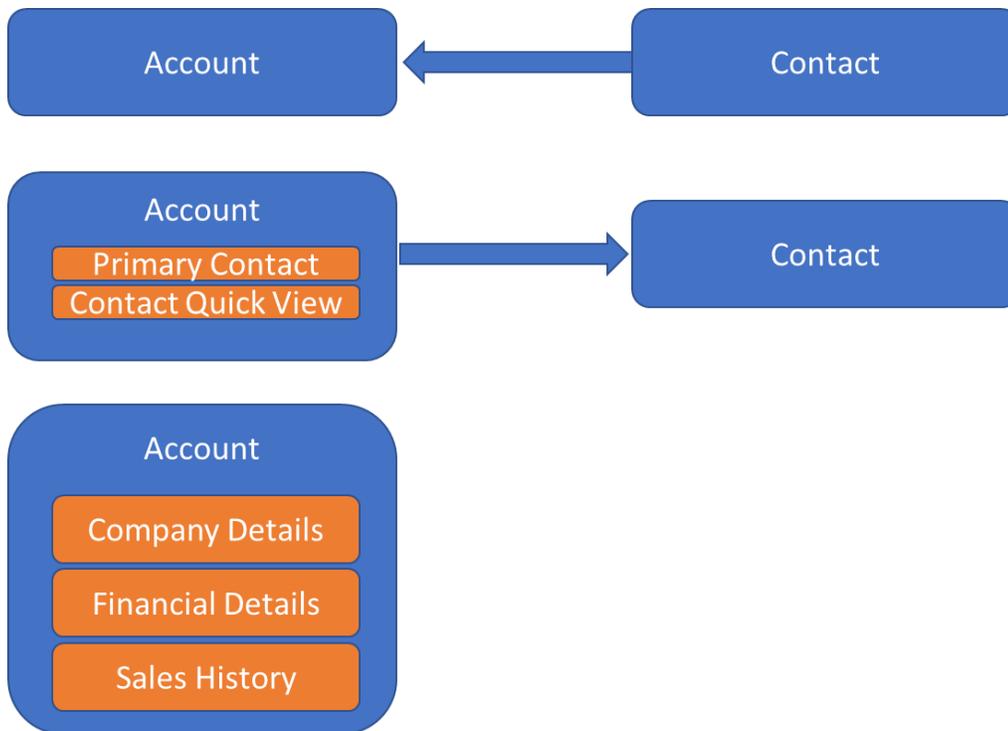
Review your data structure

When designing a system, it's useful to understand how people consume information. "Millers Law" is a valuable principle to understand and follow: an individual can process approximately seven "chunks" of information at a time.

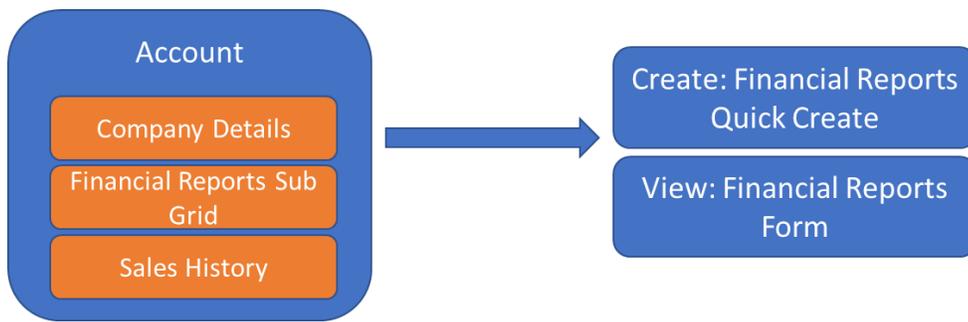
What this means in practice is that providing forms that contain long lists of fields is not an ideal way for users to consume information. It likely forces them to subconsciously break the information into chunks of related and meaningful data so that they can consume it or worse ignore elements further down the display.

It is possible to assist with this so that users don't have to do this themselves, which both reduces the amount of effort needed, and therefore the frustration felt using the system, and also allows users to consume the data better.

This can be done by recognizing and representing natural structures and collections of related data together in the user interface. Breaking larger concepts, like an account, down to meaningful parts such as the company details, financial details, and contact information can assist with this. Each of these can then be represented either on the form via tabs or as separate entities depending on the use case.



Splitting data out into separate entities has advantages. It's possible to separate the creation rather than consumption experiences for subsets of the data using quick create forms, such as in this example.



Consume to create example

It's also possible to more easily control the visibility of data using standard product security mechanisms.

One of the reasons that data was frequently combined on a single form was to avoid the transition time required to load multiple forms and ease the challenges of navigating multiple pop-up windows. With Unified Interface, these transitions are simpler and more intuitive. This means that the cost of the transitions is much lower. In fact, where a user is naturally having to switch from considering one concept such as account to another concept such as the primary contact, this transition can actually be helpful to them in making that mental context switch. This is a much improved model and experience for the user.

Multiple forms

Try to limit the number of forms an individual user switches between on a single entity. Form switching causes confusion particularly when ensuring you are in the correct form for a record. Review if it is possible to merge them into one form or, if the user carries out two distinct roles, then potentially two applications may be better. It's rare that a form change is all that is required when swapping between two different roles. Showing just the elements needed for the one role helps the user complete tasks more quickly as they can focus in on the process. The user can then switch applications to the other role when they need to perform the other job function. This make the user flow a lot simpler to navigate.

Overall, you can have as many different forms as you wish across the whole system and it's very easy to create and update them. However, it's sensible to balance the number with the need to manage and maintain them when changes happen. Equally, take care to align this to the user roles to ensure one role doesn't have multiple forms on a single entity, which could cause confusion.

Reflow

Once you have your forms in place review them on all devices to check the design and placement fits well regardless of access point. The application does most of the heavy lifting here, but it is sensible to include a testing review on multiple devices to achieve the optimum design. This is particularly important if you have updated a previous form used in the legacy web client and hadn't considered mobile in the original design pattern. It should be a standard practice to take into account the display regardless of device to ensure the working experience for the user flows smoothly.

Tabs

Our forms have the concept of a tabbed view to allow a user to click to see a grouping of information about a record. The introduction of tabs horizontally across the form helps the user to easily context switch on a record without having to scroll down a page and works well when considering a touch device. It is important to break content within tabs into a logical order aligned to business process. Test this out with the business user and observe how they use a form to see if you have the order correctly aligned to the role.

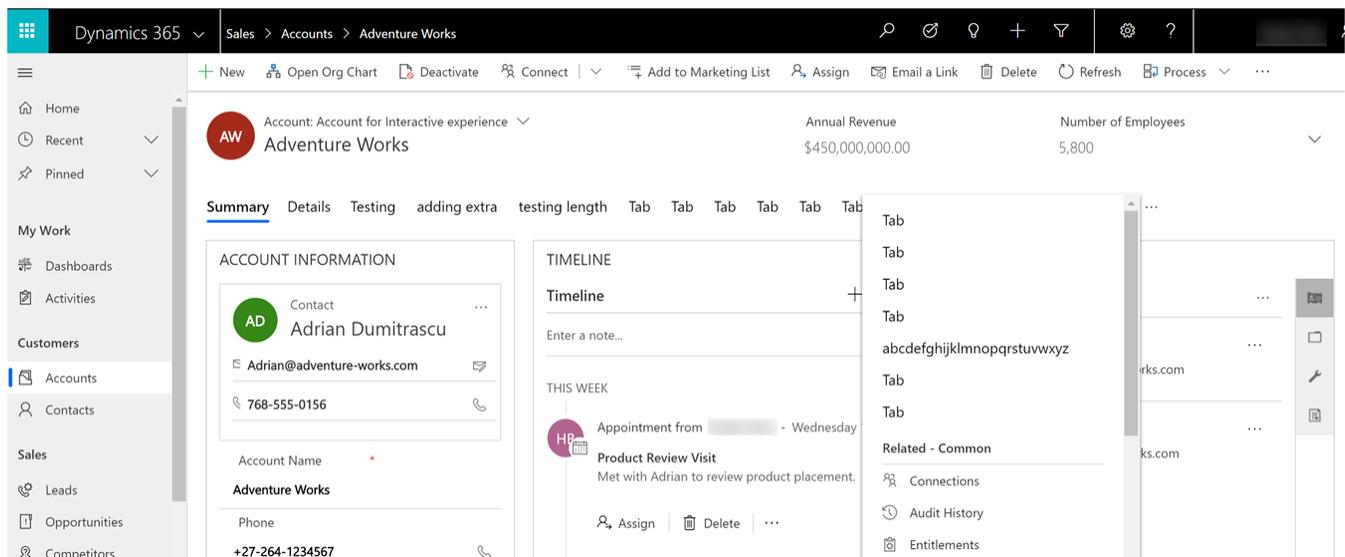
As some guidance, the first tab should be the 'most used information' for the role. Some research with your users should help you discover what they need to interact with the most and the remaining elements should be

aligned across additional tabs. We often see the first tab contain the first elements of data a user enters when they create the record but likely that is not the most used information going forward.

For example, let's take the scenario of an Account Manager who has 5 key organizations they work with on a daily basis. When the account manager created the account, the first fields they entered information into were Address and Main Reception Telephone Number. If you survey your account manager and ask them how often they look at the address or main office number, they'll likely say rarely. If the address information is not the most used information for the account manager, then it doesn't make sense to have it on the opening tab. Consider swapping that to another location and placing here instead the Timeline history of activity and some insights which help the account manager be more proactive with the account.

Some key factors to consider:

- Make sure your tabs are well labeled to match the business language. (Simple is key.)
- Review how many tabs you need per role. Consider if a dedicated tab makes logical sense to the business user rather than having primary concern about the number of tabs (within reason!).
- Matching tabs back to the business process helps with navigation.
- The longer the text on the tab label, the fewer tabs that are likely to fit on a screen. See below screenshot. Note: multiple tabs on an account screen illustrates the behavior.
- Additional tabs are then available within the More button (...) in a drop-down list. As you reflow the screen for a mobile view, the More menu just gets longer. **While you can have a lot of tabs on the screen this is not a recommended practice for user experience.** Multiple tabs could cause confusion when trying to find certain elements and should be kept to a logical balance with the user role.

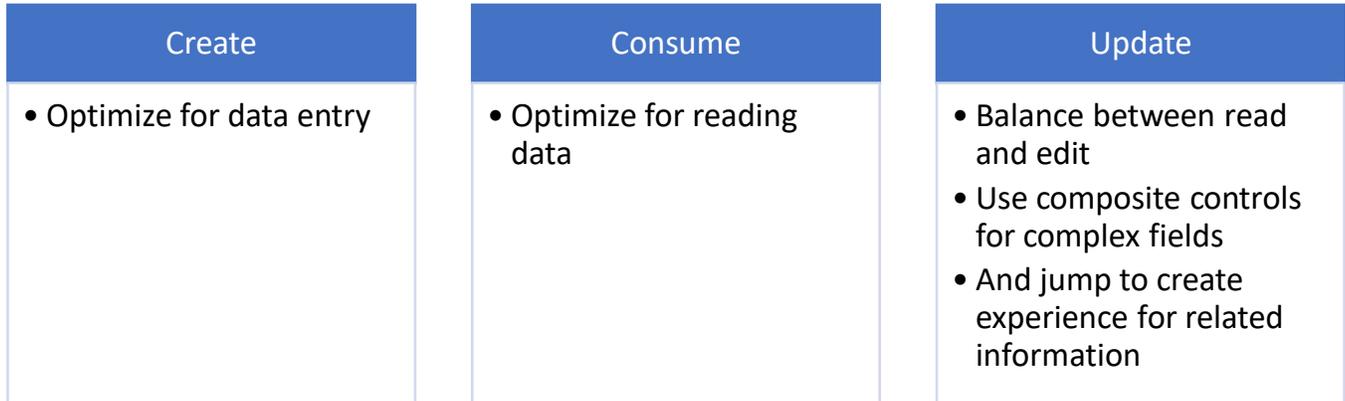


Example of user display if you have multiple tabs on a form – more difficult to navigate and likely tabs in the drop down may not be accessed.

Optimize for create and consume experiences

Another key principle of the new user experience is recognizing that creating and consuming information are different models of working.

The new user experience allows you to optimize for both creating and consuming, recognizing that many adoption and usability challenges in the past were caused by a single form design intended to support both, but in reality, optimized for neither.



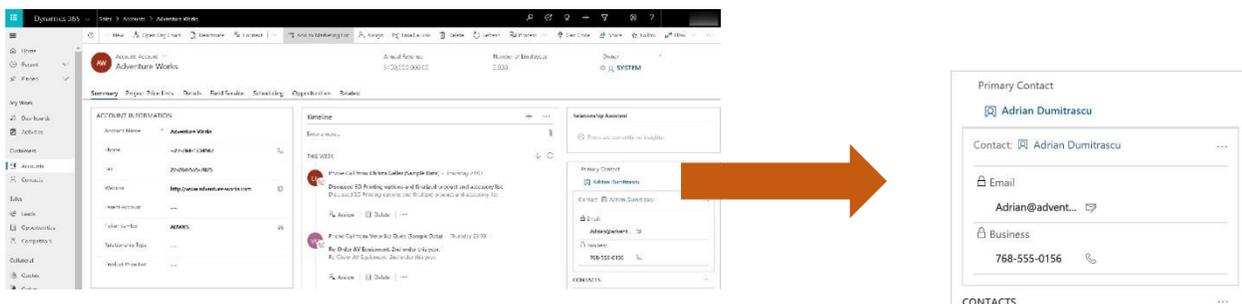
Quick view form

[Quick view forms](#) were created to provide a preview of key attributes of a related entity without unnecessary navigation to the related record. This works by surfacing key information on a parent entity form through configuration. They aren't intended for building complete de-normalized forms where multiple fields from multiple related entities are all shown in the parent form. The preceding design option is possible, but it isn't recommended.

Quick view should be used to show:

- Key information only, not replicate the entire record.
- Information that is directly actionable or related to the outcome driven from the parent record.

For example, showing the email and phone number of a primary contact on an account form is an ideal use case for quick view forms where the likely action is to be contacting the primary the customer. Rather than exploring the primary contact's full details on the contact form, quick views make it possible to directly contact them without that additional navigation. If you need to check out the contact's full details, that's still possible by navigating to the full contact form.



Note that it is possible to have a different quick view for the same entity with a different parent entity form. The more tailored the quick view forms are to the scenario, the more effective it will be. For example, the contact information needed in a quick view form in the context of an open invoice might be different from the information needed in the context of an account (see above). In an open invoice form, it might be important to show the credit limit or credit worthiness of the contact.

Capitalization

Earlier we talked about the importance of consistency to help with adoption. Studies have shown that improving the readability of a form also helps with comprehension, usability, and productivity. This is not only about what fields you select and labelling with business terminology but also the capitalization that's used.

Title Case Looks Like This

It's Especially Hard to Read the Longer It Gets

Sentence case looks like this

It's easier to scan and comprehend in short and long forms

Sentence case is:

- Easier to read.
- Friendlier because it's more casual.

Over time we will be modernizing Dynamics 365 to align to this capitalization style. As we look to ensure Microsoft's out of the box areas follow this pattern it is good to ensure you have a matching design guideline for your own custom areas. Product names are proper nouns and so will always be capitalized, but for all other areas you will see the more modern and friendlier sentence case format being adopted.

The screenshot displays the Dynamics 365 Sales Hub interface for an Opportunity record titled "Audio Equipment". The breadcrumb navigation at the top reads "Sales > Opportunities > Audio Equipment". The record header shows "Opportunity: Opportunity" with a blue callout "1" pointing to the title. The "Est. close date" is 20/04/2017 and "Est. revenue" is \$8,710,785.00. A progress bar below the header shows stages: Qualify, Develop, Propose (37 Hrs), and Close. The "Propose" stage is active, indicated by a red bar and a blue callout "2" pointing to the "Propose (37 Hrs)" label. The "Summary" tab is selected, showing fields like Topic (Audio Equipment), Contact (Ivan Komashinsky), Account (Wide World Importers), Purchase timeline (Immediate), Currency (US Dollar), Budget amount (\$8,135,000.00), and Purchase process (Unknown). A "Timeline" section shows a post from "YESTERDAY" about a competitor being added to the opportunity. A blue callout "3" points to the "Relationship assistant" section, which currently shows "There are currently no insights." and a "Stakeholders" list including Ivan Komashinsky.

Example screenshot of capitalization changes – Points 1 2 and 3 show examples

Field type selection: Option sets vs lookup fields

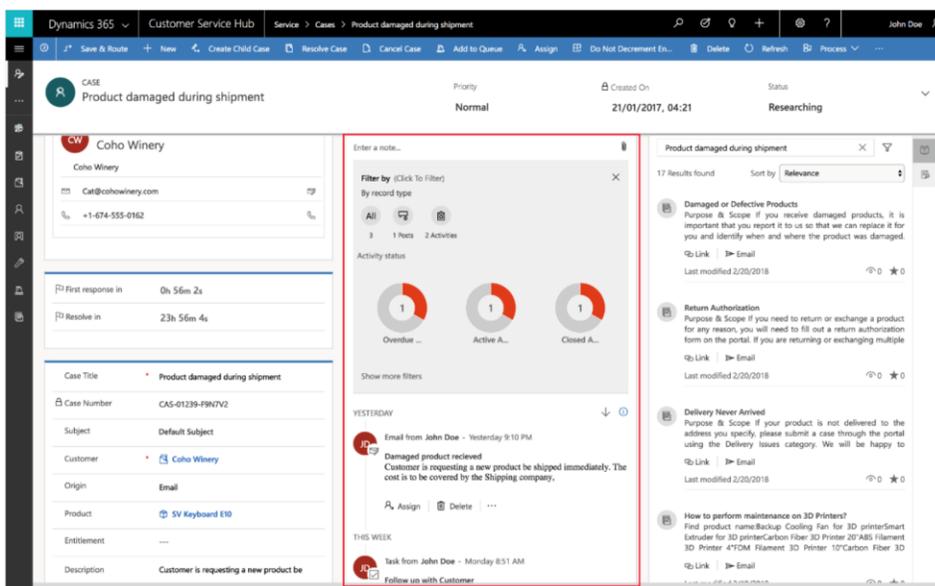
When looking to add a field to a form that allows for selection of information, typically an option set or a lookup is the right way forward. Each is a perfectly valid way to allow a defined selection but use the following as a guide on when to select the right choice.

Option Set	Lookup
Component of choice for creation of a simple list	Component of choice for selecting data that could change on a frequent basis
Designed for infrequent changes as updates require a publish and therefore reload of metadata from the server, and offline clients	Option to filter and slice the data to handle a more complex model of selection using views and security model
Handles multi-language well using the out of the box mechanisms	Data within lookups aren't designed to handle multilanguage in the same way as an option set
Option set security is linked to the residing entity and role. Further restrictions would require code	More control over visibility of entries
Simple type-ahead search with limits on slice and dice capabilities.	Requires a tool like package deployer to move 'reference data' used by lookups across environments.

Timeline Wall control

In previous releases we've had a social pane and activities grid to provide a snapshot of engagement with a particular area. The Timeline Wall is an evolution of the social pane and can be viewed on an entity record form or on a dashboard (replacing What's New). The Timeline Wall is designed to be a quick and easy way to see all activities that have happened with an entity (and related records), such as emails received, follow-up tasks, and automatically generated system posts, custom activities, plus any notes taken during customer interactions.

The filter options and summary graphics make it a lot more usable for an 'at a glance' view of the previous history. It also takes up a lot less real estate than the activities grid showing a conversation view to see connected interactions. In a world where we are all about efficiency and reducing clicks, this is a great area to review with business users. Driving a user to this area first before looking at an activities grid could eliminate some screen movement and need for multiple level drill down. Graphical filters are also in place to help a user to interact with the data as appropriate.



Grid selection

Dynamics 365 has several different ways to surface and view a grid of information. Here are some common tips that are recommended in the design:

- Ensure each grid is reviewed to make sure the most relevant fields are added as columns.
- Try to reduce horizontal scrolling by adding too many columns.
- Think about column width, order and whether the label visibility fits at a glance without truncation (List Views Only).
- Review the sort order for the grid – does it align to the business value.

Note: it is the small items like this that often have an impact to the user's perception of the system. Sometimes the biggest complaints are about how the information is displayed rather than whether the information is there or missing.

Editable grids

When a user has to update multiple records, it can be frustrating to have to launch the record for every small change. The introduction of the editable grids control is a positive step towards speedy data entry. In your re-design, take some time to look at the grids to identify where it would make sense to introduce an editable display. They are typically used when someone is working from a browser environment rather than a phone. They can be embedded as a subgrid on a form or visible as a view of data.

If a sales executive is looking to update opportunity values quickly from a list view or adjust quantity values and price discounts of a set of products associated to an opportunity, then an editable grid saves significantly time. If you want to drive users to have to click a record to move through a process rather than be able to edit quickly, then sticking to a read-only list view would be the alternative. Select the right option depending on the use case. Here's more information about [editable grids](#).

Subgrids and associated grids

Subgrids and lookups are used to provide a natural flow for quick in-context information discovery and consumption. Use subgrids on a form when the related entity information in question is needed frequently and the number of records that needs to be visible for effective consumption is low, such as less than 10). It's important to ensure that information that's key to the decision-making process, like owner, is visible on the subgrids.

For example, if there are, on average, three to five opportunities per account and the key fields needed in the context of an account are only Opportunity name, Owner, Estimated revenue, and Estimated close date, it's ideal to show the opportunities associated with an account as a subgrid on the opportunity form. If it was necessary to know the status of all the opportunities to determine if they're relevant to the current process, it would be important to add that to the subgrid as well. This avoids the frustration of users needing to open up each opportunity record individually to check the status—a simple change that has huge user experience benefits.

Associated grids are used to provide a more detailed immersive experience for entity information. They are typically navigated to via the related tab on an entity form and an example could be the Activities associated grid which is linked to the main Account entity. Use associated grids on a form when the related entity information is needed infrequently or if the number of records that need to be visible for effective consumption

is moderate to high, such as greater than 10. Associated grids take more effort and clicks to get to. However, they provide a dedicated experience to explore and consume a large number of records.

An ideal example is the standard associated grid for activities on the account form. A large account might have hundreds of activities. You can see the snapshot of recent activity in the Timeline but when you need to interrogate the complete history, an associated grid is a better solution. An example could be when reviewing an escalation trail or while analyzing the loss of a deal. Hence it makes perfect sense to show the activities associated with an account as an associated grid.

Utilizing views

A view is a filtered list of records available to the user that match specific criteria. We provide standard system views such as Active Accounts, but a user can have multiple views (system and personal) to be able to quickly see a list of records that all fit a specific criterion. Providing your users with the right views is always an easy way to help with adoption. Take some time to review the common filtered views each user role requires and make them available within their application.

Within their role-based application you can release just the specific views the user requires. This will help the user focus only on the areas they need and reduce any noise from other teams. What is important is that all views have the correct field columns, sorted correctly and in an order that makes sense. This includes thinking about the associated, lookup, and advanced find views as they are often forgotten in planning. Ensure the common fields are included in these areas as it helps a user with the visual confirmation that they have found the records they were searching for. Once you have this set up and released, go back and check they are right a few months down the line. You will likely get better insight on the right columns following usage as users will spot missing areas once they have had a chance to use it in a live environment. Usability enhancements are also constantly being made to grids and views as this is a core work area for a user. Some new enhancements have been made available in the [release notes](#). Always check the [release plans](#) to read about the latest updates from Dynamics 365 and the Power Platform.

Composing Applications

A typical application is comprised of entities, forms, charts, dashboards, and business processes and has added flexibility to extend. The app composition can really be tailored to a specific group of users allowing you to make available only the areas that are of highest relevance for the user. It allows you to create a solution tailored to a set of users within your business but keep a common design experience which is great for adoption and continuity.

Organizations can also build their own completely custom applications using the same platform technology. It is a design decision on whether to create a new app or align to existing applications and the answer will be dependent on the business need and ultimate value.

For more information about the technology capabilities, see [‘What is PowerApps’](#).

Handling multiple apps

The Dynamics 365 Home area shows the user the applications that they have access to launch and work with. To make it simple for the user, reduce the amount of applications available to them. An average user would likely only need to see a limited number on the home page (depending on how many different user roles are available to them and the amount of contexts they need to switch to). Take care to remove any training/sandbox instances from their view once you go live to ensure the user is clear about which application they should work

in with live data. Expose training applications only when you need the users to access them. You can also use the filter option to slim down the number of applications available on display.

Business process

The process bar was designed to provide an easy way for users to follow an established process. It's meant to encourage the right outcomes and ideally should not be considered a wizard for data capture. We typically see the process bar utilized as a checklist to encourage the right practices and to enforce checkpoints at various stages. Try to keep the number of stages and the steps per stage to a manageable number. This is especially important for mobile and tablet usage scenarios, where long or dense processes will more adversely affect usability. The following are the maximum numbers we allow per process but in reality, you shouldn't be getting close to these numbers on stages and steps from a user experience standpoint.

Process Bar Area	Maximum Limit
Stages	30
Steps	30
Entities	5

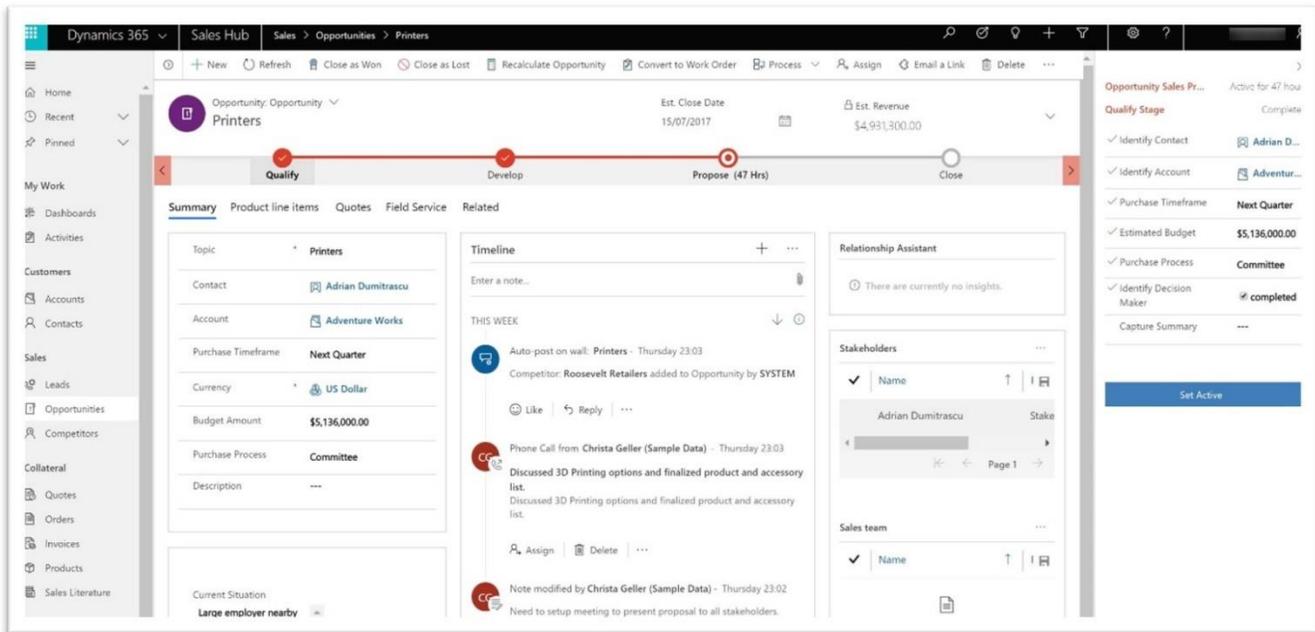
Another key purpose of the process bar was to link together related entities that are tied to a process through automatic form transitions. This goes hand in hand with the inline navigation paradigm. It is most effective to design a user flow to leverage these UI transitions enabled by the process bar along with other natural navigation methods like lookups and quick views.

For example, the first step to update an existing case might be to associate a contact to the case and if needed, create one using the quick create form. In this way, all components of navigation are user-focused.

Many enhancements target streamlining flows, in particular:

- In-place navigation rather than pop-up windows:
 - Previously, the use of pop-up windows meant that users lost context of their work as they opened and moved between windows.
 - In-place navigation and side docking allow the user to intuitively flow forwards and backwards within a process without having to think about manipulating windows to get back to the information they need.
 - View the release notes for future details about the [immersive experience options](#) and [custom control capabilities](#) within a business process flow. When designing, keep the labeling intuitive and think about utilizing visual custom controls where possible to help areas stand out. This is particularly useful on a mobile device where touch is more prevalent.
- Auto save:
 - This eliminates the need for the user to need to think or be prompted to explicitly save information as they progress through the application.
- Reduced scrolling and clicks:

- Changes to the overall presentation and flow of data reduces the need to scroll within long forms. This was a major cause of additional clicks to access information in previous versions of Dynamics 365 apps and it didn't work well on a touch screen device.



Business process flow updated design with right side dock.

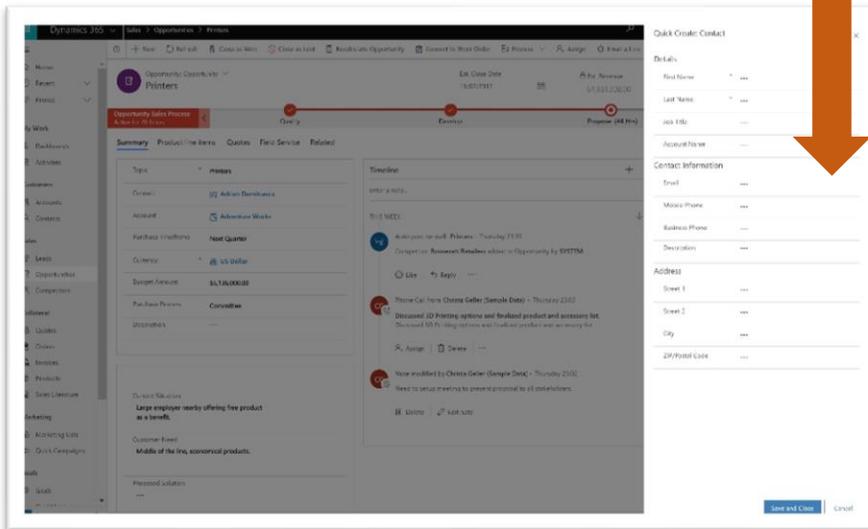
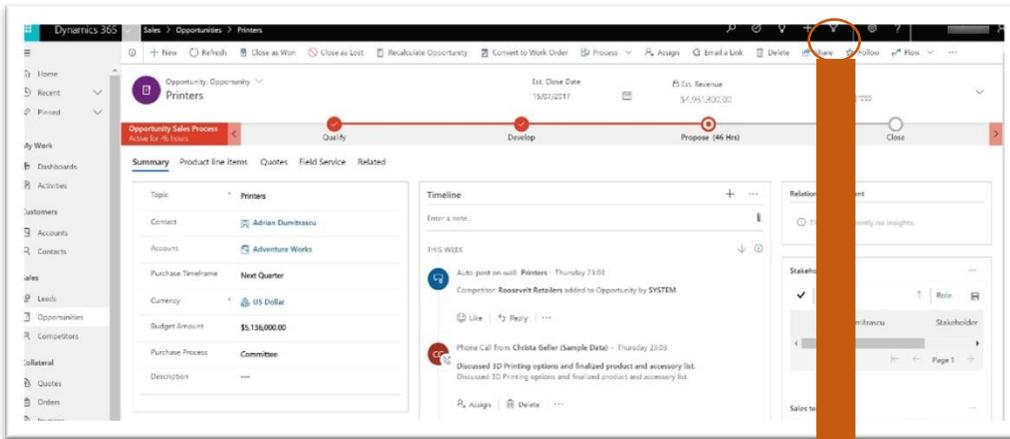
Quick create

The quick create form was designed in recognition of the fact that creating content is a different process than consuming content. The intent behind introducing the quick create form was to provide a lightweight form to optimize the experience for creation. As the name indicates, this component was designed for creating content quickly. Hence it is important to only have minimum fields. Remember to always surface the mandatory information required to physically create and save the record within the quick create. We have seen examples of organizations that have forgotten to add all the mandatory fields causing a save issue when the user moves away from the record.

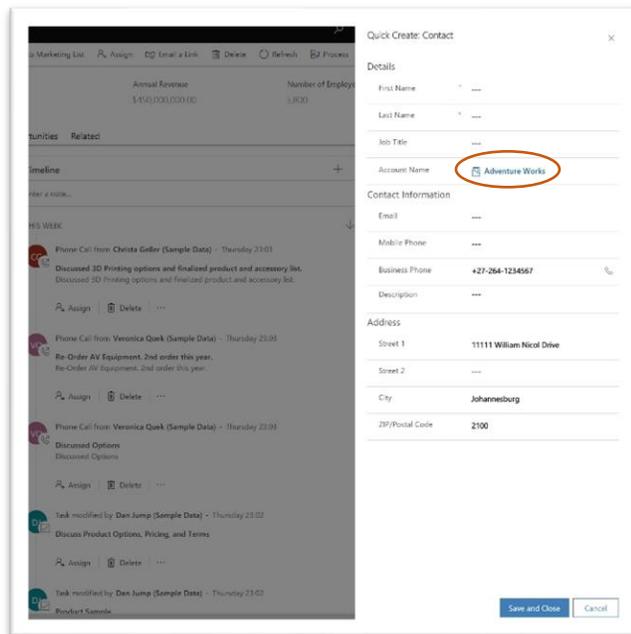
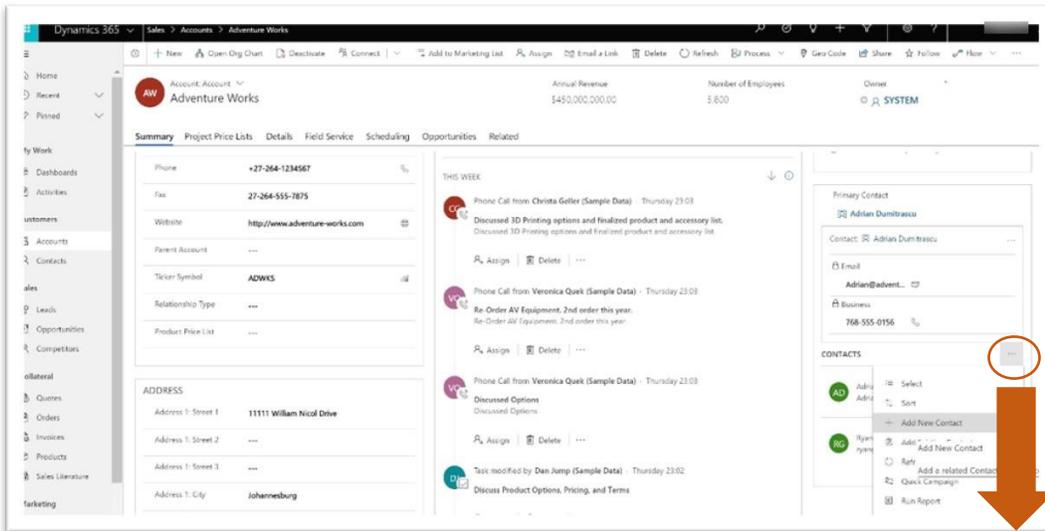
The user can transition to the full form for further updates or edits if needed. Quick create forms also allow the main form to be designed and optimized for the consumption experience. Quick create forms are often forgotten to be enabled when you create a new entity or left untailed to be effective. Ensure they are included within your design phases and switched on for a user to leverage.

Quick create forms can be used in two scenarios:

- To quickly create related information without losing flow. The form is launched from a subgrid on a form or from the global menu bar.
- To quickly create information in a new unrelated context. The form is launched from the navigation bar.



Example: Quick create from the menu bar shows a new create screen out of context with the form behind. It's good to ensure the business user understands the new and different launch patterns.



Example: Alternative quick create launch showing in context with form.

Dialog deprecation

Dialog boxes are a step by step data entry wizard designed to force a user through a specific path of prompt and response questions where data is updated or actions are fired in the background. We have announced [deprecation](#) of this functionality as it has many limitations, including:

- Linear motion.
- Not supported across multiple devices.
- Lacks progress visibility.
- Updating data outside of dialog: Abort wizard and re-launch.
- Synchronous only.
- Must be started by the user.

-
- Dynamics 365 apps: specific wizard only

In real-world deployments, usage patterns fall into a number of different categories:

- **Approvals** Triggering an action for a user to specifically approve an action. Simple or advanced yes/no.
- **Data validation** Presenting known information to a user to confirm it is valid.
- **Data entry** Structured capture of data. The more classic dialog example – simple and extensive datasets.
- **Notification** Using dialogs as a notification mechanism for outcome of a workflow.
- **Scripted actions** Guiding a user through a specific script. This is common in a phone-related role.

Organizations need to plan for [alternative options](#) if you are currently using this functionality. Options could include the following and depend on the business need:

- Business process flows
- Workflow
- Playbooks

As part of the scenario review it is important to understand dialog use in broader business process context. Solution recommendations depend on the scenario.

- What is the overall business process where this dialog is included?
- Does the solution need to be launched from within the UI on a record?
- Is it an isolated action that could be a standalone app?

Some questions to consider before moving forward:

- Before agreeing to any changes to dialogs, understand the business value and the importance of any migration effort.
- Evaluate whether you are going to deliver a quick solution or review if it should have a complete re-design.
- Build proof of concept examples of solutions before committing to an outcome.
- Gain business feedback throughout design.
- Balance the need to lock down elements with flexibility for the user to add in their own nuances.
- Try to agree on common guidelines for each category of dialog replacement for consistency. However, don't be afraid to pick the best approach for user experience for unusual requirements. It is best to provide the right solution to the problem for user acceptance rather than fit a requirement into a piece of functionality that is consistent with other areas.

Custom icons (grid and entity level)

When you create new custom entities within the application it is always a good idea to replace the standard custom entity icon (puzzle piece) with a custom icon. This will make the entity make more sense to the business user especially if you have multiple custom entities on display! The best format to have the icons in is SVG (scalable vector graphics). This tends to be crisper and sharper on a high-resolution screen over bitmap graphics and marginally faster to load. It is supported on most modern browsers but if you are using Internet Explorer or Chrome and the icon appears black or white then open the SVG file in a text editor, such as Notepad, and remove this parameter `fill="#000000"` before saving.

Recommended format (for Unified Interface)

- Width and height: 16 x 16 px
- Padding: 0 px

-
- Background shapes: transparent
 - Icon color: #FF000000
 - Icon format: SVG

[Icons can also be used within a grid view](#) to highlight key fields. This is especially useful for visual people who need to spot specific records that require immediate action. Take care not to have too many icons within a single grid as it will have the opposite effect in terms of clarity of the screen but something like a status/rating field could be a good candidate for change.

Handling web resources and iframes in Unified Interface

With previous versions of Dynamics 365 applications, a web resource or iframe was a good solution to serve up information from another source. This could be in context or as a window on another application. Typically, this capability required technical resources with knowledge in HTML files, JavaScript, and CSS to implement. [Web resources](#) can also be seen in form customizations, the site map, or the application ribbon and supported primarily for the browser rather than multi-device.

With a move to the Unified Interface, it is a good idea to review these components to ensure they are working as designed. The reflow element of Unified Interface will likely mean that a placement review will be required for these components. The styling may also be out of step with the new UI – take care to update these for a smooth fit within the new location in the application.

Client-side scripting

It's fair to question the place for client-side scripting as a topic in a user experience document. However, it is here for a reason. Generally, we have seen in deployments the predominant cause for poor user adoption is poor performance alongside fit to business value.

One of the major reasons for poor performance is poor scripting habits. If the scripting response time is slow then that will have a tax on the user in terms of time. Not only will that be frustrating for the user it could also have a negative impact on the end customer who may be on the telephone to the user waiting for a piece of information. Coding can have many benefits to help enrich a configuration to deliver a bespoke requirement but always think very carefully about the impact back to the business.

Follow these guidelines to help ensure optimal user experience.

- Avoid scripting:
 - Try to avoid complex client-side scripting. Look at alternative options like server-side coding, instigating [business rules](#) as they are declarative, so easier to maintain than code, and can also run server side. For example, instead of showing and hiding some fields based on two roles, it might be better to create two roles and use two role-based forms.
- Onload event:
 - Even if you choose to write client-side code, avoid functions with excessive wait times in the onload event. These delays directly add to the form load time. The higher the form load time, the lower the user's perceived application performance.
- Server callback functions:
 - Server callback functions in client-side code cause the most delay. If a synchronous call is made and the client is waiting for a response from the server, the user may believe the client has stopped responding. Although the client is just waiting, the perception the user gets is that Dynamics 365 is prone to hanging randomly.

Anytime you do add custom scripting into the design, remember to take time to comment in the code the intended action.

Commenting code benefits:

- Should staff turnover occur within the administration/maker teams it provides a quick way for a new individual to see the intended behavior.
- If support is required for an issue the teams can quickly review what the area should be doing.
- Training teams can take a glance to confirm understanding before documenting any materials for end users.

Custom theming

The primary purpose behind introducing theming is to help brand the application along the same lines of other corporate line-of-business applications a customer might be running. This does have a tangible positive effect on user ownership and adoption of the system as well as enhancing the experience when the system is deployed in a customer-facing scenario.

Currently, theming allows for branding the application with a customer icon and changing accent colors for hover and selection of certain areas and entities*. Theming isn't meant to enable changes to any of the current layouts or actions.

**Note: Currently there is a difference between the theming areas for the legacy web client and Unified Interface. The key difference being Form Sub-Grid 'Panel header color' and no horizontal bar, We have made steps to reduce white space and keep the design following a clean fresh experience and therefore have fewer areas for custom colors to reflect that. Looking ahead, we are working on a new unified theming system that will support a broader range of capabilities but keep reviewing the release notes for any updates.*

Dynamics 365 apps include a default theme. Keep the following best practices in mind while you use the theming functionality:

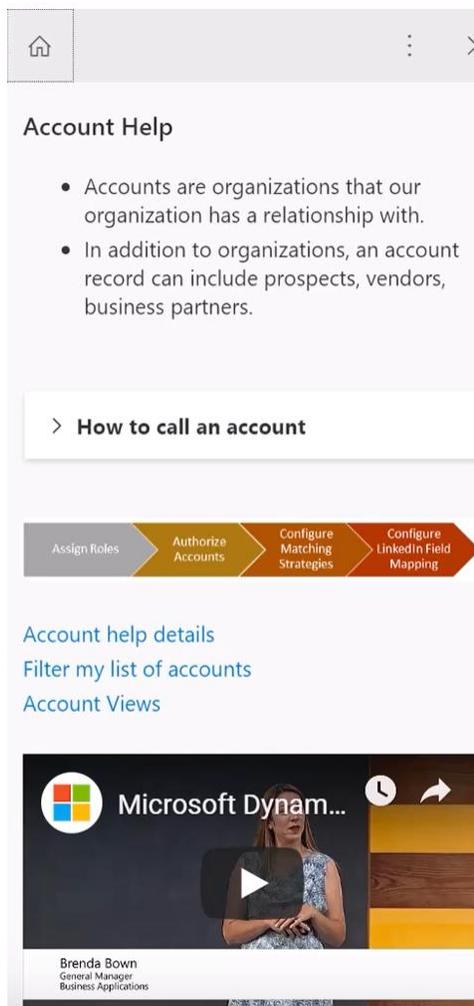
- **Accessibility.** Be aware of the color contrast for new custom themes. Contrast ratio is an important measure of accessibility. Our standard theme has the correct contrast ratios to ensure optimal usability. High contrast mode always uses the default color settings.
- **Don't overuse colors.**
- **Use few color groups.** Colors lose significance when there are too many. This includes accent colors which can be seen in a few places like custom controls and business process flows – adjust only as needed.
- **Adding a brand color and logo is common practice to help familiarity with application.** It also sets tone that it is embedded within the culture of the organization rather than just another technology tool.
- **We see organizations also use theme colors to delineate between environments.** This provides a quick visual reminder to a consultant on whether they are in a production or non-production environment and reduces potential errors.

For broader guidance of our themes to keep aligned across all of your applications, we have design guidance provided via [Microsoft UI Fabric](#).

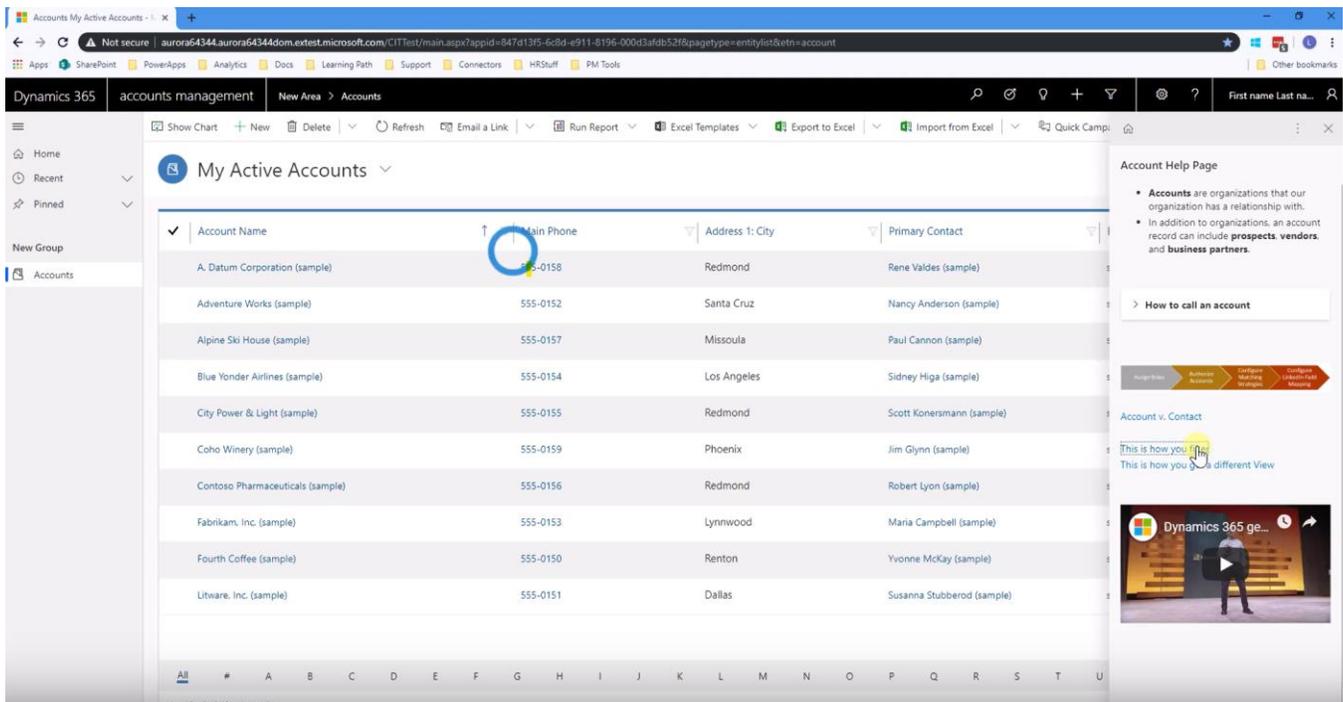
Custom Help and Guided Tasks

Providing in-context help is a great way to support your user adoption strategy. We have a tool called Learning Path which enables you to have a custom sidebar to display 'how to' content in a variety of formats. The tool also provides a facility to drop in-place notes throughout the application to help guide a user through the steps to complete a specific action or process.

For greater flexibility on the Unified Interface this has now been superseded with Custom Help and Guided Task functionality. This new area can be easily adjusted in place with custom content to help a business user. Use this area to provide helpful tips and procedure content to guide a user through a specific area of the system and reduce the on-boarding element. Editing the sidebar is very simple and allows for a wide variety of component types from inserting links to an external site, providing text content, launching videos, coachmarks and balloons which highlight elements on a page to orient a user. A preview video of the functionality can be found [here](#) or see [Create guided help for your Unified Interface app](#). We recommend ensuring this is part of your deployment strategy to maximize user adoption.



Custom Help Pane example.



Guided Task coachmark.

Design example

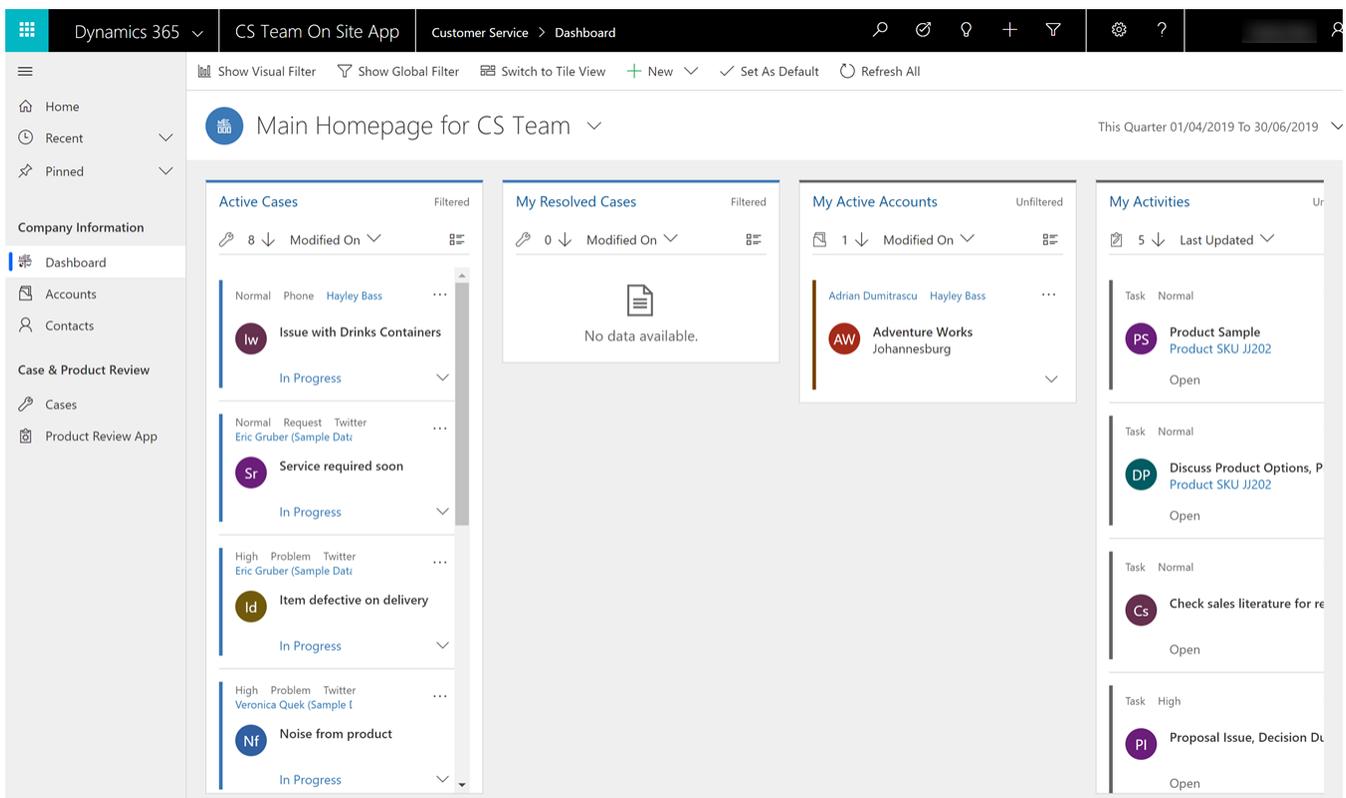
Consider a scenario where a customer service executive for a beverages company meets with key clients to review and create cases on any issues regarding the services they offer. The executive could be conducting the following actions:

- Reviewing key account information and creating new contacts.
- Creating and reviewing cases following a specific customer service process.
- Conducting on-site product reviews

The following screenshots showcase a sample application for the customer service executive with minimal effort. Exact placement of content would be refined based on user feedback and observations.

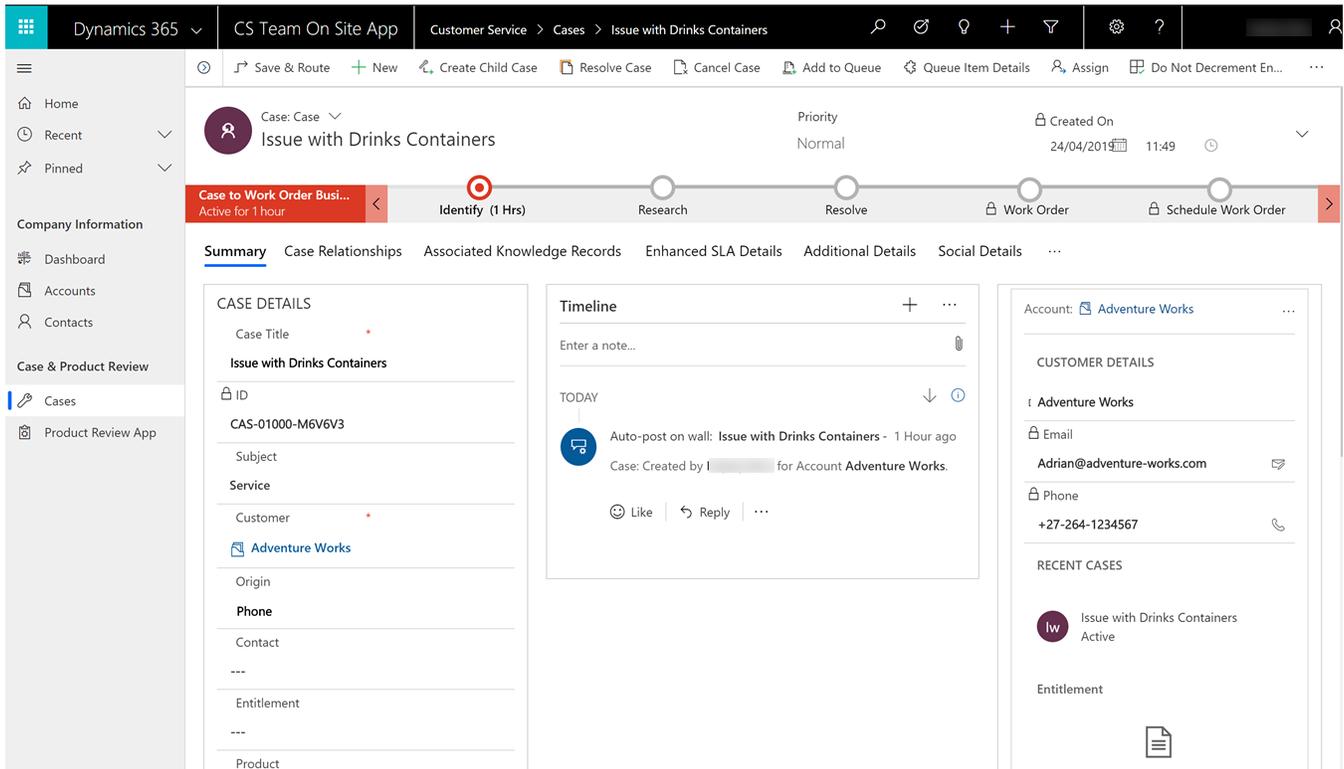
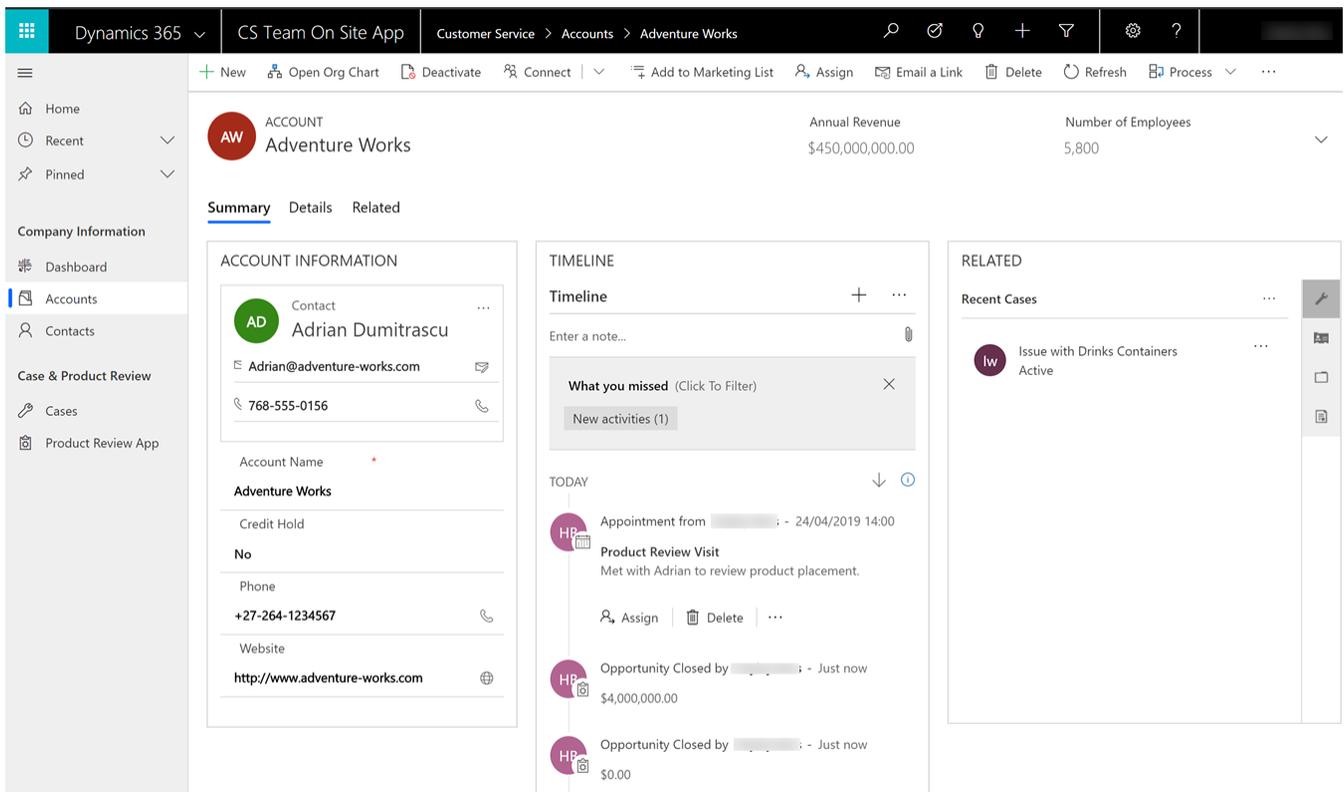
You should note that creating a new application for the customer service executive allows for a slimmed down navigation bar with just the entities they need to reach. It allows the user to focus on the areas they need to work with and eliminates noise from other roles requirements.

When the user logs into the application, the interactive dashboard acts as a default home page to show the data relevant to the job. The executive has the ability to drill down to the record they need without requiring any search or clicking through the navigation bar.



Main dashboard for the customer service executive – this drives their working day.

The main account screen has all the components on the form checked against business requirements and placed into the relevant tab so the user can find them easily. The Timeline is a great productivity feature showing the history against the account and anything the exec might have missed since last login. Additional artificial intelligence could also be added here to guide the user on the conversations they should be addressing with the client.



When the user drills down into the case screen they can see the business process bar has been enabled. This guides the user through the actions that need to take place to complete their customer visit. Within the application designer you can set what dashboard, forms, views, and business processes the user can see. Providing just the minimum under the one application prevents a user from potentially missing some vital

information to be filled in. When approaching the screen design, again, remove all fields except the essential and then work with the business to evolve the form if it makes good business sense.

Summary

With the exciting release of the Unified Interface and the technology enhancements, there is no better opportunity to take stock of your existing implementation or review a new deployment to make sure it is optimum for the business. Before approaching any user experience re-design or a new implementation, think about user personas and objectives. Designing for the most specialized roles yields the best results. It's also important to focus on actionable insights rather than just on data. To measure the success of an implementation, measure outcomes and not actions. Finally, use the right design components in the right place for optimizing the user flow. Following the guidelines in this document can set you on the right path to achieve a valuable and intuitive experience for your end users and customers.

Appendix

Sample Employee Satisfaction Survey Questions

Question	Strongly Disagree	Somewhat Disagree	Neither Agree nor Disagree	Somewhat Agree	Strongly Agree
Overall, I feel that the CRM/customer engagement system we have in place today meets the needs of my daily job role					
When I use the system, I am comfortable I know how to navigate to the correct area					
I find the product easy to search to find information to help me in my daily tasks					
The look and feel of the solution supports my needs					
It is easy to see the history of previous customer interactions from within a customer record					
Often, I have to move between the customer engagement system and e-mail to speak with internal departments about my customer					
I feel confident all the core information I need about my customer is held or accessible within the system					
The dashboards and reports generated are useful for my daily needs					
I have a clear understanding on how my company would like me to utilize the customer engagement system in my job role					
The performance of the system is generally in line with other solutions on my computer					
I feel confident that I know who to ask if I have a question about using the customer engagement system					
Making quick updates to the system is easy to do					
	Open Ended Questions				
What areas of the system work well for you?					
What daily processes do you feel are missing from within the system?					
What is your most used area of the system?					
Are there any information/capabilities that you would like to see included within the system that is not there at the moment?					
Any other comments?					

Specific Area Delivery
Questions

For new feature/capability X, was what you experienced in any way different from what you expected?

Were any of your expectations not met? (Ask how in follow up if req)

Were any of your expectation exceeded? (Ask how in follow up if req)

-