Architecture Views

An architectural view is a specific representation of an architecture that focuses on a particular aspect or set of concerns, tailored for a specific group of stakeholders.

Views are not just diagrams, but tailored representations for specific audiences. The goal isn't visual perfection — it's clarity. These diagrams aren't blueprints; they're storytelling tools.

Views should enabler the viewer to understand the 'why' and the 'how' and not get lost in the wiring. That's the power of a well-shaped view.

Stakeholders need different types of IT architectural view depending on the task at hand. An approach is to break these down into 3 'levels' and label each view as such. Views are compromised of Solution Building Blocks (SBB). Views can then be recast for the audience

Whilst views are categorised based on these levels any individual view may combine aspects of multiple levels if the author so wishes. In these cases, the view is categorised based on 'best fit' as judged by the view author.

Three levels of views:

1. **Outline-level**: Business Applications, Interfaces, Data, General context.
2. **High-level**: IT Components, Associations, Dataflows.
3. **Low-level**: Named IT assets, environments, configurations.

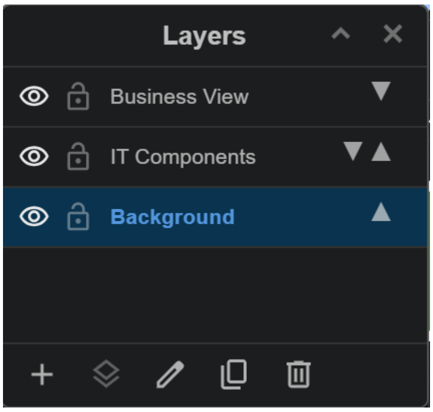
Views can also be reframed to tell a specific story:

- **Business view:** tailored for stakeholders who care about outcomes, not infrastructure. The focus is on roles, outcomes and processes.
- **IT view:** A combination of the business view with an IT view layered in. These views are detailed but still anchored in the business narrative. They focus on components, metadata, and provisioning and show how the components support the outcomes.

This layered approach helps IT understand the architecture deeply, while staying aligned with the broader goals. It's not just about what's in the diagram — it's about why it's there.

Creating Views

Layers are used to define the individual views.



The visibility of layers can then be used when views are exported to PDF for consumption. The process is:

1. Create a master (Architect) view using the full range of Solution Building Blocks (SBB)

2. Apply Connectors (which also double as associations and dataflows)
3. Add supporting text and solution users
4. Assign elements to layers
5. Generate views using the layer control and exporting the result to PDF

Avoid overloading the master view. Use layering to manage complexity and tailor outputs for different audiences.

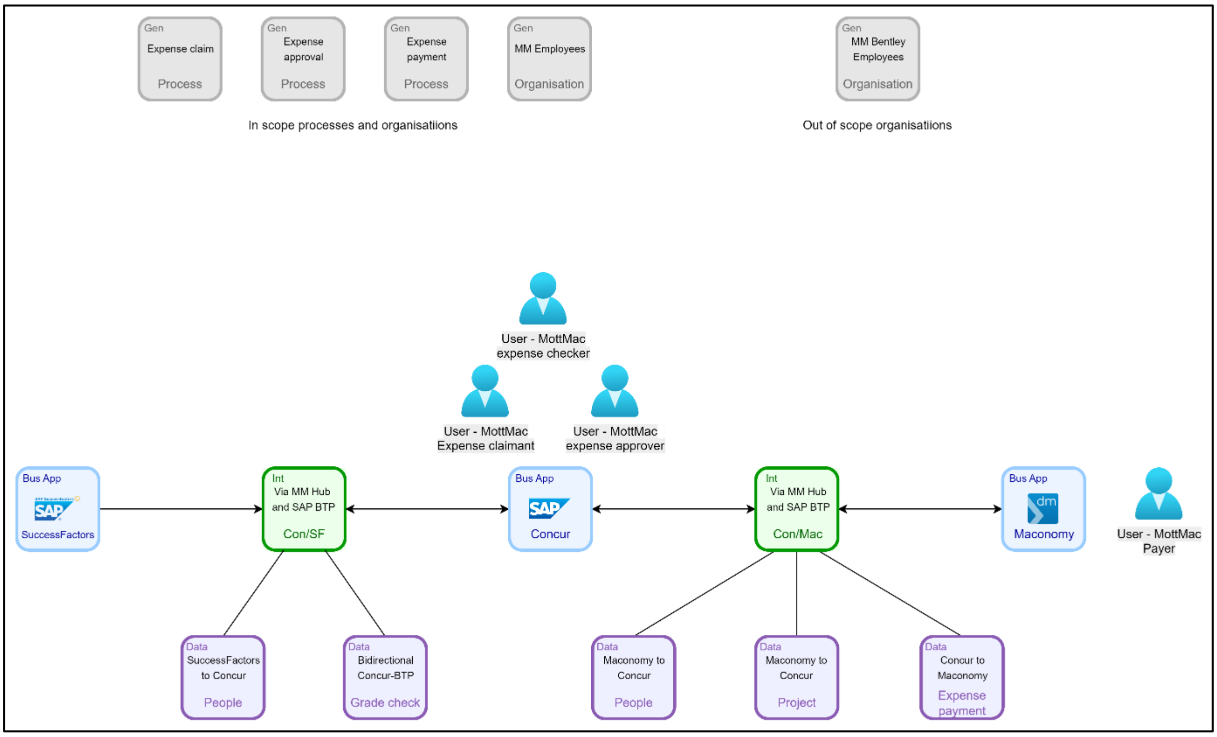
Outline-level

This level specifies the Business Applications involved, their interactions with each other and their relationships with business process and the organisation. These diagrams give important context to any technical design work and are often used to explain complex technical changes in a simpler more abstract manner to business / senior stakeholders.

An architectural view purely focused on communicating an outline-level story will use only the following meta-model entities:

1. Business Application SBBs
2. Interface SBBs
3. Data SBBs
4. General SBBs

Example of an Outline level context diagram: Proposed design for introducing SAP Concur



High-level

This level specifies the IT components (types of IT asset, rather than individual IT assets) used to implement each Business Application and Interface. It also shows the dataflows and associations between these IT components.

High-level views give all technical stakeholders (e.g. IT Leadership; IT Programme & Project Managers, Business Analysts, Operational and Development teams) an overview of the technical scope and complexity of any works required to transition to / operate / transition from technology services based upon these IT components.

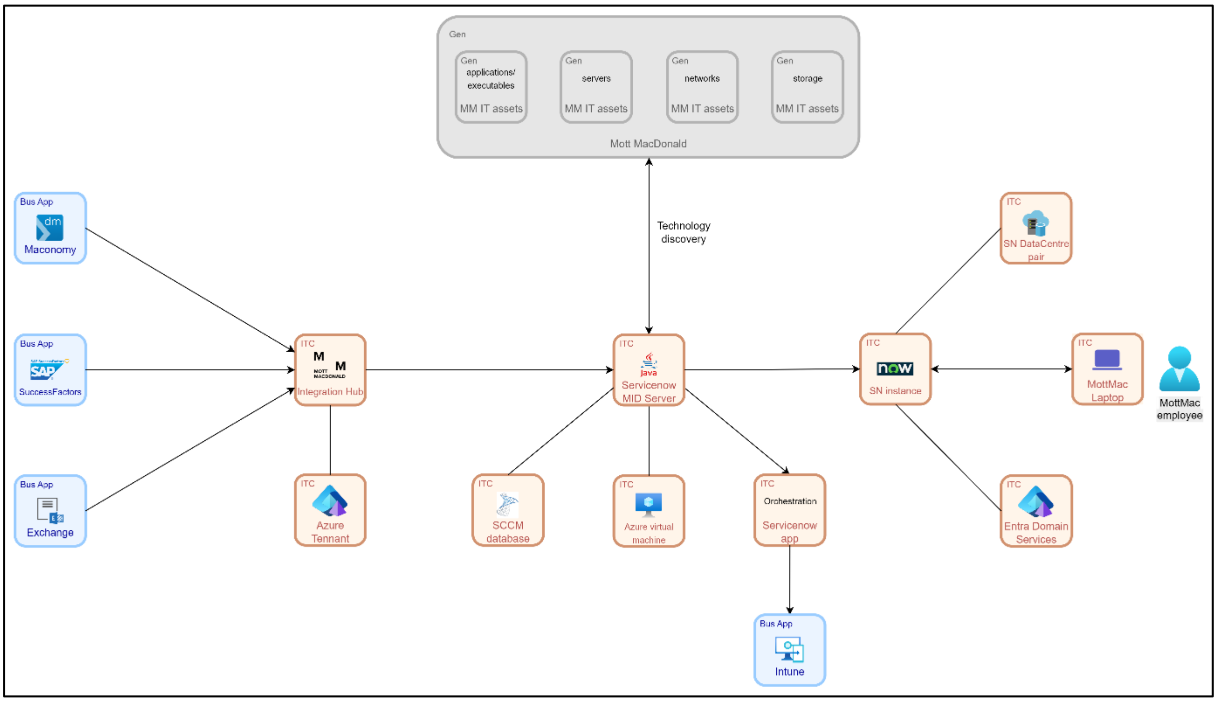
An architectural view purely focused on communicating a high-level view story will have a technology focus and so will typically only use the following meta model entities:

1. IT Component SBBs
2. General SBBs
3. Association SBBs
4. Dataflow SBBs

This view will not contain IT assets and IT asset names

Example of a High-level landscape view: ServiceNow showing key IT Components.

Note additional use of Business Applications to provide context for the key IT components



Low-level

This level usually specifies the named IT assets that will be found in the design. It will often be the level at which multiple environments (dev/test/pre-prod/prod/training/sandbox) are specified as each can be different if lesser investment is to be made into non-production environments.

Low-level views will be largely used by operational and development IT teams and those supporting them.

An architectural view purely focused on communicating a low-level view story will have a focus on IT assets and their configuration and so will only use the following meta-model entities:

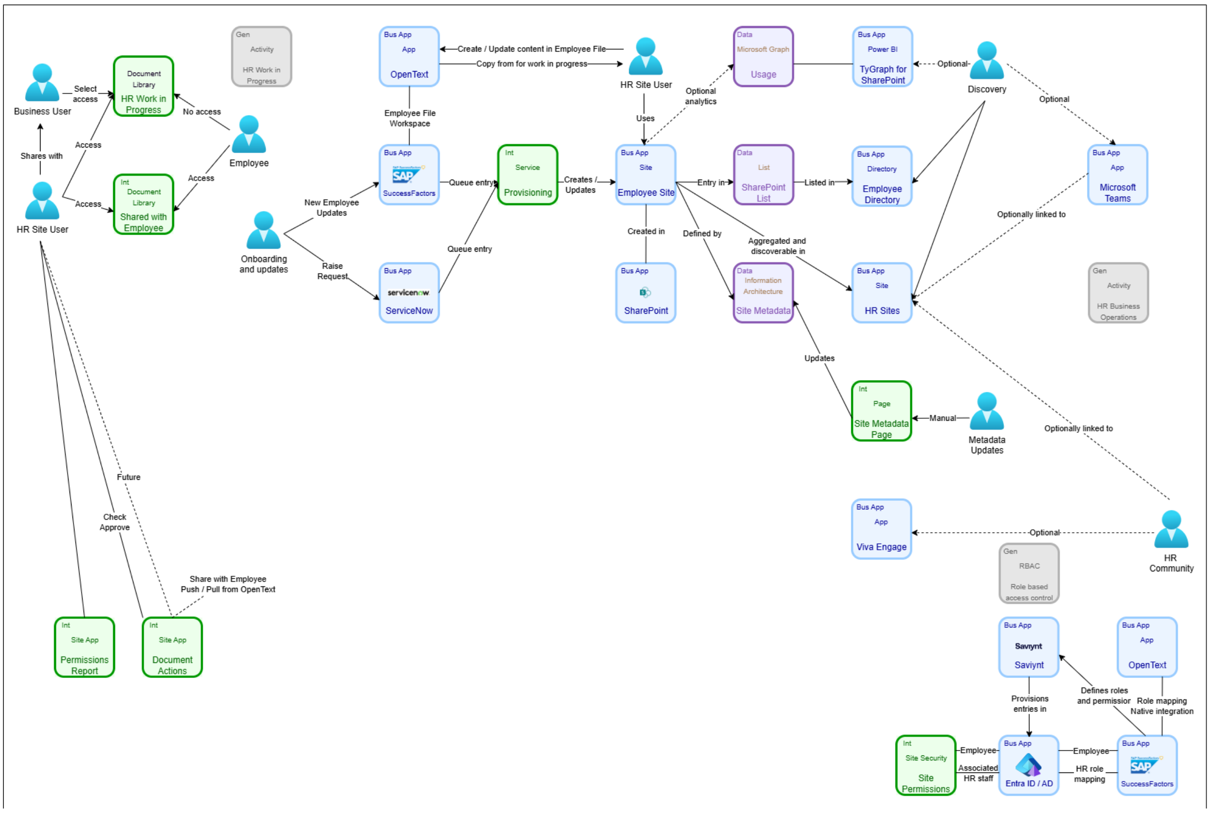
1. IT Component SBBs with specific IT asset names
2. General SBBs
3. Association SBBs
4. Dataflow SBBs

Business view

A business view is tailored for stakeholders who care about outcomes, not infrastructure. This view is about clarity and relevance designed to help stakeholders see how the solution supports their goals.

Example of a Business view: SharePoint Sites supporting SAP SuccessFactors.

This view focuses on the creation of individual sites where HR work associated with the employee will get done. This view includes the onboarding workflows, and integration with systems like SuccessFactors and ServiceNow. Every IT component is not shown, and technical building blocks have been excluded unless they directly impact the user experience.



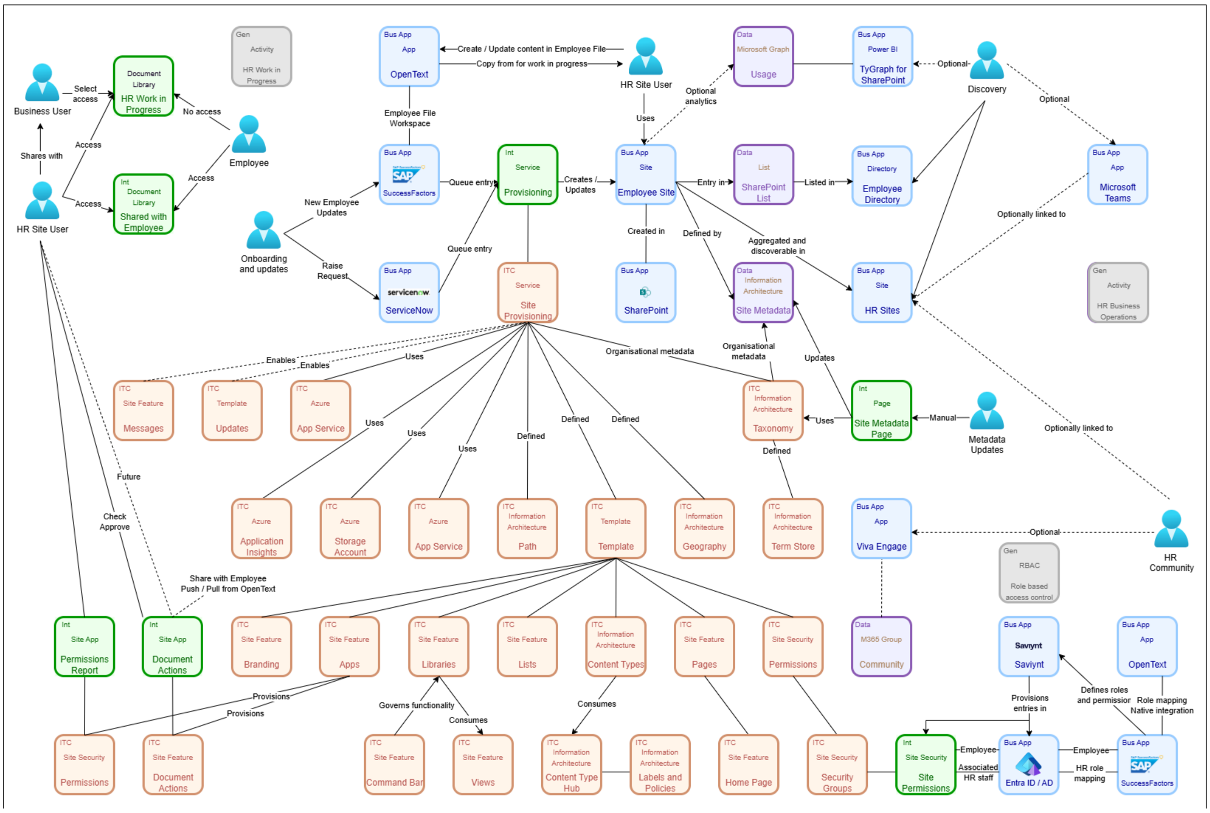
Business User
Select access
Document Library HR Work in Progress
No access
Employee
Shares with
Access
Access
Access
HR Site User
Int Document Library Shared with Employee
Gen Activity HR Work in Progress
Bus App App OpenText
Create / Update content in Employee File
Copy from for work in progress
HR Site User
Employee File Workspace
Bus App SAP SuccessFactors
Queue entry
Int Service Provisioning
Creates / Updates
Onboarding and updates
New Employee Updates
Raise Request
Bus App servicenow ServiceNow
Queue entry
Uses
Optional analytics
Data Microsoft Graph Usage
Bus App Power BI TyGraph for SharePoint
Optional
Discovery
Optional
Bus App Site Employee Site
Entry in
Data List SharePoint List
Listed in
Bus App Directory Employee Directory
Bus App App Microsoft Teams
Optionally linked to
Created in
Defined by
Aggregated and discoverable in
Bus App SharePoint
Data Information Architecture Site Metadata
Bus App Site HR Sites
Gen Activity HR Business Operations
Updates
Int Page Site Metadata Page
Manual
Metadata Updates
Optionally linked to
Future
Check Approve
Share with Employee Push / Pull from OpenText
Bus App App Viva Engage
Optional
HR Community
Gen RBAC Role based access control
Int Site App Permissions Report
Int Site App Document Actions
Bus App Saviynt Saviynt
Bus App App OpenText
Provisions entries in
Defines roles and permissior
Role mapping Native integration
Int Site Security Site Permissions
Employee
Associated HR staff
Bus App Entra ID / AD
Employee
HR role mapping
Bus App SAP SuccessFactors

IT view

The IT view presents the same architecture but with a technical focus. It's detailed, but still anchored in the business narrative.

Example of a IT view: SharePoint Sites supporting SAP SuccessFactors.

This view adds detail to the corresponding business view. This view includes user interfaces, reporting, provisioning templates, taxonomy, and security including Role Based Access. Even so, not every IT component is shown not because it is about connection back to the business value and needs.


Business User
Select access
Document Library
HR Work in Progress
No access
Employee
Shares with
Access
HR Site User
Access
Int
Document Library
Shared with Employee
Access
Gen
Activity
HR Work in Progress
Bus App
App
OpenText
Create / Update content in Employee File
Copy from for work in progress
HR Site User
Employee File Workspace
Bus App
SuccessFactors
Queue entry
New Employee Updates
Onboarding and updates
Raise Request
Bus App
servicenow
ServiceNow
Queue entry
Int
Service
Provisioning
Creates / Updates
Uses
Optional analytics
Bus App
Site
Employee Site
Created in
Bus App
SharePoint
Data
Microsoft Graph
Usage
Bus App
Power BI
TyGraph for SharePoint
Optional
Discovery
Optional
Entry in
Data
List
SharePoint List
Listed in
Bus App
Directory
Employee Directory
Bus App
App
Microsoft Teams
Optionally linked to
Defined by
Aggregated and discoverable in
Data
Information Architecture
Site Metadata
Bus App
Site
HR Sites
Gen
Activity
HR Business Operations
ITC
Service
Site Provisioning
Enables
Enables
Uses
Organisational metadata
Organisational metadata
Updates
ITC
Site Feature
Messages
ITC
Template
Updates
ITC
Azure
App Service
ITC
Information Architecture
Taxonomy
Uses
Int
Page
Site Metadata Page
Manual
Metadata Updates
Optionally linked to
Uses
Uses
Uses
Defined
Defined
Defined
Defined
Future
Check Approve
ITC
Azure
Application Insights
ITC
Azure
Storage Account
ITC
Azure
App Service
ITC
Information Architecture
Path
ITC
Template
Template
ITC
Information Architecture
Geography
ITC
Information Architecture
Term Store
Bus App
App
Viva Engage
Optional
HR Community
Gen
RBAC
Role based access control
Share with Employee Push / Pull from OpenText
Int
Site App
Permissions Report
Int
Site App
Document Actions
ITC
Site Feature
Branding
ITC
Site Feature
Apps
ITC
Site Feature
Libraries
ITC
Site Feature
Lists
ITC
Information Architecture
Content Types
ITC
Site Feature
Pages
ITC
Site Security
Permissions
Data
M365 Group
Community
Bus App
Saviynt
Saviynt
Bus App
App
OpenText
Provisions
Provisions
Governs functionality
Consumes
Consumes
Provisions entries in
Defines roles and permissior
Role mapping Native integration
ITC
Site Security
Permissions
ITC
Site Feature
Document Actions
ITC
Site Feature
Command Bar
ITC
Site Feature
Views
ITC
Information Architecture
Content Type Hub
ITC
Information Architecture
Labels and Policies
ITC
Site Feature
Home Page
ITC
Site Security
Security Groups
Int
Site Security
Site Permissions
Employee
Associated HR staff
Bus App
Entra ID / AD
Employee
HR role mapping
Bus App
SuccessFactors