servicenow

Information Management Implementation Guide using CSDM 5

How to align your information management and data governance practices with the ServiceNow Common Service Data Model v.5

v.1.0

Kristine Naess, Senior Advisory Enterprise Architect, ServiceNow EMEA

Rob Koeten, Engineering Fellow, ServiceNow

servicenow

Table of Contents

Why organisations must know what and where data lives in their IT systems

The question of what information — be it personal data, confidential production data, or simply any data—runs through which IT system is no longer just a matter of good housekeeping—it has become a fundamental governance requirement and a strategic imperative.

The last years have shown us that countries that have been allies can become threat actors, and countries who have had piece for decades can suddenly become war zones. The political climate in a country can change so fast and introduce new laws and regulations at such a speed that the need for exit if your information relies somehow on those countries must be well planned for and preferably tested in advance. Becoming aware of your inventory of important assets is the first step to achieve this and using a data model that encompasses all types of assets, configuration items, services and foundational data will be an important instrument in doing this. The Common Service Data is exactly this and is steadily capturing new perspectives and elements as the digital industry moves on.

As organisations increasingly rely on interconnected digital infrastructures, the ability to trace, document, and control information across systems has moved from a compliance checkbox to a core operational capability.

The European Union's evolving Digital Single Market framework has accelerated this shift. While the General Data Protection Regulation (GDPR), in force since 2018, established the foundational obligation for organisations to understand and document their data processing activities, subsequent legislation has expanded the scope considerably. The EU Data Act, which became applicable in September 2025, now extends data access and portability rights to both personal and non-personal data generated by connected devices—from smart vehicles to industrial machinery. Meanwhile, the European Commission's November 2025 Digital Omnibus Package proposes further amendments to the GDPR and related regulations, refining definitions of personal data and streamlining enforcement mechanisms for cross-border cases.

With the publication of the Cloud and AI Development Act (CADA) in June 2026, an additional focus on data sovereignty as well as digital autonomy for EU and EEC countries was introduced. It is still not certain what will happen until the regulation is ratified, but establishing exit strategies and data sovereignty measures will most likely be part of what society critical enterprises must take on in the near future. Being able to fulfil critical corporate missions with full access to needed data, without the usage of foreign technology is a huge endeavour. It calls for full oversight over not only where data resides and is being processed, but also on the technology with which it is being processed. And the ultimate exit to open-source technology to take over this storage and processing activity will need years to finish.

This layered regulatory architecture creates a complex web of obligations. Organisations must not only demonstrate lawful processing under the GDPR but also ensure data accessibility under the Data Act, maintain interoperability for cloud switching, and prepare for AI-specific requirements under the EU AI Act. Each of these frameworks presupposes one fundamental capability: knowing precisely what personal data exists, where it resides, and how it moves through an organisation's IT landscape.

Information confidentiality such as product secrets, patents, price calculations, customer data, etc. is crucial for company survival because it protects competitive advantages, trade secrets, and strategic plans from rivals. Breaches can lead to lost market share, damaged reputation, legal liabilities, and financial losses. Maintaining confidentiality preserves customer trust, ensures regulatory compliance, and safeguards the intellectual property that drives business success.

Without this visibility, compliance becomes guesswork. With it, organisations gain not only regulatory confidence but also the foundation for data-driven innovation, operational efficiency, and the trust of customers and partners operating within the worldwide increasingly integrated digital economy.

With the introduction of agentic AI and LLMs the volume of data processing has exploded. Suddenly we see a large potential climate impact as well as battle for energy sources. Figuring out where your LLMs pull computing power and cooling from can be another factor to control and may come as a wanted side effect of controlling data residency.

In this implementation guide we seek to help those who are starting out on the complex endeavour to gain control and oversight of where information is stored, how it is being used and protected, as well as secured and backed up to ensure operational resilience. But hopefully also cover some mechanisms on how information should be understood as part of this endeavour.

Understanding ‘Information’

While focusing on overall Information Management it is critical to understand what is defined as Information, how and where it exists within and outside the ServiceNow platform, as well as where Information fits in the Data-Information-Knowledge-Wisdom continuum.

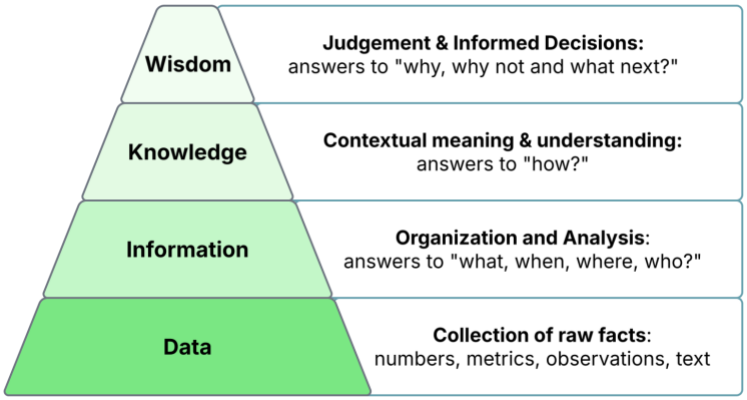

Figure 1 -The Data-Information-Knowledge-Wisdom (DIKW) pyramid

Data

Data is the raw, unprocessed, and uncontextualized representation of facts, measurements, observations, or signals — existing as discrete symbols, numbers, characters, or binary values that carry no inherent meaning in isolation. Data is the foundational substrate of information systems: it is produced by transactions, sensors, instruments, and interactions, and persists in storage as records, files, streams, or messages. In its raw state, data has no semantic content. E.g. a measurement or datum of 90 has meaning without context.

Information

Information is data that has been given meaning through contextualization, structure, and interpretation — transformed from raw signals into coherent, purposeful content that answers a question, describes a state, or supports a decision. Information emerges when data is organized within a frame of reference, including context and purpose. The earlier data-point of 95 gets meaning and understanding when it’s associated with human body temperature, human age, ROI return percentage, or SaaS Availability SLA. It may also carry properties such as accuracy, completeness, timeliness, and relevance that determine its fitness for use.

Knowledge

Knowledge is information that has been absorbed, validated, internalized, and integrated into a structured understanding that enables reasoning, judgment, and purposeful action. Knowledge goes beyond describing what is true to encompassing why it is true, how things relate, what patterns hold, and what consequences follow — it is information enriched by experience, inference, causal understanding, and contextual expertise. Knowledge exists in two primary forms:

- Explicit knowledge, which is articulated, documented, and transferable through language and formal representations such as ontologies, models, and procedures
- tacit knowledge, which is embodied in skills, intuitions, and expertise that resist full formalization.

In computational and enterprise architecture terms, knowledge is the layer at which semantic relationships, inference rules, and formal ontologies operate — enabling machines and humans alike to reason beyond the immediate data and draw conclusions that the data alone does not directly state.

Wisdom

Wisdom is the capacity to apply knowledge with sound judgment, ethical discernment, and long-term perspective to make decisions that are not only effective but right. Where knowledge answers *how* and *why*, wisdom answers *what should be done* — integrating accumulated knowledge with values, experience, foresight, and an understanding of consequences across time, context, and stakeholder impact.

Wisdom is inherently normative rather than merely descriptive: it encompasses the ability to recognize what matters most in a given situation, to navigate ambiguity and competing priorities, to anticipate second-order effects, and to act in ways that are

coherent with enduring principles rather than immediate pressures.
In organizational and enterprise architecture terms, wisdom manifests as the institutional judgment that shapes strategy, governs the application of AI and automation, arbitrates between competing architectural principles, and determines when rules should be followed and when they require exception — making it the ultimate governing layer above data, information, and knowledge in any system that aspires to be not merely intelligent but genuinely purposeful.

Information and the ServiceNow AI platform

How does this apply to the ServiceNow AI platform and the AI-native products, applications and services that it provides and integrates? Figure 2 depicts how the different user and AI-native experiences enable and/or act upon the semantic knowledge, its underlying information and the data that the platform persists, ingests and/or integrates.

- Knowledge and Context is made available to a range of user experiences models, including legacy Forms, Lists & Workflows, Portals and Workspaces and various AI-native Assists, AI-Agents, AI-Specialists & AI-conversations. The Business Semantic Ontology provides a well-structured, goverened semantic model that LLMs can traverse, act upon and reason about.
- Knowledge and Context supporting Information is aggregated and/or established from business domain data (action oriented data as well as business records, business events and reference data) and analytics data. This applies to both SN internal as well as data integrated from external sources.

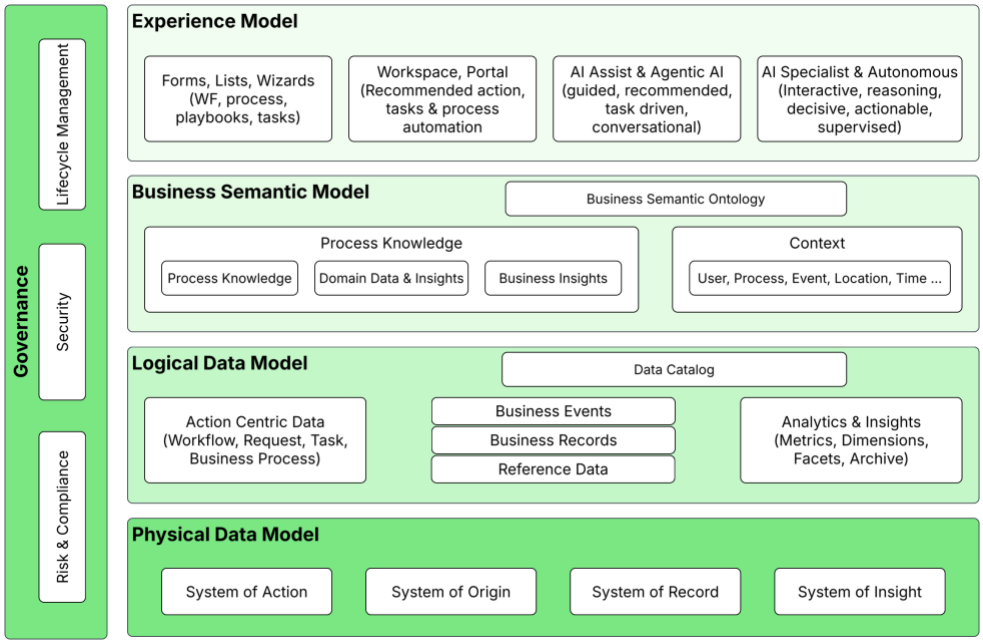

Figure 2 -ServiceNow Knowledge, Information and Data architecture..

To map this information architecture to the CSDM reference architecture we need to introduce a cross-domain information management representation: the Information and Knowledge Layer.

The Information & Knowledge Layer

The **Information & Knowledge Layer** is the architectural layer that describes the semantic, structural, governance, and lifecycle properties of information as an enterprise Information Asset — independent of the business processes that create or consume it, and independent of the applications or infrastructure that store or transmit it. It is the layer at which *meaning*, *policy*, *lineage*, *classification*, and *semantic relationships* are formally modeled and governed.

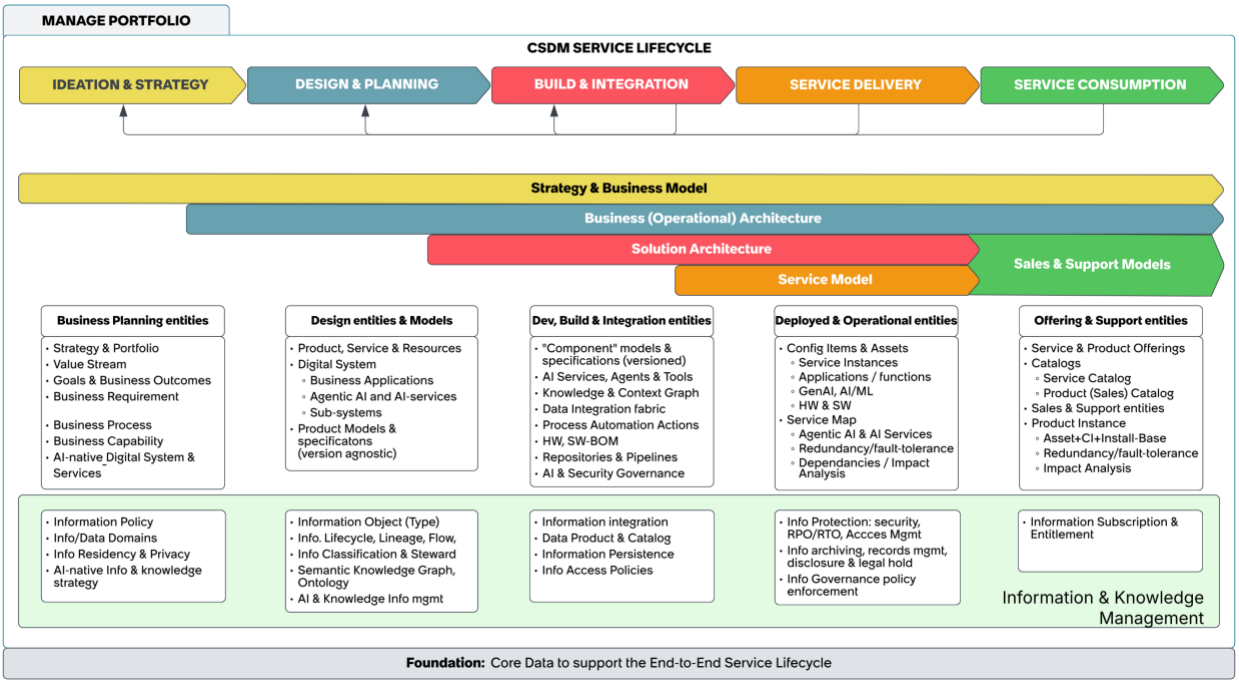

Figure 3 -Information Management mapped across the CSDM domains

While the Strategy, Business, Application (services & function) and Infrastructure have been described and mapped in CSDM, the Information & Knowledge layer is introduced for the explicit purpose of defining Information Management capabilities and patterns. Both Business and Information & Knowledge layers are explicitly associated with the CSDM Planning & Design domain, reflecting their origin as Business & Enterprise Architecture design entities.

Data lineage

Data lineage is the ability to trace where data comes from, how it moves, and who has touched it — from origin to use. For organizations managing sensitive information under GDPR, FHIR, or internal governance policies, this is no longer optional. ServiceNow acts as the operational layer: managing workflows, access, incidents, and compliance processes that govern data in motion. With the acquisition of Data.World we can provide a much broader platform as the Data.World part acts as the intelligence layer: cataloguing data assets, documenting lineage, and making data discoverable across the organization. Together, we close the gap between knowing what your data governance policies say and proving that they are being followed.

A data lineage is a documented record of:

- Where data was created or ingested
- Which systems it has passed through
- Who accessed or modified it, and when
- What it was used for

In the following chapter we will seek to clarify how the ServiceNow platform can assist you in gaining these insights as well as show how CSDM covers the universe of information and data.

High level architecture artifacts

Information does not exist in a vacuum, rather it is a main ingredient in what we do and aim to achieve. To understand the main objects in the Common Service Data Model, it is often useful to align them with a more generic interpretation of the high-level business architecture. Alongside goals and strategies, the very outcome of what we do, or what we need to produce as an outcome, can be summarized in a structure of Business Capabilities.

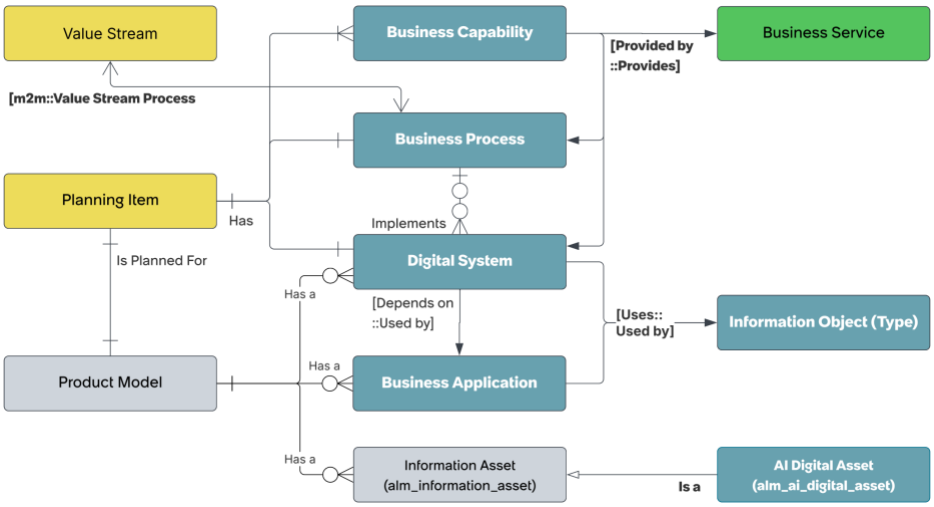

Figure 4 – Key relevant data entities for the governance of your Information Object(Type)s and for providing a context within which they are used and maintained.

They define what value we must create, rather than describing how, by whom and using what tools. The 'how' can be modelled as Business Processes, which can be building blocks in Value Streams, and have detailed process activities to define them more granularly. The Business Process objects are where we usually add policies to describe what we are allowed to do or how we should do things. They are thus normative. But they can also represent automated or manual steps taken to achieve a certain outcome and are descriptive of those activities or actions.

The Business Services describe the organization and accountability to produce a needed outcome, but also the level of quality and customer experience one can anticipate. If you need to pay for receiving the service, this is where cost is shown. It is also here the expected experience of receiving the services offered is accounted for, as well as the expected quality of them. Service Level Agreements on the offerings are examples of this. There isn't any direct link between a business service and an Information Object (Type). If the Information **is** the service, this will involve using other entities such as Products and Offerings.

The Business Applications summarize the "digital tooling" needed to produce the wanted outcomes. These **use** the Information Object (Type) as input source, for processing and as output. If the business capability needs digital tooling, a landscape of Technology Management services enters the scene. They are the ones that ensures that the underpinning technology is kept at an acceptable level so the Business Services can perform as they should and that you will ultimately uphold your business capabilities. We will get back to this later.

Design & Planning - Information as a Design artifact

To view how information is used in multiple contexts, you may want to create design objects for them. These are the Information Object (Type), belonging to Data Domains. You should have a purpose for collecting, storing and using information. You can use the Business Capabilities to define what you need to be able to achieve on a high level, i.e. the needed outcome of your business. To do so, you need to have at least one Business Application as the Business Applications use the Information Object (Type) to provide Business Capabilities:

Figure 5 – Information Objects (Types) as something that is used by an application to provide a certain capability.

The Information Object (Type) records are the correct ones to use for classification and categorisation. You assess the criticality and confidentiality of your information to be able to take the correct protective measures to secure necessary availability, integrity and confidentialty on the data at rest and in transit. You can read more about this in the chapter **CLASSIFYING AND CATEGORIZING INFORMATION OBJECTS.**

Business processes using Information Objects

You can also use the Business Process records to state in what activity the data is used, and what business application the Business Process needs to perform its activities.

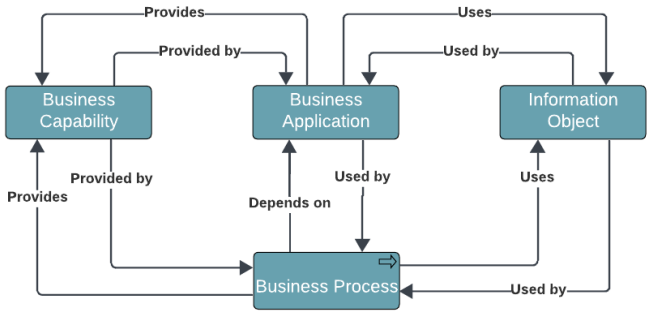

Figure 6 – The relationship types and directions are utilized to map out potential dependencies and the utilization of each individual artefact in a larger context..

Content Product Models

The Information Object (Type) records don't keep track of versions. That belongs to the product dimension. And how many copies you have stored of each of them, belongs to the asset dimension.

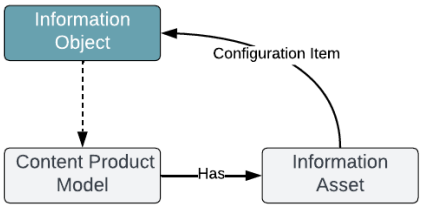

Figure 7 – The Information may be given a Content Product Model record to keep track of versions and life cycles of the data it consists of. And one or more assets to show back individual items of these.

We have now seen that the Information Object (Type) has an asset dimension. That means that we ultimately should find traces of these within the asset table structure. Think of it as the occurrence of information content of a certain type, stored somewhere. Like a customer registry where you have collected useful information about your customers for years, placed in a database, an email solution or in a spreadsheet. Or a design document displaying a patented product you manufacture, stored

within an CAD-tool. It may also be a collection of strategy documents about a planned acquisition of a company you don't want your competitors to know about, stored in as presentations on a file share.

The information needs to be protected from unauthorized access, be kept so that it is available to those who needs it when they need it and protected from becoming corrupted or otherwise made untrustworthy.

Information Object (Type) and Data Domain

In CSDM the high-level entity we use to define information content is the Information Object (Type) [cmdb_ci_information_object]**.** It's better to think of these as "registries" or bulks of information content types that share some similar traits or purposes for use, than as individual information elements. For example, if you have information content about your customers stored in a database, you will rather define Customer Registry as one Information Object Type, than having Customer Names as one, and Customer Email Addresses, as another. Having said that, there can be reasons why you would like to maintain a granular structure on your Information Objects, for example if you have introduced domain separation between information elements on field or column level, and maybe even set up API services for each of these. The rule of thumb is to define these as granular as your system and service architecture demands, without losing yourselves in maintenance tasks but rather keep in control.

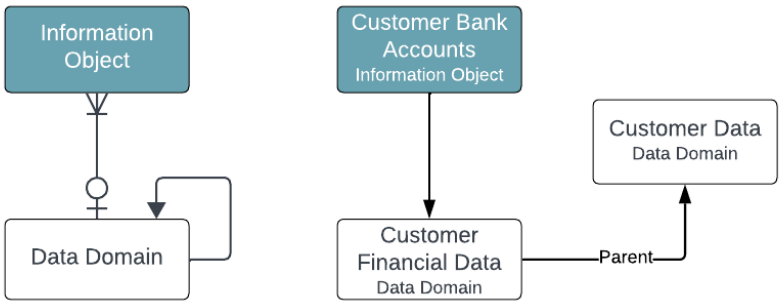

Figure 8 – An Information Object (Type) may reference one Data Domain only. The Data Domains have a hierarchical structure.

For those who have the ServiceNow Enterprise Architecture product, you can add **Data Domains** [sn_apm_data_domain] to your Information Objects. A data domain is a logical grouping of related Information Objects (Types) that represent a specific area of a business- such as Finance or HR. Data domains enable you to analyse data in a way that aligns with business capabilities and operational needs. Data domains enable you to analyse data in a way that aligns with business capabilities and operational needs. These are useful "wrappers" that can help you organize the governance of information content that naturally belong together. They come in hierarchies, and one Information Object (Type) cannot belong to multiple domains. Hence the need to divide the Information Objects (Types) that have two or more "competing" data domains. Being logical groupings rather than pointing to specific content, the data domains don't have owners. Thereby this can be a means of joining information under the same "umbrella" despite containing subsets belong to different owners across your organization. For example, if you define Customer Contact Registry as an Information Object (Type), you can have Customer Support Data as a data domain, with a parent called Customer Data.

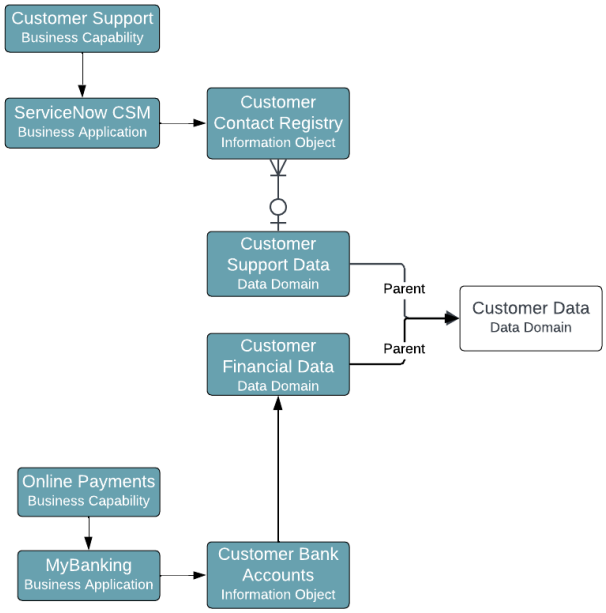

Figure 9 – Data with similar characteristics or usage, but which are owned and governed by different persons and parts of the organization, should be represented individually but can have the same parent Data Domain.

Ideation & Strategy – Information as strategic artefacts

Information is crucial for all processes and activities in a company. Gaining insight into relevant factors that can improve business outcomes, understand your customers and service consumers and leverage analytics tools to foresee and prevent risks are some examples of this. Modelling how information flows in your most critical value streams to ensure that it is being utilised in the best possible way is another important element.

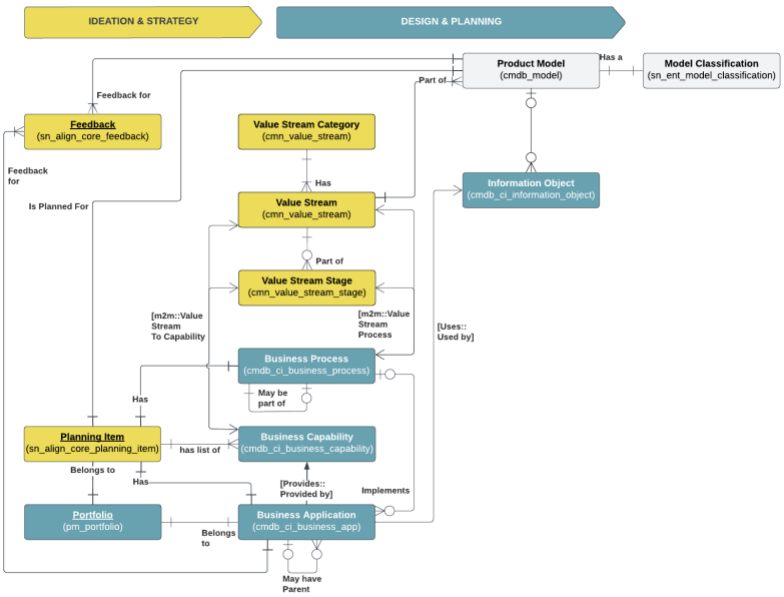

Figure 10 – Combining insight from your architecture artefacts to run product improvements.

You may have received feedback from your consumers on how your information containing product can be improved to better align with business needs, and you need to assess internal policies and the architecture before you can add new types of information or reuse existing for new purposes.

If you use the Strategic Portfolio Management product in ServiceNow, you may want to set up all your Goals with Targets for each of those and start to monitor how all of your Initiatives are meeting them.

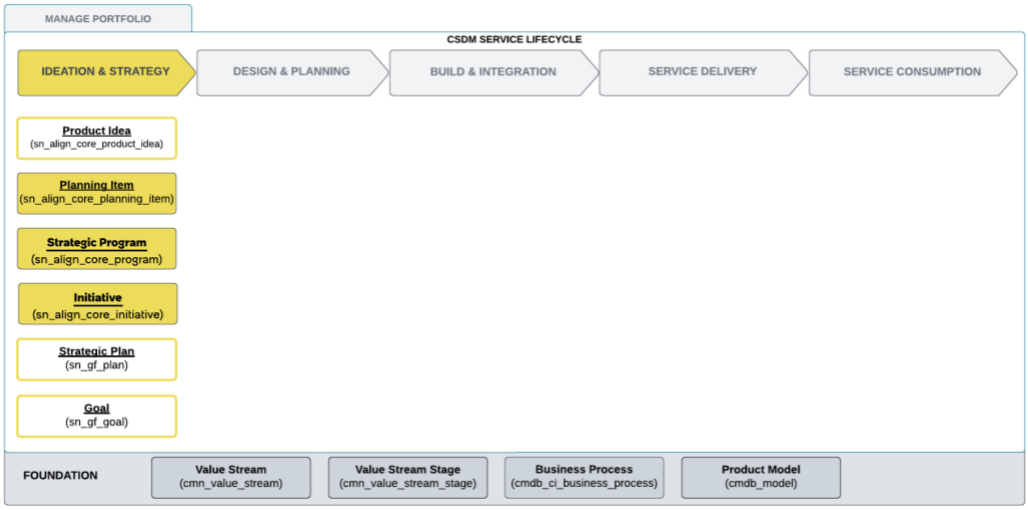

Figure 11 – Arranging your goals and strategies with ideas and plans

Planning the creation and improvement of information products

Even information has a life cycle, and it usually starts with an idea on what information you need to gather, store and provide digitally. As soon as the idea has formalised into a demand, epic or other scope description, you will want to start planning when to construct it and who to do the job.

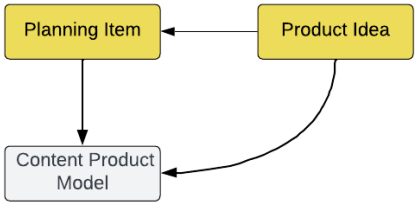

Figure 12 – Ideas and Planning Items can be related to any product model, also Content Product Models.

If you need a new or improved Information Object (Type), you can relate both Product Ideas and Planning items to Content Product Models to isolate it from them from the application product models you may later need to use to process the information. An entirely new Information Object (Type) would have a content product model with a life cycle stage set to 'Ideation' to show that it is not ready for use yet. There aren't any out of the box product model categories set up for general information content, so you must either configure this allow it to stay blank. If this is part of a planned investment, you may even want to capex the effort using Information Assets with a Fixed Asset record to calculate based on decpreciation and amortization rules.

Information as Capexable assets

Some organizations buy information content from a content provider. This may represent operational expense, but in some cases the injected data can have value beyond the fiscal year it was purchased and can therefore be placed on the balance sheet as capital expenses. The content you want to train you LLM on can be an example of a capexable cost. For this you can leverage an asset record and place an expense line on it. To keep track of depreciation and residual value, you can create a related Fixed Asset record as seen in the figure below:

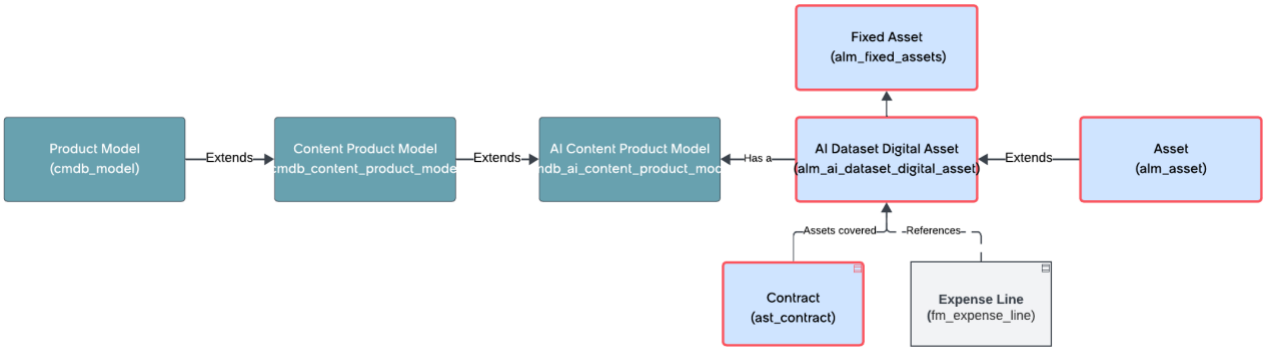

Figure 13 – The product structure provides the commodity context of a digital asset, whereas the Fixed Asset allow you to amortize and/or depreciate one or more assets following a set of business policies.

Contract rate cards (related to contract records) and expense lines are among the building blocks you may want to use to gather cost and show these back to auditors later to prove that the capitalisation is calculated correctly.

Value Streams

If your information is important in a long line of interdependent activities, potentially crossing internal and/or external boundaries, you can use Value Streams to capture this.

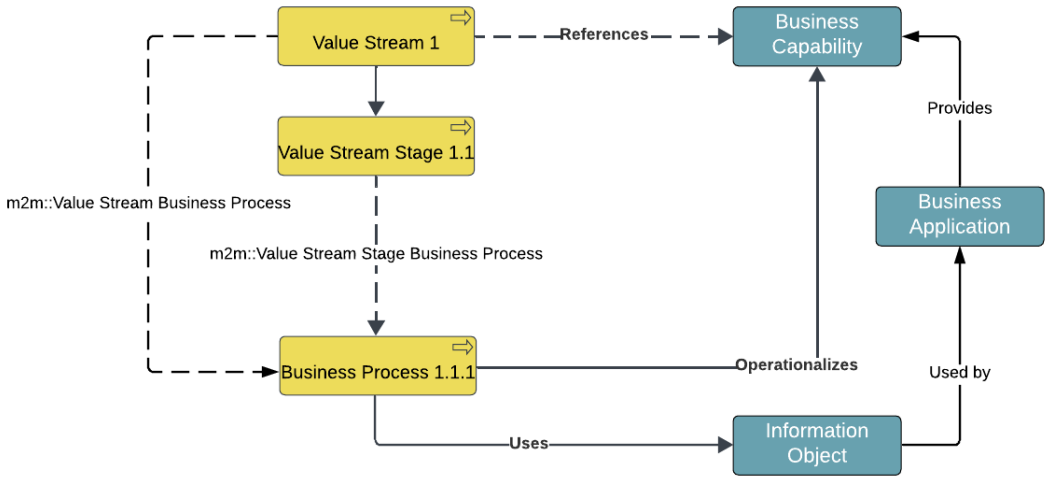

Figure 14 – The Value Streams allows you to place activities in chronological order and to show back responsibilities across organizations.

Note that there is no real relationship between a Value Stream and Business Capabilities and Information Object (Type), only references, as Value Stream is not a CI class. It needs the Business Process to show how it is involved.

Service Consumption - Information as a Commodity

"Information is gold" is a phrase often heard. And it is so true. If you haven't got correct and relevant information at hand, you are in the blind. As a result of the need for digital information, it has become a commodity that is often traded as such. Meaning that you can buy the *actual content*, not just subscribe to an API or other interface to continue receive information.

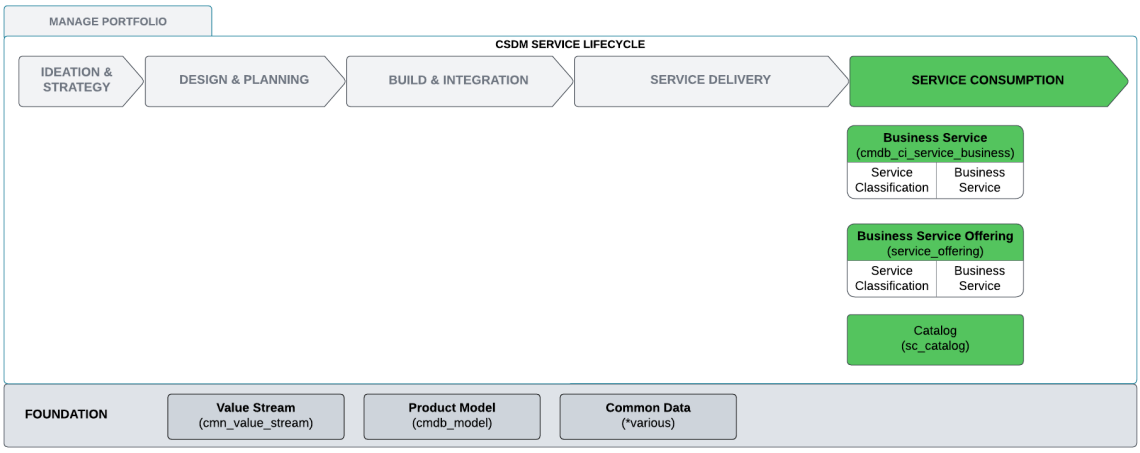

Figure 15 -The Service Consumption domain has core components for showing what and how products and services can be, and are consumed.

When you buy information in bulk, the information can be added a monetary value. This is something we often see when a company has gone bankrupt, and another company wants to buy the residual value of it. Then the customer registry is part of this due diligence and asset assessment process.

More often, of course, information is purchased as a continuous flow of data across a digital integration of some sort, alternatively as a file transfer. And with the introduction of laws and directives such as PSD2 and GDPR, providing information in digitally consumable formats are sometimes mandatory. Even controlled to ensure that it is being provided according to certain time limits and that it is correct.

Information in digital form doesn't just magically appear when you need it, someone needs to make sure you are able to consume it. Providing you the information as a one-off product, in the shape of an export to a csv-file or other format is one way of getting it, but in most situations, you will receive access to the information as a subscription and/or a digital integration. In all three cases, the data as a product will be modelled as a product model, and for you to be able to self-service you will be able to select it from a portal experience as a catalogue item. Your request will go through an approval process which can be designed to differentiate between data products that are classified as Open, Internal, Confidential or Strictly Confidential.

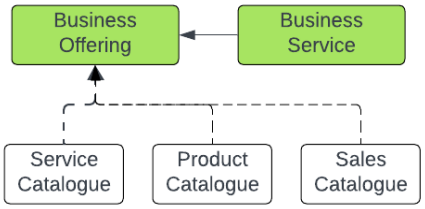

Figure 16 – The Product catalogue contains all the records in the cmdb_model table. The products you want to offer as part of your internal services can become items in the Service Catalogue whereas those you sell externally belong to the Sales Catalogue.

Data catalogues

With the acquisition of Data.World, the ServiceNow universe has expanded radically into the domain of data catalogues and knowledge graphs. A new chapter describing this will be added in Q4 2026, as this is the official launch of many of the features

that tie this into the CSDM. For now, let's have a look at the high-level information asset consumption you can use CSDM to model and govern.

An information asset should reference a product model. And if you want to enable self-service, creating a catalogue item for each of those is a good idea. If your catalogue items can be ordered by someone outside your organisation, you may want to place them in the sales catalogue. If they are meant for internal users you can offer them as catalogue items in the service catalogue. If there is no self-service option set up and they only exist in the product model table as records, you can still manage their life cycle, specify versions and add them as references from CIs in the CMDB.

To make your information assets consumable via self-service mechanisms, you need to create catalogue items for them.

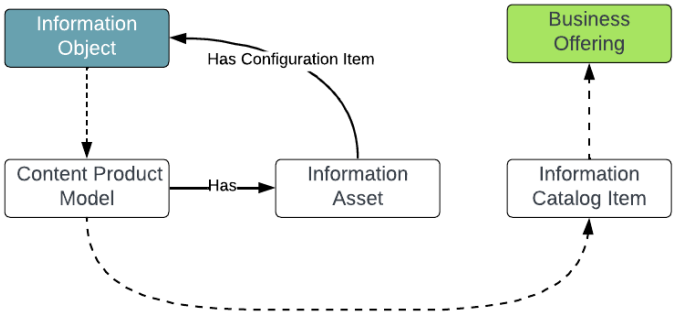

Figure 17 – How an Information Object (Type) may have a product equivalent that can be consumed via a catalog item governed by a business offering.

This brings along the need for having various other perspectives on information management in addition to the ones mentioned in the introduction. Not only on the *management of information*, but also on the concepts defining what *information is* in the eyes of different stakeholders.

Digital Integrations Management

Throughout the years most companies have realized that the number of integrations between applications have grown out of hand, so they have lost control of the totality and life cycle of these. "Shadow integrations" have taken over from the former "shadow IT", as the abundance of available APIs and ready-made spokes have made it so easy to connect that developers fall for the temptations to "just do it" rather than wait a year for permission to properly onboard the integration platform they were supposed to use.
Rather than closing these, it would be better to start governing them and have a realistic plan to phase them out if they bring along security vulnerabilities or legacy technology that's hard to maintain.

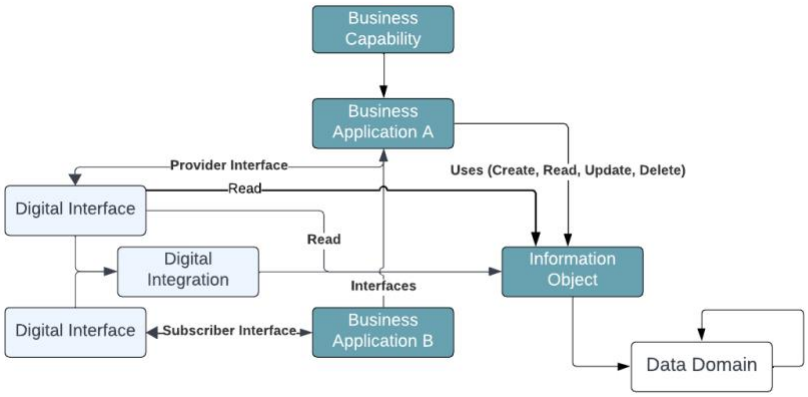

Figure 18 -Digital interface records have one or more digital integration records to show the life cycle of usage of those.

As there are different needs for how to process information, most organisations create APIs or other interfaces that are fit for purpose. Some provide read access while some allow more privileges.

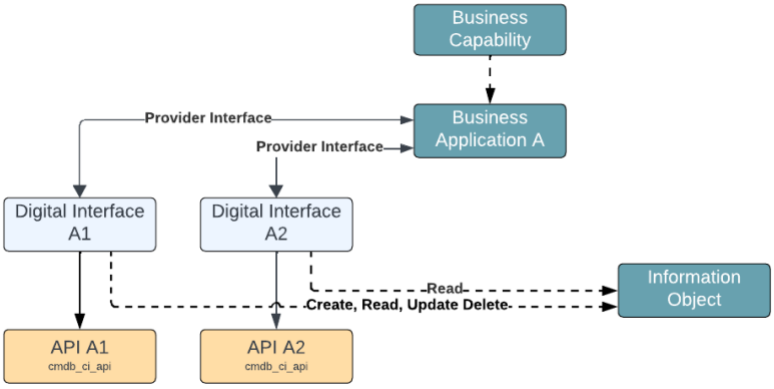

Figure 19 – By relating the digital interface records to the discoverable API configuration items you are able to add architectural contexts to those. You can even relate them directly to the API sub-classes such as Managed API.

Rather than adding contextual information directly on the discoverable API CI, you can make use of the digital interface records for this purpose. That way you may govern the information exposure on a higher level, avoiding getting lost in hundreds, even thousands of CIs.

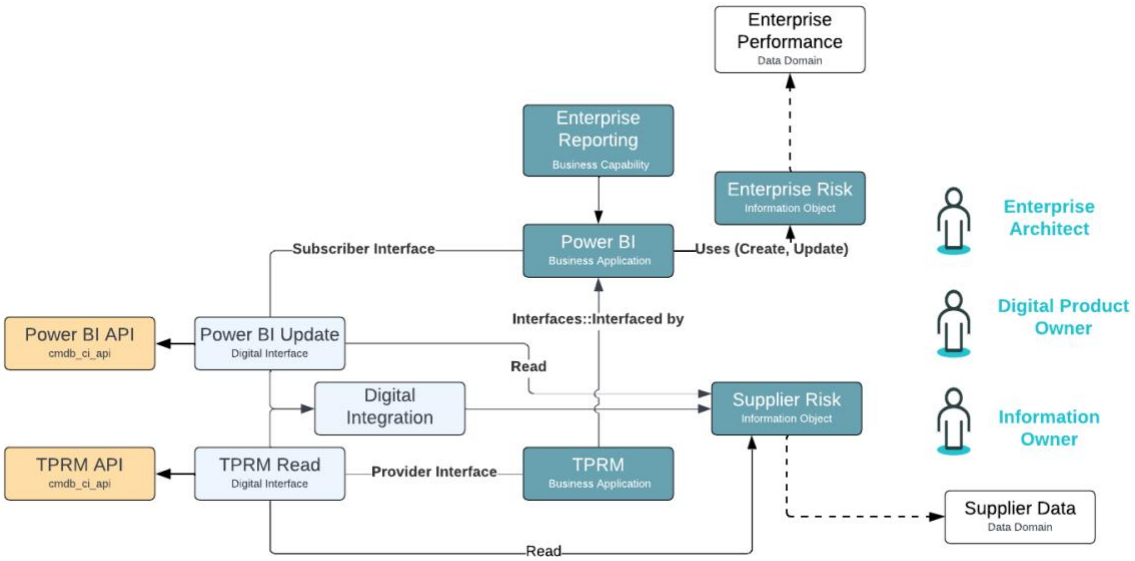

Figure 20 – Example of how a read-only interface from one business application can be leveraged by a digital interface from another to pull out data from one Information Object and update another Information Object.

Using interfaces to build integrations between applications means that you can make use of a *subset* of a data processing potential but may not make use of its *full* potential. Meaning that even if a digital interface allows you to both create, read, update and delete data, the integration we set up may be configured to only read the data. This is heightening the risk of losing control at a later stage, so a better practice will be to create more than one digital interface for the same data with different privileges. In the example above, the TPRM Read digital interface will only allow integrations to pull data, and not create, update and delete data. If at a later stage it is needed to create, update or delete data from another application, a new digital interface should be designed for this purpose.

We have now seen how the information can flow between applications on a high level of abstraction, and with what configuration items (CIs) the applications can be integrated. But information needs to be stored somewhere in the discoverable and operational side of things. To model this, we need to add artifacts from the Delivery Domain of the CSDM:

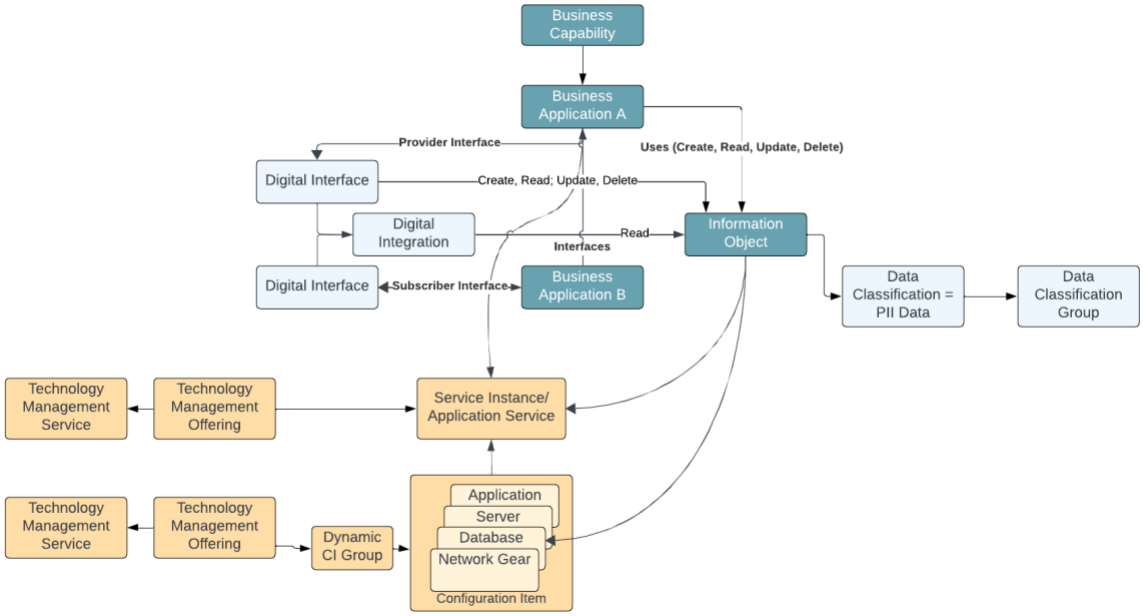

Figure 21 – The Information Object (Type) may be stored in an instance of one business application and transferred to another instance through a digital integrated through a digital integration, via a digital interface.

When you monitor traffic between two service instances of a business application you may want to show the expected relationship between those directly as a ci relationship of the type Sends data to::Receives data from. If it is synchronous you need two relationships in the cmdb_rel_ci table.

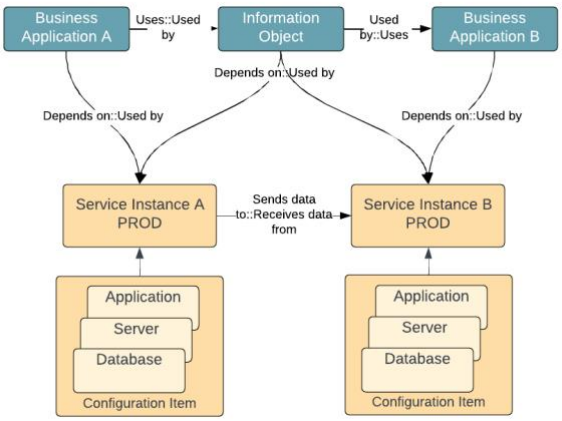

Figure 22 – Data transmitted from an instance of business application A to an instance of business application B

There are several ways of storing and accessing information, depending on the digital systems involved. You may see that information can flow through an email system and thereby be stored on an exchange server. Or you can have an application that stores all its data in a database. It can be a “zero touch” architecture where the data is viewed in an application but not stored in it, which means it can be stored in a data lake or other data cataloguing tool, etc. Why is this important? Because if the information is critical to your business, and/or confidential so it needs to be protected from unauthorized usage, you need to ensure that those “storage” objects are protected and backed up to meet your requirements.

Many organizations have enablement teams that manages databases as a service. To avoid unnecessary cost increase they should be able to separate between configuration items that contains highly critical information assets, as well as confidential information assets, so they can configure protection and recovery mechanisms where it is needed. Showing back the related Information Objects (Types) with a proper classification can allow for them to define policies that are context aware. By using IRM Policy & Compliance you can also measure (control) compliance automatically on for example Privileged Access Management settings and automated High Availability and Disaster Recovery processes and redundance in the shape of DR Configuration Items or other CIs.

Information governance in the Build and Integration domain

The Build & Integrate Domain is providing a window into the construction of applications, services, APIs, Large Language Models and other components in the shape of code, CICD pipeline executions and other artifacts. Code may be stored in code repos in the cloud, alongside AI datasets that are kept in data lakes or other storage solutions. All under constant modification. This calls for control and structure to prevent total chaos.

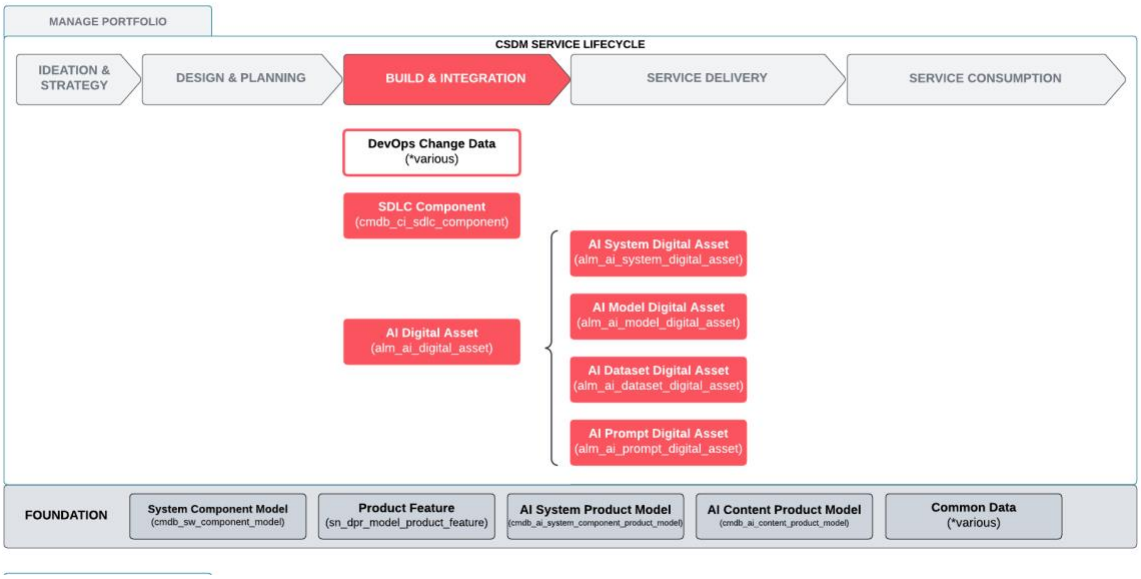

Figure 23 – The main objects in the Build & Integration Domain are the DevOps Change Models and data plus the Service Delivery Life Cycle Data, the Software Components (formerly labelled SDLC Components).

The results of these perpetual activities can be found as versions of decomposed products in the shape of asset. In the example above we see how AI Datasets are captured as parts of AI Digital Assets. These datasets should never contain PII data, as these were not intended for LLM training purposes. When an integration is being built to reuse AI Datasets from for example a data lake tool, it can be useful to reference Information Objects to see if there are any PII data in your dataset.

Building, testing and deploying Digital Interfaces and integrations

Interfaces and integrations are needed to transfer data between systems, and with a growing decomposition of systems in most organisations the control on how these are constructed, tested, deployed and maintained is increasingly difficult to keep.

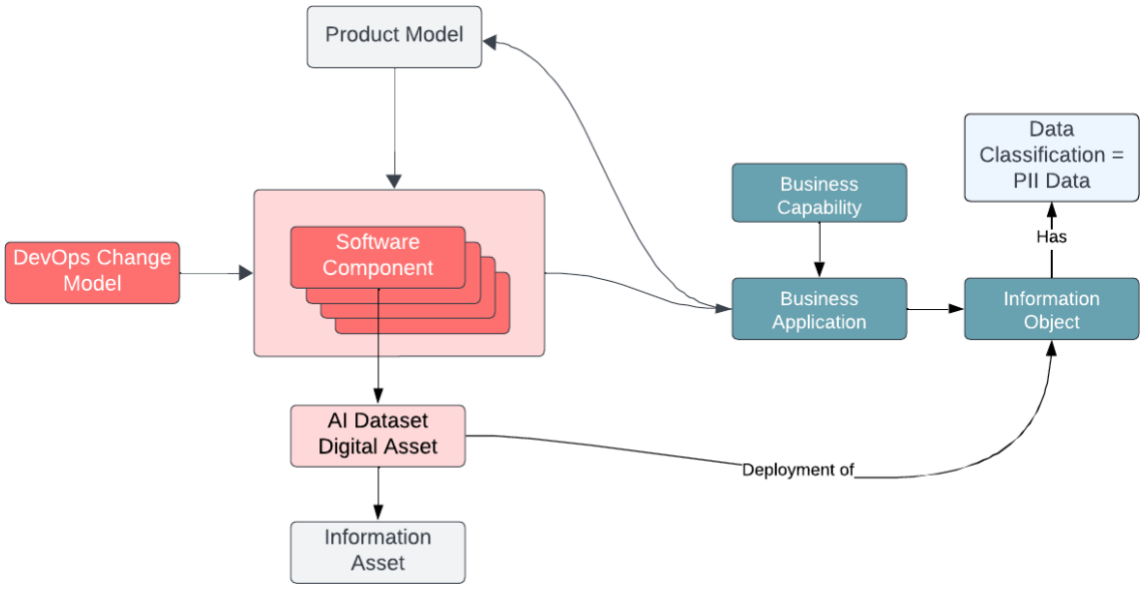

Figure 24 – When a digital interface is being designed, it will be a design artifact on the Business Application record. As soon as the build phase starts, you need a Software Component record to represent the code increment and the activities that will take place throughout build, test and release steps, including change management.

The Software Components are decompositions of digital systems, often representing building blocks of one or more product features. When used for building APIs and other digital integration types, they represent a means for you to be able to investigate the actual code, executing tests and other control mechanisms that are in place to protect and avoid malfunctioning code to be applied to a digital product.

Let's have a quick overview of what potential relationships we should apply to our Software component records. The best starting point is to relate them to the relevant Business Applications they are decompositions of. If they cannot be seen as decomposed Business Applications, they should at least reference a relevant Product Model of a suitable category. If you have not implemented Digital Integrations Management, you can relate your Software Component to the Service Instance that represents your API instance, or directly to an API CI. This is how you can link your development, test, production and other environments to the correct build artifacts so that they are not mixed. To comply with for instance GDPR, you need to show back that you don't store Personal Identifiable Information data in a non-production environment.

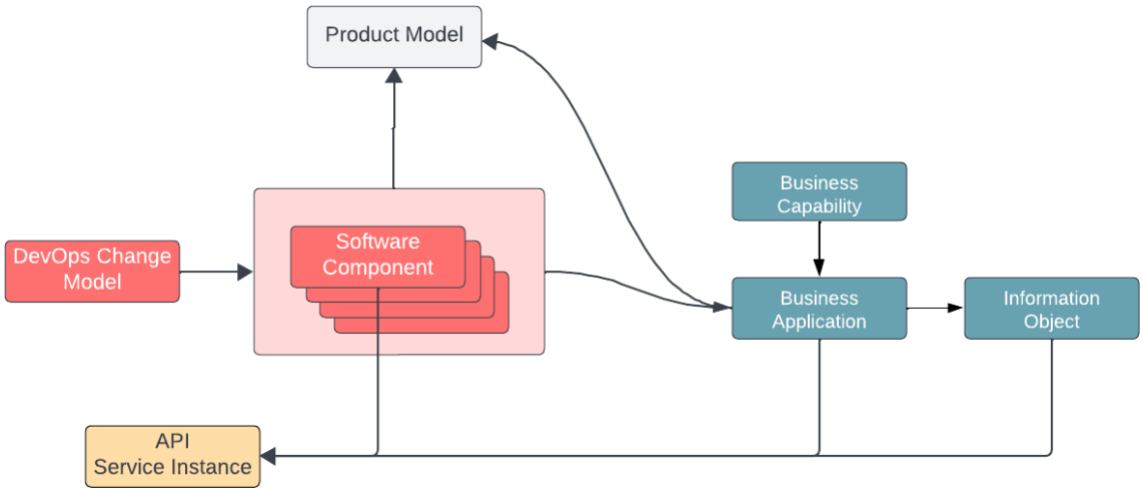

Figure 25 – The API can be represented as an integral part of the instance of the business application it provides access to data from, or as its own service instance modelled separate from the application itself

In the above example we focus only on how the code increment(s) making up the API feature is represented as a production service instance that can be leveraged in incident, problem or change tickets, as well as for events and alerts management.

If you have scrambled or synthetised data in dev and test, you may want to reflect this by adding a new Information Object (Type) representing this, that does not have the same classification as the production Information Object (Type). Whereas your Disaster Recovery instance, where your data is being backed up regularly, should point to the same Information Object (Type) as the production service instance.

Using the Digital Integration Management feature in the EA product

Building blocks of an integration of some sort should always reference the relevant Digital Integration. This is much easier to do correctly if you use the Digital Integrations Management feature. When you build a new digital interface, you can show how it is designed in terms of authorization method, use of protocols, encryption and other security mechanisms. When you have deployed the new integration, you can use tags to automatically relate API CI to the Business Application CI.

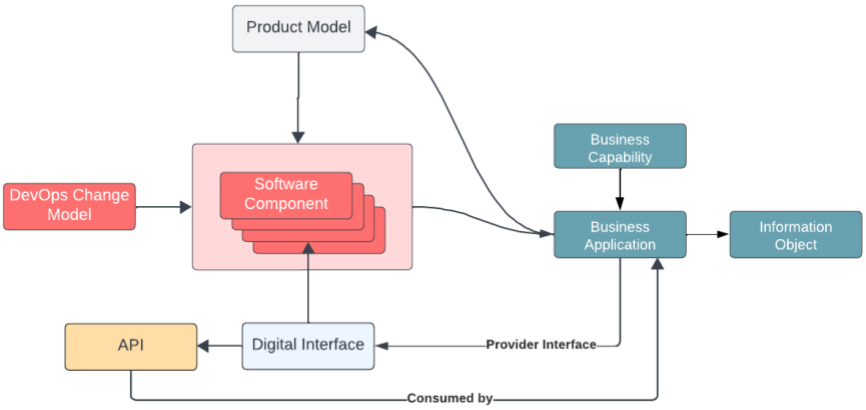

Figure 26 - In this example we tie the API (cmdb_ci_api) record to the Business Application via a Digital Interface using the API relationship table (sn_apm_di_dintf_api).

The Digital Interface is now ready to be used. By applying a Digital Integration record, you can show how the interface is being used by a subscriber Business Application, by applying what CRUD rights (Create, Read, Update, Delete) you make use of, who is responsible for the integration life cycle, etc.

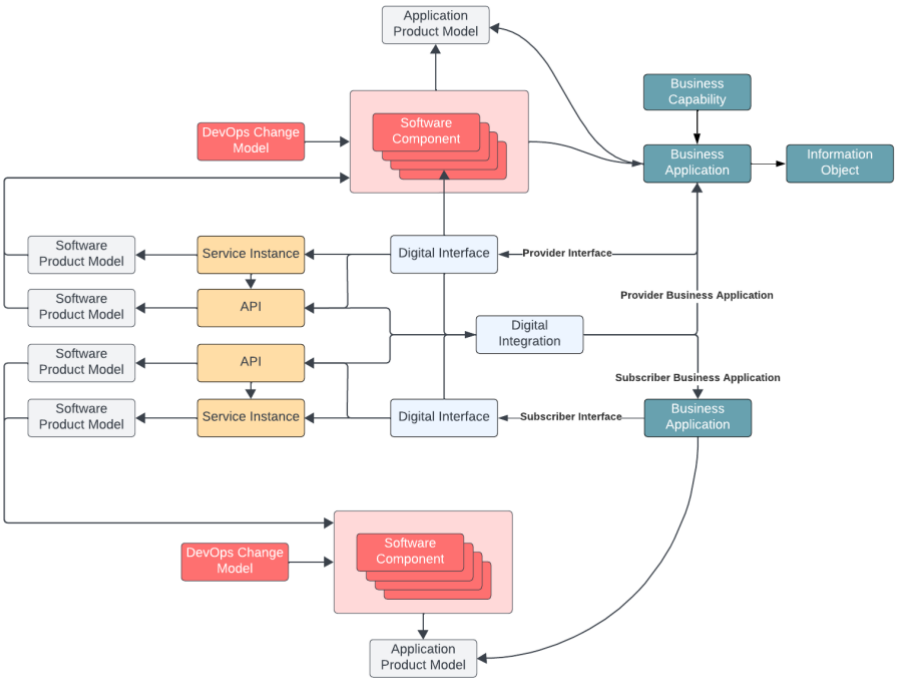

Figure 27 – Showing the composition of your digital interfaces through Software Components will enable control of the entire service delivery life cycle of the interfaces.

Tying in the observable CIs will allow for integration monitoring. This is important information for the developer team of the SOFTWARE Components that make up the digital integration, as it can be leveraged to ensure proper regression testing when the interface is being changed and updated. But also, by operations teams to figure out the criticality if they see that the integration is failing in their monitoring systems. More about this in **DIGITAL INTEGRATIONS MANAGEMENT**.

The Service Delivery Domain – Protecting and maintaining information CIs

Governing information requires life cycle management of the assets that contain, transfer and protect digital information. Knowing how these are dependant of each other requires a configuration management database structure and constant updating.

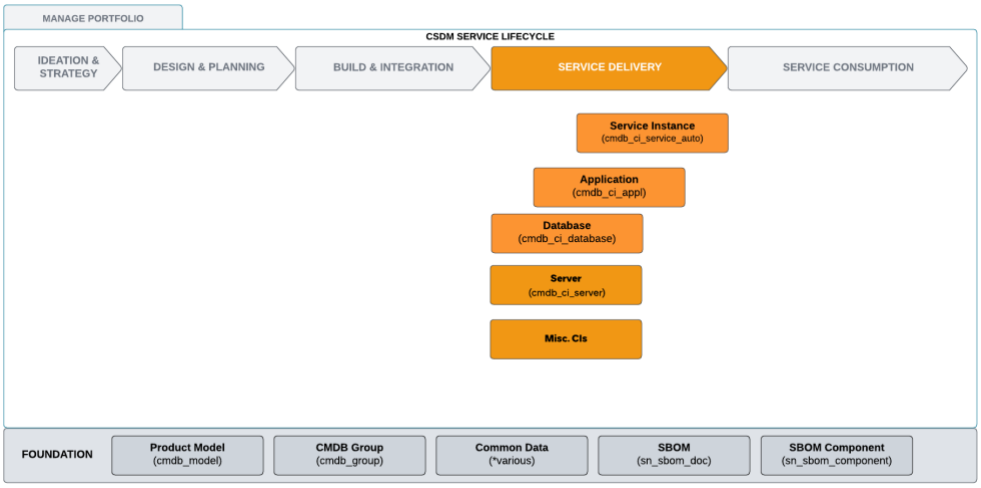

Figure 28 – Some of the main configuration item classes you can utilize to show where your Information Objects (Types) are stored.

Maintaining information storage assets

Your information may be stored as an integral part of an application, in a database, as a file on a server, etc. It is not necessarily discoverable and identifiable using normal scanning tools, so it may be necessary to create and maintain records manually or by using tags when you store information. This would usually happen as part of an agreement between the information owner and one or more technology management service owners.

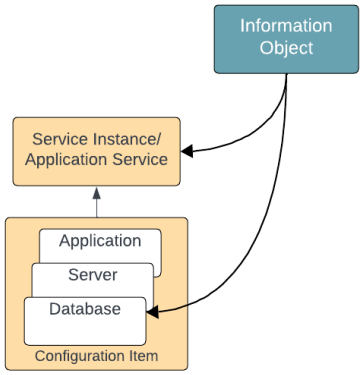

Figure 29 – Information Objects can be related directly to server-, database-, application or service instance CIs depending on how precise you want to govern the assets that store them.

Though having access to files stored in a storage medium doesn't always require

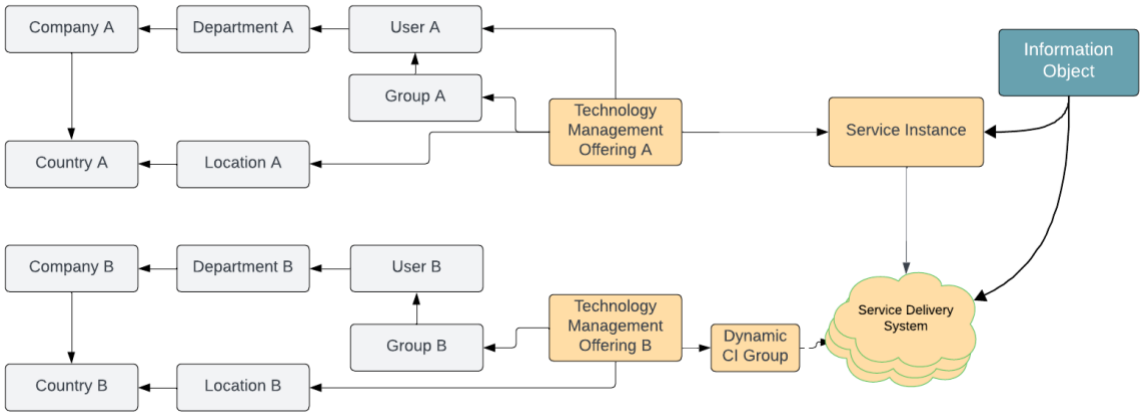

Figure 30 -The organizational and technical structures surrounding your physical data are important governance structures.

By showing where data resides physically, you are one step closer to find out who have access to the information via privileged accesses, i.e. access to servers, databases or other storage media. Data can be accesses through direct access to the database, via integrations and via the user interface on your service instance. Figuring out who should have access to your data and what risks it may cause to allow them access is crucial. The Technology Management Service Offerings describe the scope of work and responsibilities certain individual users, groups, departments and companies, in their respective locations have. Logging and reviewing if they make use of their privileges according to your policies is an important step towards control. Avoiding misuse is easier if you have Privileged Access Management processes in place. More about that in **PRIVILEGED ACCESS MANAGEMENT.**

Governing Information Object (Types)

The Information Object (Type) should have one and only one Information Owner. This is the person responsible for the collection and management of the data, and preferably also for defining who's allowed to use the data. It's not necessarily the largest consumer of the data that owns it. For example, if the most frequent user of banking customer contact information is the customer service desk, the initial collection and structuring of the data may be done by the Retail Banking Services department, and hence the Information Owner of the Banking Contact Registry will be the head of the Retail Banking Services department. The Customer Support Services department will normally be allowed to collect and update new or additional contact information on behalf of the Retail Banking Services department according to an internal agreement.

In the eyes of an Enterprise Architect or a Digital Product Owner, information is something that flows between digital systems or simply resides within one. The purpose of keeping this information in its digital form is to use it to achieve the very outcomes of doing business. This outcome is in CSDM formulated as one or more Business Capabilities. These are also useful objects to utilize in a business purpose definition to legitimize the collection, storage and use of information that can be confidential, but also to define criticality of availability and integrity. To get to the digital information content itself, you need access to the Applications they are digitized within. On a design level these are named Business Applications. They summarize the numerous installations or instantiations of applications, making it easier to keep control on a governance level.

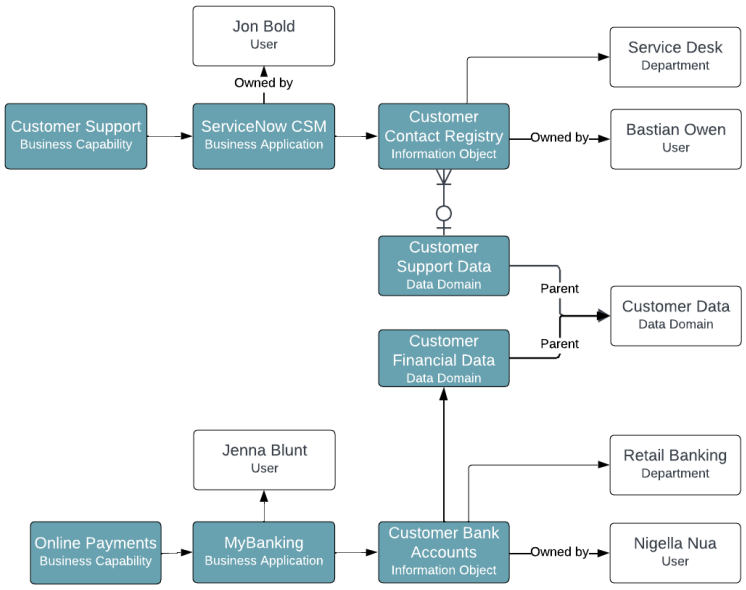

Figure 31 – showing how an Information Object (Type) is used by different business applications to provide different business capabilities but still belonging to the same data domain.

As information assets can be leveraged by multiple business processes using various IT systems, it is often hard to find someone who is willing to own this asset. Breaking them down into subparts has the last decade been an often-used strategy to mitigate this problem. It also mitigates the risk of losing control of what the subparts of such a digital registry is used for, since different processes need different sub-part. The problem remains if everything lands in the same "bucket" and is stored without any form of domain or field separation, as this will lead to either a too strict access regime or a too week one. The solution is to categorize the data and potentially protect them different sub-parts in different tiers according to their classification on Confidentiality, Integrity and Availability risk measures.

Classifying and categorizing Information Objects (Types)

1. How important is it that you protect the information, i.e. maintain its ***confidentiality***?
2. How important is it that you can provide the information at all times, i.e. maintain its ***availability****?*
3. How necessary is it to know that there has been deliberate or undeliberate tampering with the data, i.e. protect its ***integrity***?

These are questions you need to ask when deciding on what measures and controls to place on any information content carrying CI. With the Data.World acquisition the ability to discover and assess categories of information using AI will be enhanced, as well as the ability to perform a stronger metadata management for confidentiality and integrity (quality and reliability) purposes. But for now, let's look at what's already part of the ServiceNow platform.

Categorising information

There is a difference between having access to production data and data used for development and testing purposes. Access is therefore provided to one or more instances of the business applications, or to the APIs or other digital interfaces of them, in CSDM called Service Instances. You will read more about these later.
Through the usage of Privacy Management, you may add classifications of data, listing eventual special categories of PII data (GDPR and other privacy regulations). These are grouped so that it is easier to follow up on whether the data is protected at rest and in transit according to laws and regulations. There is currently no support for other types of data categories such as Intellectual Property or Competition Sensitive Data in the Classification Groups, as all of these are treated as Personal Information.

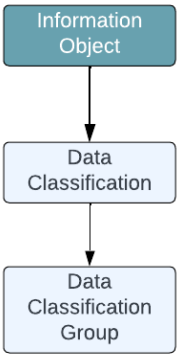

Figure 32 – How the Information Objects can be added one or more data classification tags to help you define needed measures to take on protecting them.

The data classification tags allow you to control the subset of your Information Objects (Types) that need the same protection,and will also make it easier to run scripted control on how your protective measures are performing.

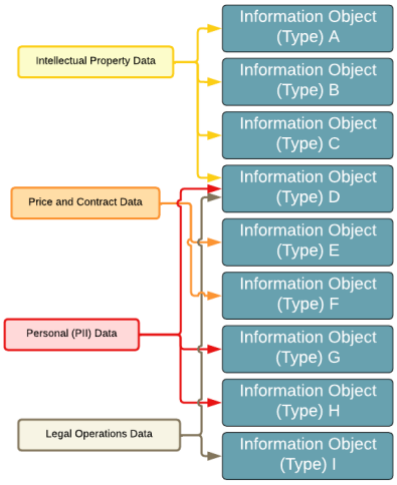

Figure 33 – Various types of information can be categorized by using classification tags so that they are easier to govern according to policies and regulations. Some may belong to multiple categories.

If you need a more holistic Privacy Management process to comply with laws and frameworks such as GDPR, NIS2 and ISO27001, you may want to make use of the Information Object Categories and functionality you find within the Privacy Management product.

Availability of information assets

You need to ensure confidentiality to avoid loss situations, such as monetary and reputational impact if it ends up exposed to unauthorized persons. But at the same time a portion of this data may be crucial to your operability. Such as the unavailability of your customer data can block you from following up on your immediate customer needs, leading to customer churn, or the processing of your services would be blocked if the information about the recipients is unavailable. Doing a value assessment on the information assets you depend on will help you to prioritize the protection of the most critical information rather than protecting all with the same resources. The latter will lead to cost increase as well as an increase in the time to resolve critical outages as you will not know where to focus.

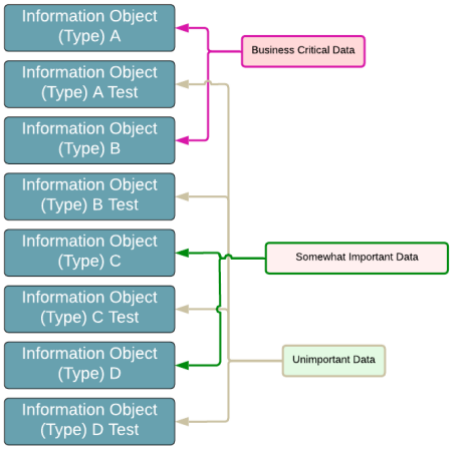

Figure 34 – Adding business criticality levels by using Information Object Categories will enable you to see the value of the information for your business across all domains, not just the confidentiality levels.

Data product versioning

Data normally comes in two or more versions/instances: one version containing production data, and one containing scrambled or synthetic data created for development and testing purposes. You do not want to mix these, as it can lead to either loss of confidentiality (people who are not authorized gets exposed to "real data", or the injection of wrongful data leading to loss of integrity. But it can also lead to a temporary loss of availability to the data you rely on in your application's production instance.

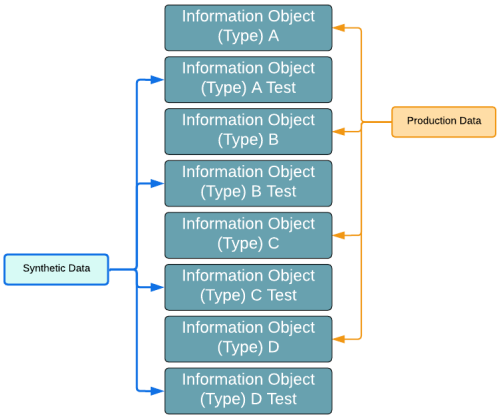

Figure 35 – Keeping production data separate from non-production data on a design level by using Information Object Categories.

In this picture the darker colours show production data, and lighter colours show non-production or modified data. These are normally not classified equally high, especially not when the data products contain Personal Identifiable Data (PII Data).

More mature organizations also scramble or modify Intellectual Property (IP) and competition sensitive data, as they want to ensure that confidential information is not exposed to unauthorized personnel or systems.

In other words: the "versions" of these information content products, or Data Products, need to have their own product models describing the differences between them to avoid misunderstandings when someone requests to buy or use them. This is described in the chapter on ERROR! REFERENCE SOURCE NOT FOUND..

Once you deploy the data, your storage media or other relevant CIs should use the Environment field to show whether it contains production data.

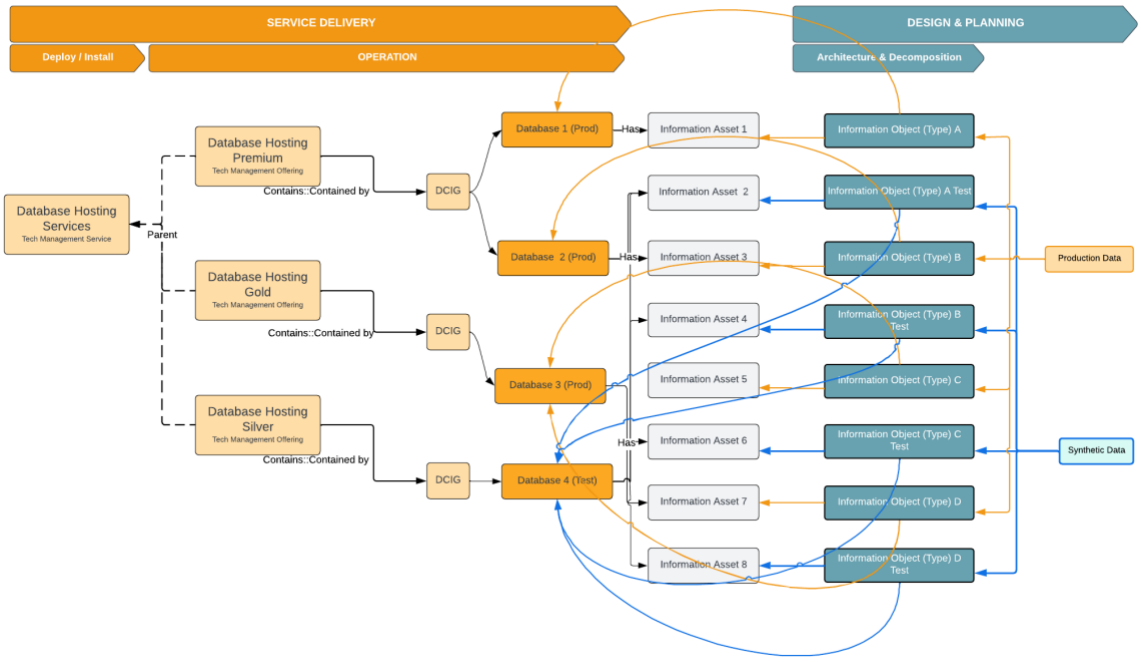

Figure 36 - Model showing how production data and test data are separated

There is normally a demand for both types of data, and setting up the access to these as a self-serviced request process is a good approach. Which means you need to "commoditize" them as two different commodities. You now have added several perspectives to your data assets, allowing multiple personas and stakeholders to pull out their meta-information of interest:

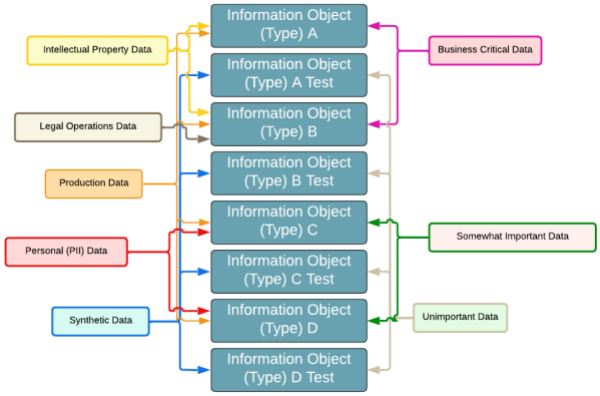

Figure 37 – Combining multiple factors that can be used for governing and maintaining Confidentiality, Integrity and Availability on all Information Objects (Types).

Prioritising your information assets according to risk level

You can leverage the same structure to do a risk-based prioritisation of what assets to handle in each way.

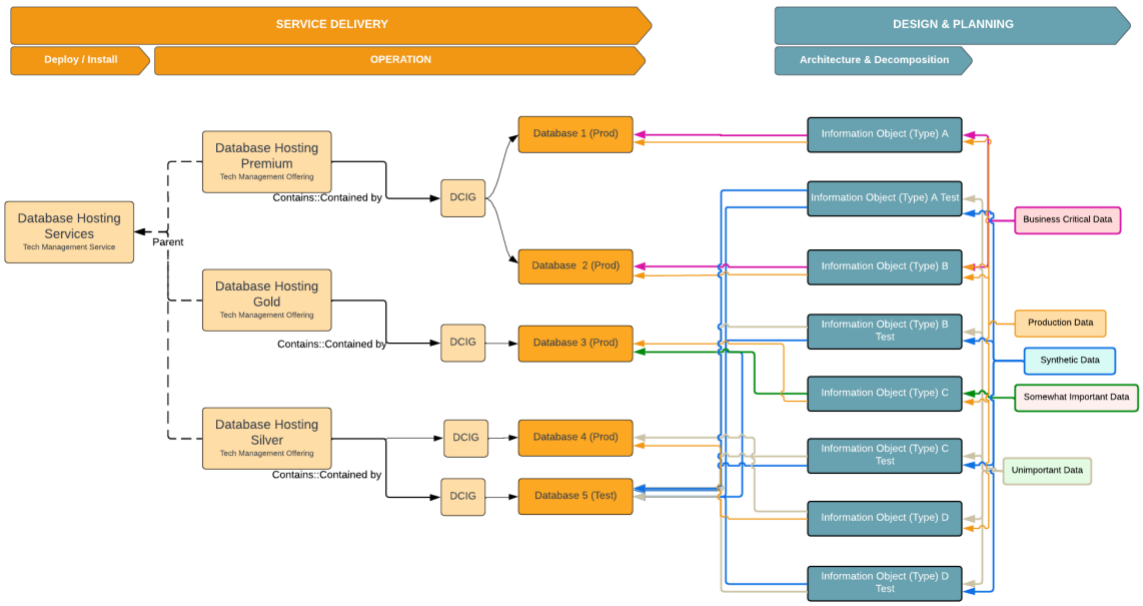

Figure 38 – Using multiple metadata types combined to prioritize what assets you should secure first.

The combination of Business-Critical Data and Production Data means that the information must be highly protected as well as frequently backed up to ensure high availability. If you use the Business Continuity Management product you can have a structured process to assess dependencies and set availability thresholds on your storage CIs based on the criticality of the Information Objects (Types) they store.

Sometimes you store confidential information that is not important for the business, such as PII data collected for an event. Best practice is to ensure full deletion of data (GDPR's Right to Erasure) when it is no longer needed. You also wouldn't need to have a full backup regime for these data. But you do need good access control, and you should also be aware of what governing laws protect those individuals before you decide on where the storage and processing of the data takes place. The question should also be: Do we really need to collect and store that data at all? Can zero-copy connectors be utilized to avoid storing the data within the processing application?

Even synthetic data can be of high importance for your business, as it may be used daily to test and verify in a Software delivery life cycle. This is especially true if you are a Software manufacturer. This means that the data needs to be protected from for instance hackers to avoid loss of availability, more than to avoid loss of confidentiality. In the model above you see dark pink colorings on the most business-critical Information Objects (Types), and green on the somewhat critical ones. The grey objects are the ones you don't need any high availability measures on, as these are of low value to the business.

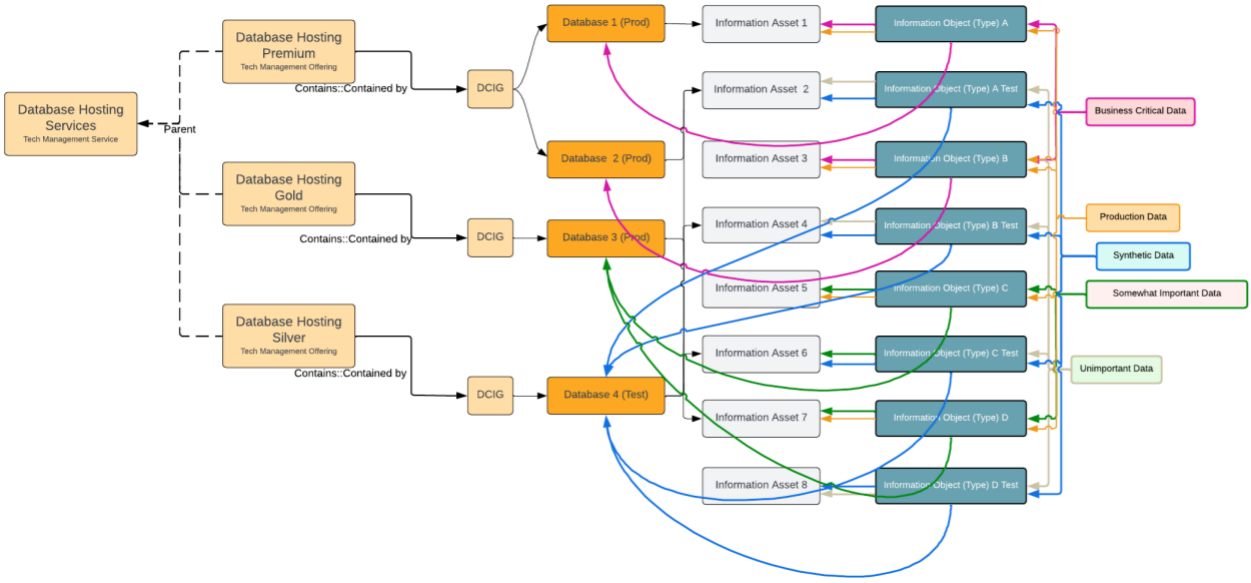

Figure 39 -Showing that Information Assets can be found deployed on Database instances.

Without the context provided via the Information Object (Type) it is difficult for the staff involved in maintaining and securing the databases to know how critical they are to the business and what policies that need to be met. Instead of maintaining this information directly on each individual Information Asset and Database CI, it's easier to govern this on an abstract and less decomposed level. In the figure above we see that the legal policy stating that production data and test data should not be mixed, may lead to an internal policy that the databases containing production data should be kept separate.

Since doing frequent backups cost more there may also be a good idea to store business critical data on databases under a premium hosting offering, while less important data are maintained under an offering with a less strict OLA.

Some data may require domain separation or even air-gapped storage. This is also possible to apply as tags on an Information Object (Type) record level, though sometimes it's a customer requirement and not necessarily a generic policy. In the figure above, Business Critical data (Information Object (Type) A and B) are kept in separate databases as part of a policy that Business Critical Data should reside in individual storage media.

Processing and distributing information

To model where and how information is collected and processed, it is useful to make use of a Value streams in combination with business processes and business applications. The value streams overarch the business processes to provide a more holistic understanding, allowing potentially many business processes to contribute. Without the value stream context, it may be hard to show back just how information is collected and updated chronologically. But you would also need to account for *why* the information is collected and/or updated in a business process, and for this you may use the business capabilities.

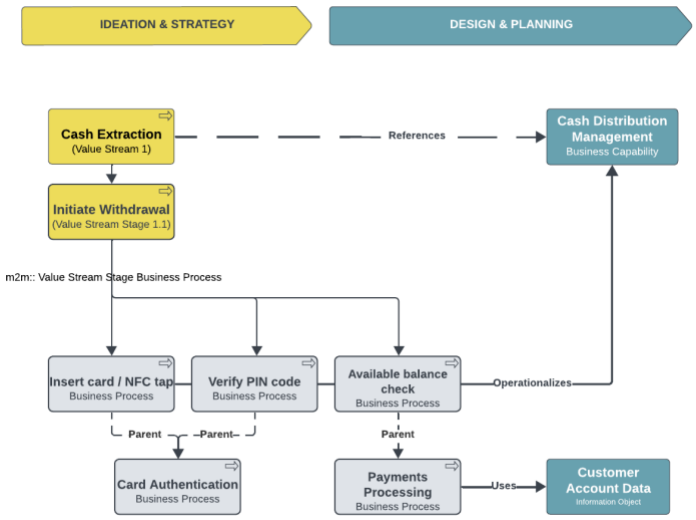

Figure 40 – Value streams allow you to see across multiple business processes and how information is utilised in those.

If you want to model that one business process is collecting data and another is updating or even removing data, you can achieve this without losing the overall picture.

Knowledge graphs

One thing is to retrieve and offer bulks of information content; another is to understand and to use this information correctly. This is where Knowledge Graphs come in handy. A knowledge graph is a way to provide necessary context to an information element so that it can be used for one or more purposes without misunderstanding or misinterpreting it. This is how Claude.ai defines it:

"*A knowledge graph is a structured representation of knowledge where information is stored as entities (nodes) and relationships (edges) between them, forming a graph that can be traversed, queried, and reasoned over.*
The core idea is simple: rather than storing facts in isolated rows and columns (as a relational database does), a knowledge graph stores facts as triples — subject → predicate → object. For example: "Customer Service *depends on* CRM Application" *or* "John Smith *is employed by* Finance Department." *Each of those statements is an edge in the graph. This is what your CMDB already aspires to be — but extended beyond configuration items to include everything that gives those CIs* meaning*: the business processes they support, the data they carry, the people who depend on them, the policies that govern them, and the decisions made on them. Where the CMDB stores* records*, a knowledge graph stores relationships and context*."

With the use of such knowledge graphs, we can ensure reliability across systems and processes making it easier to build workflows across such boundaries.

Subscriptions to information via APIs

With the on growth of stricter privacy regulations as well as abundance of accessible data causing cost growth and latency, we see that the increments of data being offered digitally becomes smaller and are more precisely adjusted to business needs.

Event based integration architecture allow you to receive the deltas between the information you have already received and what is added, updated or deleted since last time, as well as allowing you to subscribe to the exact datasets you need. Many companies are now leveraging data provisioning tools for subscription management, self-serviced developer portals for provisioning APIs and API gateways to govern the APIs.

This is great but makes it even harder to document and track for compliance across these, unless you are fine with manual reporting and ad-hoc audits. Plus: the overall context for the data you want access to, who the owners are, what you need the data for, how you handle the data you get access to, and how you ensure that no policies are violated across these perspectives, is often lost along the way.

To avoid loss of control, you may want to add this simple structure around the consumption side of Information Management:

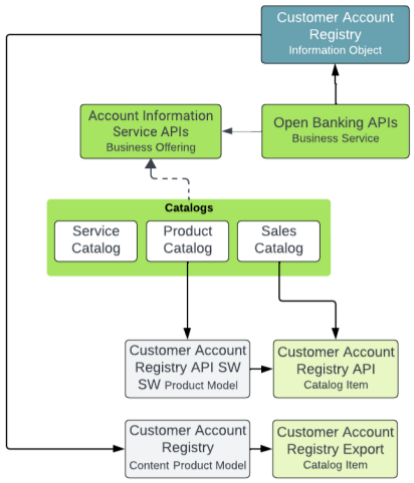

Figure 41 – Examples showing what self-serviced "Information as a Service" can look like.

For each API there is a product model equivalent. To allow self-service, you can choose to create catalogue items directly from these. By adding them to the external sales catalogue, you can open for outside consumers as well. When someone is approved to use them, they will be added to the list of Sold Products, even if they're free of charge. This will make it easier to report on service performance directly to the consuming party.

Some may claim that it goes without saying that if you request the use of an API, you also request to use the data that is supposed to flow through it. Though that is usually true, there is most likely different owners of the data content, i.e.. the Information Object (Type), and the digital interface you want to make use of. So, whereas the owner of the digital interface may approve your request from a technical and service delivery timing standpoint, the information owner may approve based on whether you fulfil the requirements to receive, use and maintain the information according to certain confidentiality perspectives.

The solution would be to create two different products and hence catalogue items for this, but to sell/deliver them as a bundle. Then you maintain the responsibilities and approval workflows with less scripting needs.

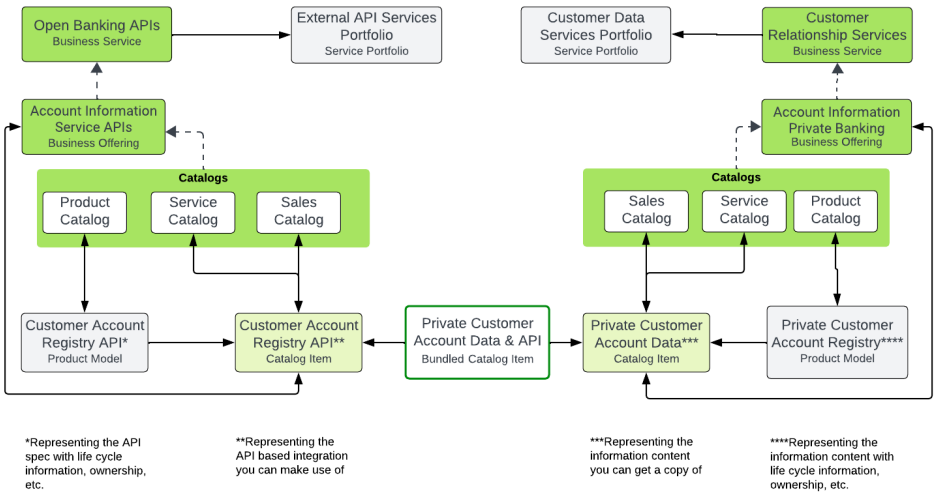

Figure 42 – Creating product model equivalents for your business offerings to generate service-as-a-product type catalogue items that can be bundled but still made autonomous regarding life cycle management and ownership, etc.

Referencing external information sources

If you receive information via external sources and want to model and possibly monitor these for operational resilience or business continuity planning and reporting purposes, it is a good idea to create configuration items for those as well.

Since you don't see the detailed composition of those source systems, you only need a service instance record, potentially also an API CI record to map to the endpoints you want to monitor.

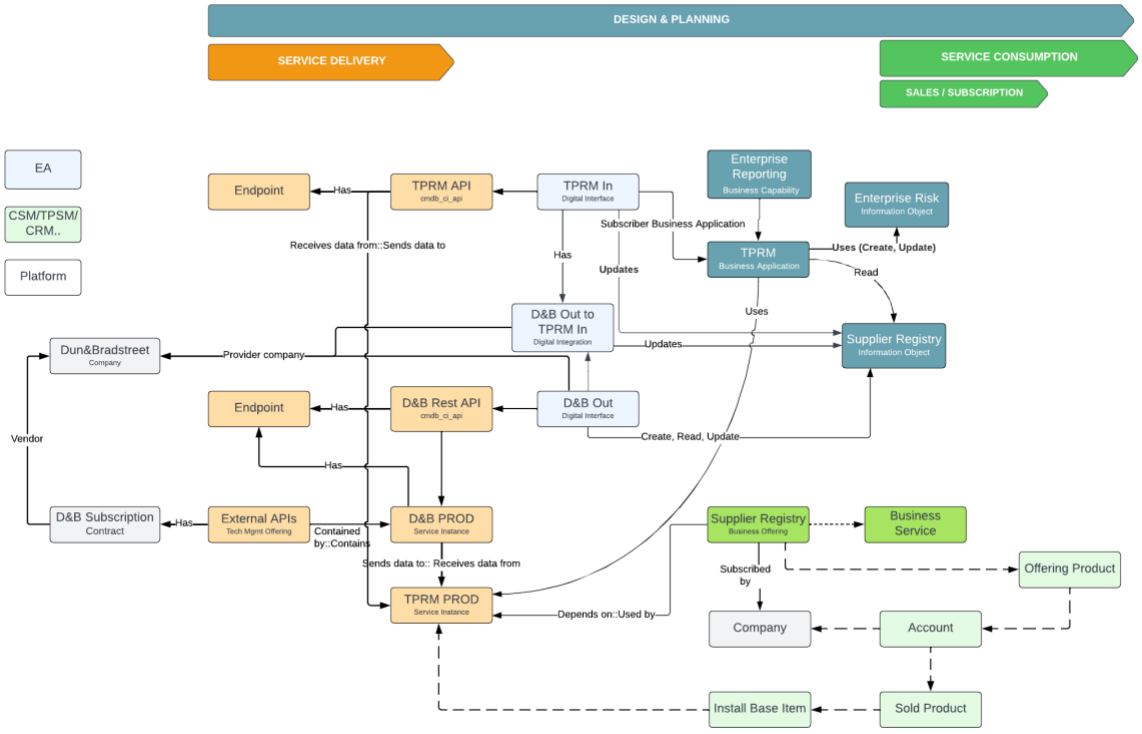

Figure 43 – Creating business application records to represent external source systems is not mandatory for governing their APIs in a good way.

As soon as you have Service Instance records (Dynamic CI Groups are also sub-classes of those) you can relate them to technology management offerings that show who's in charge of the services that provide, deliver and maintain these. If your company is providing information services to another company (subsidiary, branch, etc., you may want to show those as subscribers. In the example above, we see how it is possible to model data flowing from an external source system, Dun&Bradstreet production API (represented as a service instance), into the TPRM production instance. To add metadata about the integration that has been built, both digital interface records and one integration record is added. Since we don't know much about the system behind the API from D&B, and for all we know this may be changed at some point without us knowing about it, it is not necessary to represent it in the Business Application table. But since we want to monitor how it performs, we add endpoint information. We can show what company (third party/supplier) we rely on for this integration, even contracts stating our rights of usage.

Since we may resell the data we collect and process in our internal Supplier Registry to subsidiaries (internal business offering subscribers) or to external customers using the CSM/TPSM Accounts Sold products and Install Base Items to show how this Offering Product is consumed.

Data lineages modelled using CSDM

The data lineage describes the flow of data from its very origin (source system), via potentially multiple systems and processing activities, to its current state. It's needed for ensuring compliance to internal policies as well as external regulations if applicable. For example, to comply with GDPR a company has to report on how personal data is protected from unauthorized access, both for keeping its confidentiality, but also to ensure correctness. Showing *what* has happened to the data *when* and by *what or whom* is part of this.

Data Lineage is the foundation for several critical capabilities:

- Impact analysis — before changing a source system, you can see everything downstream that will be affected. For example, if you change an application to a new one, you need to consider all integrations the old one has to other applications as those may need that data.
- Root cause analysis — when a report shows a wrong number, you can trace back through every transformation to find where the error was introduced. Logging all transactions and activities may be needed to achieve this.
- Regulatory compliance — regulations like GDPR require you to know exactly where personal data came from, where it went, and whether it was processed lawfully at each step. Without lineage, that audit is a manual investigation; with it, it's a query.
- AI trustworthiness — when an AI model produces an output, lineage tells you what training data and source data it was grounded in, which is increasingly required by regulators assessing AI systems, such as the EU AI Act.

Again, this will be supported even better with the Data.World data catalogs and knowledge graph features. But even without it there is a lot you may do to map out some of these data lineages. This is useful when/if you later choose to add data from the data catalog feature.

Figure 44 – Even without the Digital Integration Management feature in the Enterprise Architecture product it is possible to map out where data resides (at rest, red line) and where it is transmitted (API to API), as well as the business processes that use the information as part of a value stream.

Governing AI related information

Artificial Intelligence, and especially the Large Language Models you can use to build AI Agents, introduce a whole new set of concerns and risks. Putting up the necessary level of governance around these, as well as proper life cycle management and cost control is just as important as it is to be able to measure and improve their efficiency towards business goals and targets. The same goes here as everywhere else: you cannot govern what you cannot see. There are some useful decompositions of these AI systems:

- The Large Language Model as SOFTWARE Components, comprising of code and other resources
- The Service Instances they execute from
- The Data Sets they are trained on
- The information they have access to and their system access privileges
- The Skills and Prompts they have
- The versions and life cycle of all the above

- The log files showing what they have done

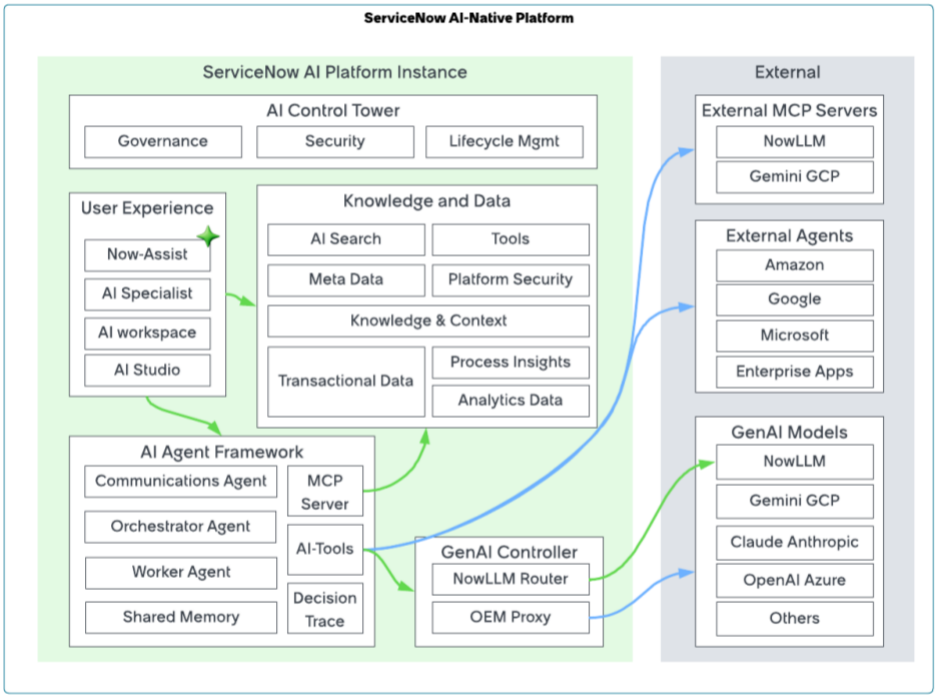

Figure 45 – The high-level structure of the AI Control Tower capabilities

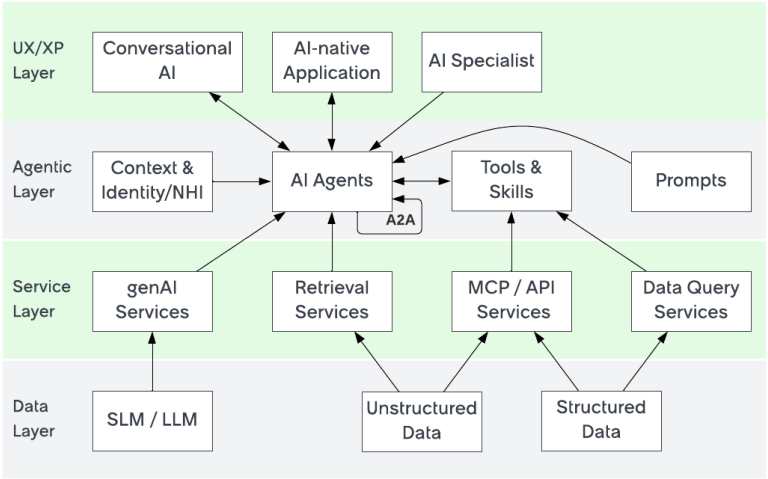

Figure 46 – The AI Data Layer consists of the SLMs and LLMs themselves as well as unstructured and structured data

Specific AI and LLM related assets

The introduction of AI Governance has some interesting new tables you can leverage to gain control of AI systems, AI Datasets and LLMs as commodities or assets. Over the years there will be many new versions of these artifacts, supporting new business needs at a high pace. Errors and regulatory compliance issues can be discovered at one time, and the judicial processes and audits may happen later, and so you will need to show back snapshots for each main version of your AI model and its related AI content. If an incident occur you would want to relate it to not only the instance of the LLM, but also the exact version of it at the moment the incident was flagged. The same will be the case for reported bias or inconsistencies in the datasets the LLMs

are trained on. Since these datasets will potentially change at a rapid speed, fetching the versions of these that were deployed when the error was discovered, will be critical for fast resolution and reporting when the incident is solved.

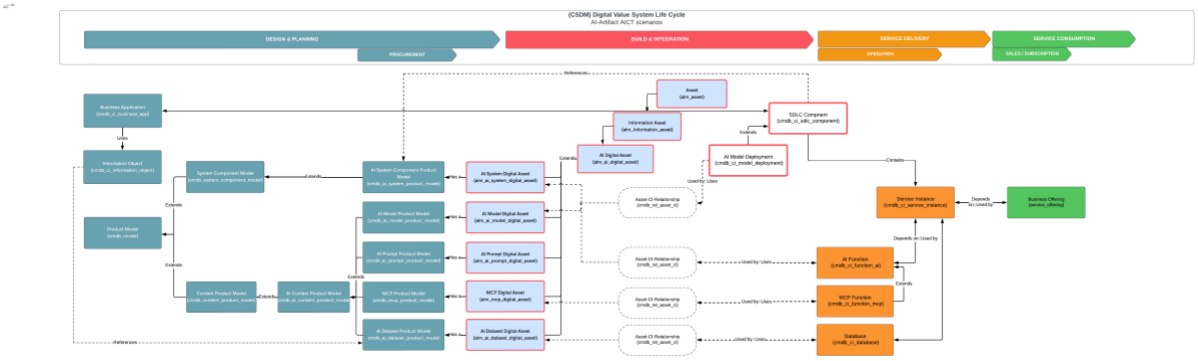

Figure 47 – Large Language Models and other AI Models can be decomposed and contextualized into the configuration item, product and asset dimensions of them.

In building your operational resilience you need to show back how you safeguard your most valuable assets. In near future the AI Agents will replace people in many processes and services, making them critical for operational stability and business continuity. Different versions of LLMs may be applied differently across geographical locations, making it even harder to monitor and govern them.

The ServiceNow data model has been extended to handle the AI system information. The AI Dataset information asset information needs the AI Dataset Digital Asset (alm_ai_dataset_digital_asset) table to describe the asset dimension of it. The product specifications of the same are stored in the AI Dataset Product Model (cmdb_ai_dataset_product_model) table. Whereas the configuration aspect uses the AI Function (cmdb_ci_function_ai) and AI Model Deployment (cmdb_ci_ai_model_deployment) tables to describe how the deployment of the AI model is related to the other parts of the IT landscape.

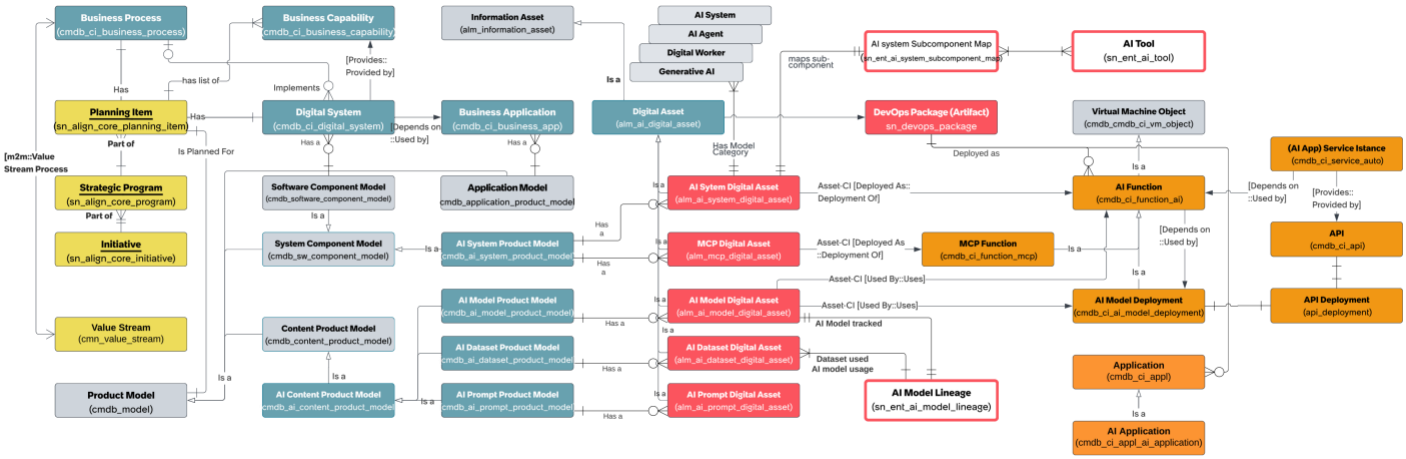

Figure 48 – The AI specific assets and products tie into the larger CSDM context to provide information and metadata on how the AI systems are constructed with its individual components and their specifications and life cycles. In the above figure, a database is being related to an AI dataset, but the storage mediums may leverage other types of CIs in your architecture.

Decomposing the LLMs and other AI systems and scrutinizing their individual assets and product definitions can provide insight that can be leveraged in incident resolution, vulnerability response and business continuity, to mention some processes. If

there is a weakness in a certain version of one component, it can trigger an action to mitigate the risks this causes. Using AI Control Tower application, you may also immediately shut down an agent that causes potential or actual harm.

Since the AI systems process information, being trained on datasets, consuming datasets when performing their tasks, and potentially altering information in the source systems they have access to, the information about these activities also needs logging. Ensuring that your architecture allows you to find these AI datasets, relate them to your AI systems and use them in reporting, incident handling and continuity planning, is therefore crucial.

Common questions about AI and data residency

Where is data stored when I use an AI Agent? The answer depends on the architecture surrounding your LLM deployment, i.e. the AI Model Deployment CI. It uses and stores data as it is designed to. On the ServiceNow platform the AI Datasets are stored on the ServiceNow service instance it is deployed in, unless you design it otherwise. This means that data residency remains in the same data warehouse, secured the same way as your other information.

What if I do AI Agent orchestration using AI Control Tower? Well, the individual AI models will store data according to the architecture surrounding them, though metadata on flow performance and results will potentially be stored in your ServiceNow instance unless you specifically design it not to.

This means that if you use a ServiceNow AI orchestrator to kick off AI agents in for example Microsoft Copilot, the metadata and input prompt for the initial tasks will be stored in ServiceNow, whereas the input for the individual Copilot agent will be stored on your Microsoft platform.

Protecting data at rest

Data should be handled according to policies that are contextual:
Strictly confidential information should be accessible only to named individuals on a need-to-know basis. The mechanism needed to ensure that it is impossible for others to access this information, must be enforced. Year by year the threat actors become better at intruding, and the mechanisms to prevent intrusion gets increasingly refined. That's why the detailed policies that are applied needs to be updated on a regular basis as well, and the controls that check for compliance should follow the same frequency.

The same is true for data integrity. The more critical it is to ensure trustworthy, updated information unmanipulated by threat actors, the stricter the controls on who and what system is allowed to insert, update and delete information. Also, when there is a major outage mechanisms must be in place to ensure that data is not overwritten by less up to date data from other sources in the value chain.

When availability to the information is of high importance, the frequency of back-ups must be higher, as well as the mechanisms for protection against ransomware and other data loss prevention mechanisms.

How can you ensure that data is kept at rest according to these policies? In most companies there are centralized maintenance and enablement teams handling the databases, data warehouses and other resources. Well, it starts with having full control over the confidentiality and criticality of the information, and that the confidentiality and criticality metadata are known to those in charge of maintaining and storing the data. If they are unaware of what they are dealing with, they may either spend time and money on not so important information or not place enough guardrails and service recovery measures on their assets.

Then comes the national, regional and multinational laws and regulations. Laws such as the EU Cloud and AI Development Act and local security laws are there to enforce guardrails on information assets that, if ending in the wrong hands, can pose a threat to national, regional or even international social stability. Or if made unavailable to those who need it to uphold their national or regional functions, can destabilize national security and economy.

Open-Source data storage and provisioning solutions

To maintain full sovereignty of your information assets you may need to build up a zero-copy architecture, where you utilize best of breed proprietary applications or application platforms to engage with the information, without storing data in those. The data can be stored in Open-Source data warehouses and data cataloguing systems and processed by Open-Source solutions such as Apache Spark or Kafka. You can still leverage the metadata (data about the data) needed for performing tasks, logging

activity and run compliance control measures. If you want to store data on the platform, you can use encryption keys or other mechanisms to protect the confidentiality of the information you need on your instances.

For ServiceNow purposes there is a Zero Copy connector available for the most common of those solutions.

Back-ups and archives

Regulations may sometimes demand that you store information for historical purposes. Data retention rules define the time frames for this and will often overrule internal deletion policies. An example is the Right to Erasure which is an important part of GDPR, and the laws demanding full archive of all monetary transactions in national finance laws or international Anti-Money-Laundering directives. Retention rules can be configured natively on the platform.

Disaster Recovery and High Availability

To ensure full availability to your critical information assets in a potentially long-lasting, full impact outage situation, you may have set up a fully redundant solution. Leading the traffic over to these resources is something you should test at a regular basis, to ensure that it will in fact work when the disaster hits you. This is where the usage of live, updated information about those resources, is necessary. Excel spreadsheets tend to be error prone and outdated. Plus, securing that the configuration on the redundant assets match that of your normal production instances needs to be monitored and included in your patch and upgrade regime. Setting up a timely backup regime and automating this according to business needs requires a business impact assessment to avoid unnecessary high cost on low criticality information or too infrequent backups on high criticality Information Objects. This is easily done using the Business Continuity Management product on the ServiceNow platform.

Audits and governmental insight into digital information

The US Cloud Act applies to all cloud providers located in the US and all cloud providers who are fully or partially owned by a US company or whose parent or ultimate parent has main office in the USA. The law states that US courts could compel companies under US jurisdiction to disclose data, potentially overriding EU privacy protections in practice. Auditors and regulators may also have a legal right to gain insight into your stored information regardless of if it's in the cloud or not. US regulators can also demand that the cloud provider blocks all services for named persons, leaving the data out of reach for them. This raises the issue of Data Sovereignty, as described in the introductory chapter.

To show back what regulations impact your data and workloads, you need to show back where and by whom the information is 1) collected, 2) stored and 3) processed.

1a) What is the main location, and thus country, of the company who runs the service that collects the information, 1b) what is the location of the people who collects, reads and updates the information and 1c) what is geographical location where the information is being collected, usually the systems and/or individual persons that provide the information, also when this is provided from system/people outside the company where the collection of data is stored and further processed. If any data processing is taking place outside the geographical boundaries of the responsible company, this is important to capture and safeguard according to the laws regulating the information processing activity.

For information residing in tables on the ServiceNow Platform, the Data Privacy tool allows you to apply Data Classes (f.ex. Open/Public, Internal, Restricted, Strictly Confidential) to the various tables those are stored within. This allows you to govern many tables using the same mechanisms based on Data Class rather than handling the tables individually.

Protecting data in transit

The whole point of having information is to be able to use it to achieve a wanted outcome. This normally takes place using an application. But oftentimes the application you use to produce the outcome is not the origination of the information. Therefore, we need integrations between systems where data can flow. This is what we call "data in transit". Data is moving between the systems. The reason why we now call it "systems" is that this does not always happen using what we normally call applications. It can simply be to store data as a file on a memory stick, reusable for a multitude of applications that can read the selected format on the file.

It is important that we can trust the data after it has been transferred, for example that no one has tampered with it or that the file has been corrupted by some technical error. It should also be understood at the consuming side, meaning that the format must be interpreted correctly.

We don't want anyone without a legitimate purpose to access the data in transit, since this may breach the confidentiality of it. We therefore may need to protect it using various forms of security measures to deny unauthorized access to it and perhaps encrypt it so that it cannot be understood. For data in transit to and from the ServiceNow platform, there are many security methods, such as multi-factor authentication, single or double encryption, and various protocols to select from. Again, selects a propriate method based on the criticality and confidentiality level of the information.

Lastly, we want to ensure that the data does indeed run through, making it available to those who need it at the appropriate time. We therefore often monitor the transmittal using log monitoring tools.

Monitoring transactions and API calls

There are several monitoring tools that can track how your APIs and other integrations are performing. Fetching data from these for usage in the Events & Alerts management process, and even into the Incident process, you can manage and control the availability of important information across your system landscape. Using the criticality scores from your Business Impact Analysis (if you are using BCM) or by setting those values directly on your digital integration record may provide you with insight into the criticality from an architecture perspective. If the API is delivered as a service, you will most likely define the business criticality on the service offering record and place service commitments there to start measuring planned versus actual service availability.

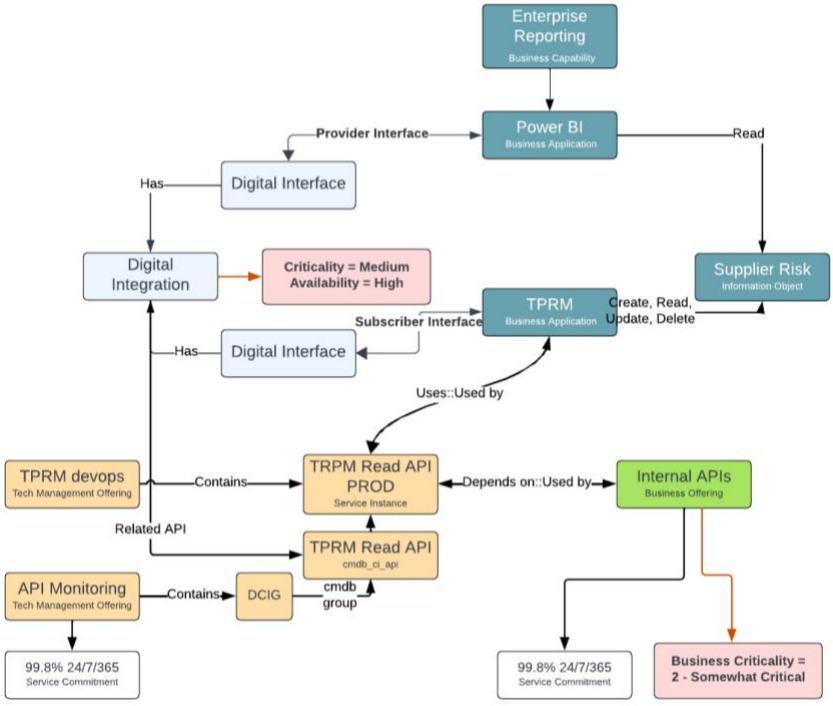

Figure 49 – Example showing how you may relate your digital integration records to API configuration items and on what records you can define availability targets and criticality levels.

Access Management

Increasing use of data lakes, data cataloguing tools and decomposed architectures with microservices will allow you to provision information and apply access controls and privilege management on a very granular level. The risk is that it will be increasingly difficult to keep a high-level view on the overall compliance to corporate access policies, as well as to provide a comprehensie report on how you have secured your information assets. The result is often week-long audits with excel spread sheets at best. At worst it opens you vulnerable to critical security breaches that lead to data leakages or loss of availability or integrity of information from internal or external threat actors.

End User Access Management on the ServiceNow platform

In ServiceNow the main mechanism for defining and governing the access to information systems is by using access roles. You find them in a table called user_roles. There are numerous ways of managing access to applications within and beyond the ServiceNow platform, but if you want to achieve a holistic architecture of this, you may want to use some of the following tables to set it up:

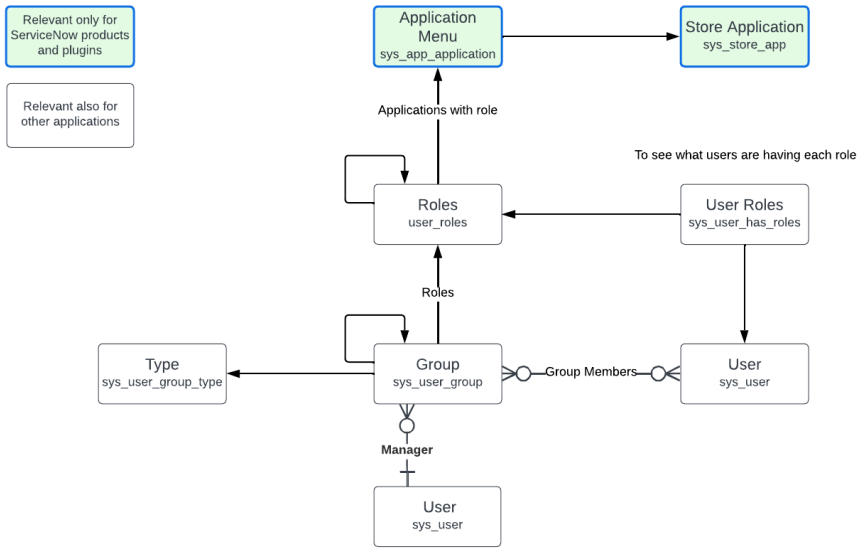

Figure 50 – Roles can be applied to users (human or non-human) to show what privileges has been provided them.

Privileged Access Management

A *Privileged Access* is when we allow someone to go straight into the application layer, database or storage layer of your application instance, or into the infrastructure layer to handle code and/or data. Usually this is something only trusted persons and systems can do, according to the level of criticality and confidentially of those information resources. A good practice is to allow this per session only, and place one or more approval steps in the access request process for control and logging purposes afterwards.

In ServiceNow it is possible to create this request and approval process as a workflow, preferably integrating with a PAM tool or using the ServiceNow IAM product. A good practice would be to relate the Configuration Item you request access to the request item, so that it is easier to generate reports and potentially policy controls and audit logs based on this.

Encryption and cryptography

Applying cryptography is a good method to ensure confidentiality (and potentially also integrity) of your data in situations where other access management methods are not sufficient. This is a method that allows customers of SaaS products and services to restrict access to the stored and/or transmitted information to actors who hold the encryption keys. Double encryption has become increasingly popular both for data in transit and at rest. This calls for governance of cryptography, where key rotation, CBOM (Crypto Bill of Material) and cyber resilience threat monitoring are vital parts. Unifying your data about how your total security design and the testing and continuous improvement of this regularly will make it easier to report on your resilience not only to cyber-attacks but also other threats to your data sovereignty. You can read more about this in the chapter

COMPLYING WITH CORPORATE AND LEGAL INFORMATION SECURITY POLICIES AND requirements

Complying with corporate and legal information security policies and requirements

Information needs adequate and risk-based protection. Some of the risk drivers may be strictly internal needs, such as criticality for the company to be efficient and productive. Or avoiding information to fall into the hands of competitors who thereby can outcompete you. But normally a company must comply with laws and regulations enforced by national, regional or cross-national regulators.

Establishing Information security policies and relating them to Authority Documents and Risk Frameworks

The EU regulation on digital operational resilience for the financial sector (DORA) defines in Article 3 (6): 'information asset' means a collection of information, either tangible or intangible, that is worth protecting'

More specifically defines the EU General Data Protection Regulation (GDPR) in Article 4 (1): 'personal data' means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person'

Control 5.9 in the revised ISO 27002:2022 describes how an inventory of information and other associated assets, including owners, should be developed and maintained.

To carry out its activities, the organization needs to know what information assets it has at its disposal. An inventory of information assets (IA) is a list of everything an organization stores, processes, or transmits. It also includes the location and security controls for each item. The goal is to identify every single piece of data. You can think of it as the financial accounting equivalent for data protection.

An IA can be used to identify gaps in your security program and inform cyber risk assessments where you may have vulnerabilities that could lead to a breach. It can also be used as evidence during compliance audits that you've done due diligence in identifying your sensitive data, which helps you avoid fines and penalties. The inventory of information assets should also include details of who owns each asset and who manages it. It should also include information about the value of each item in the inventory and how critical it is to the success of the organization's business operations.

According to control 5.9, the inventory of information and other associated assets should be accurate, up to date, consistent and aligned with other inventories. Options for ensuring accuracy of an inventory of information and other associated assets include:

1. conducting regular reviews of identified information and other associated assets against the asset inventory.
2. automatically enforcing an inventory update in the process of installing, changing or removing an asset.

Data sovereignty in a changing geopolitical situation and in cross-border operations

We have seen changes of regimes and war intrusions in countries that have for a long time been peaceful and reliable to do business with and store data within. Having an exit plan for how to act if you need to stop doing business with or move your data from a given country is crucial for your resilience. Having a disaster recovery plan where you also practise on re-locating your IT operations and data storage to another country allows you to act fast and can even lower your cost in the long run as you will not need to set up an extensive task force if a sudden breakout of war or destabilisation should occur.

In the unstable political climate, we are facing globally cryptography can be the solution for ensuring data sovereignty. Although the government of a country is seen as "friendly" today, they may not be so tomorrow and may want to access your data for purposes you cannot accept.

This is also relevant where you are faced with mutually competing regulations and country laws. An example is the GDPR regulation within EU/EEC, which may be challenged by the US Cloud Act (Clarifying Lawful Overseas Use of Data Act, 2018). Whereas GDPR demands full data sovereignty and access of Personal Identifiable information restricted to EU/EEC countries, The US CLOUD Act requires American companies to provide US law enforcement access to data they store, regardless of whether that data is held domestically or abroad. This is an example of how one regulator can pose a threat to another regulator's requirements without hacking or themselves commit violations of law to demand access to your data. Although you

may in the end hand over your information as required by US law, at least you should have control over when and how this is done so that you are able to apply confidentiality measures before exchanging the information.

Implementing Information Governance Step by Step

You may have a clear view on how you would like to build up the records needed in ServiceNow to do proper information governance. If not, this is a suggested stepwise approach.

Step 1: design time elements

The most critical table to start injecting information into is the Information Object table. You can't do much wrong here, as you will be able to change the scope descriptions, naming and other metadata dynamically later – or apply business policies on who are allowed to inject and update what.

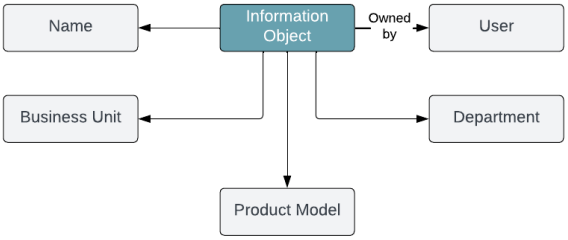

Figure 51 – Starting by injecting Information Object records per digital registry or bulk of information content and adding contextual, foundational data.

Let' use an example of an Information Object (Type) called "Core CIs in the CMDB". To know what business applications we use for keeping this CI registry, we need to add a Used by::Uses relationship to a business application.

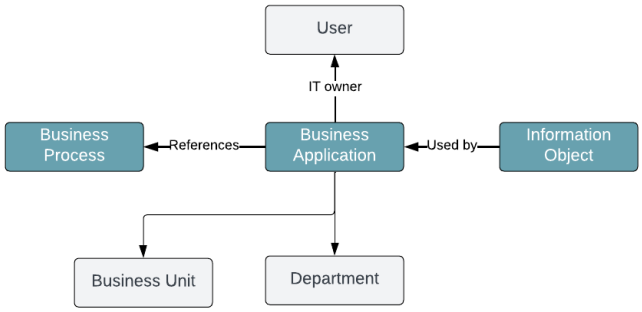

Figure 52 – Relate your Information Object (Type) to one or more relevant business applications which in turn can reference a business process.

Let's call the business application "ServiceNow Discovery". By opening this business application either directly from the list of business applications, or if you have the EA product, from the Enterprise Architecture Workspace, you will see the Information Object (Type) related list. By clicking on the the button **Add**, a pop-up window will allow you to search for and select the "Core CIs in CMDB" Information Object (Type), as well as to let you select the permissions this application has (create, read, update or delete records in this registry).

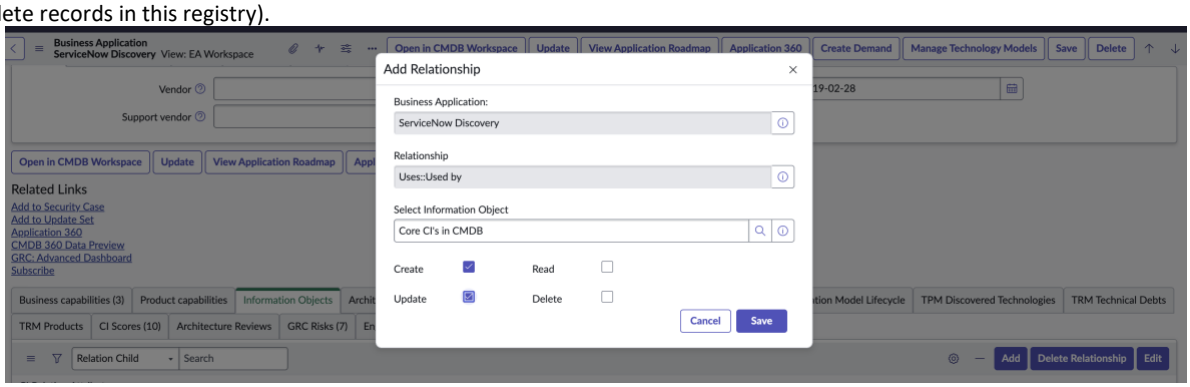

Figure 53 – Opening the related business application directly from list view and adding the Information Object (Type) from the related list in Enterprise Architecture Workspace.

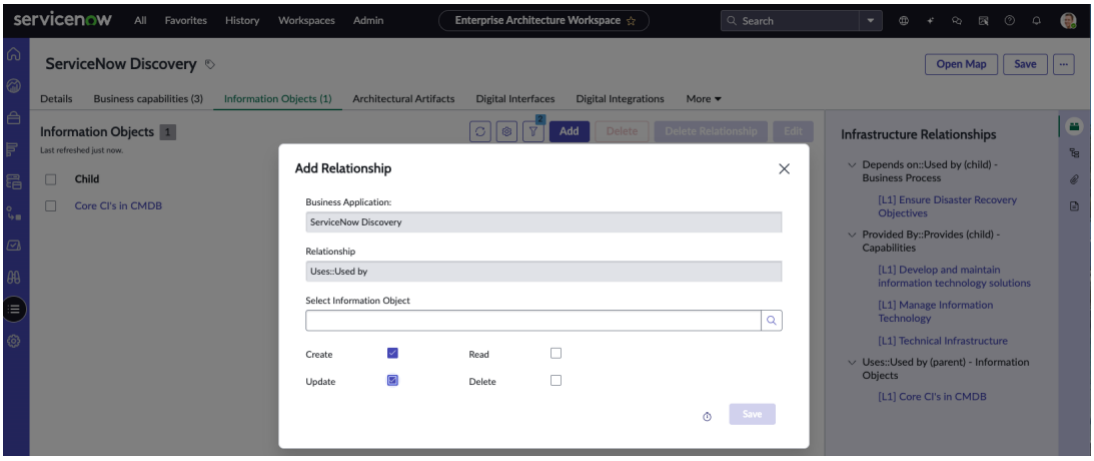

Figure 54 – Adding a relationship to relevant business application(s) from the Information Object (Type) tab in the Enterprise Architecture Workspace if you have the EA product.

You may want to see why you have this Information Object (Type) and what it's useful for.

Figure 55 – Adding a relationship from a business application to the business capabilities it underpins.

The business capabilities are only indirectly related to the Information Objects (Types) via the business applications. You can therefore stay in the business application form and select the related list/tab labeled Business Capabilities. As with the business applications to Information Objects (Types), you can leverage the two different methods (list view or EA workspace) to add a CI relationship between the business application and one or more business capabilities.

By opening the dependency view (label button or icon button), you will see something like this:

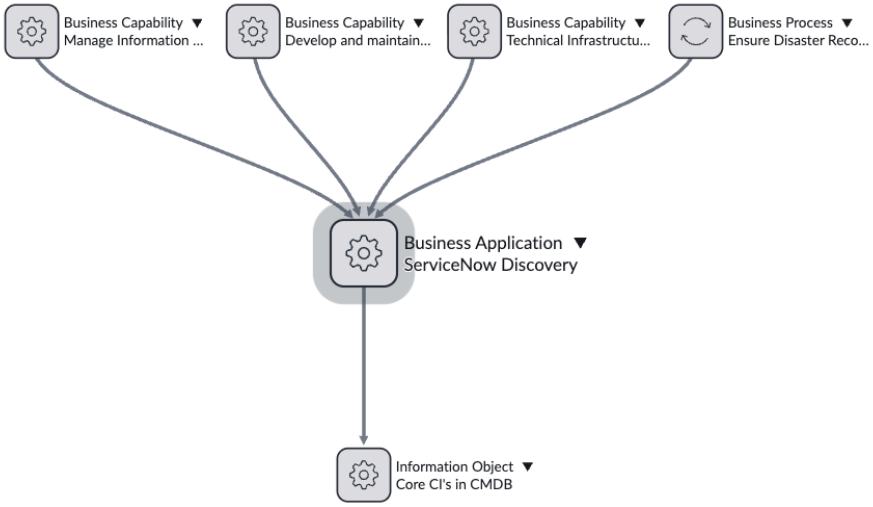

Figure 56 – Dependency view showing dynamically the related business application and business capabilities relevant for the Information Object (Type).

For the trained eye you will see that there is a Business Process called Ensure Disaster Recovery related to the business application. What kind of dependency this business process has to the Information Object (Type) can be determined using the Business Continuity Management solution.

Step 2: Relating your Information Objects to discoverable CIs

You now have the upstream CI relationships in place. Adding the CIs where your Information Object (Type) is stored is the next step. If you don't know the details around the technology behind the business application, you can relate your Information Object (Type) directly to the service instance of it using the CI relationship manager. It will preselect the correct CI classes Exchange Mailbox, Database Catalog, File system and Configuration file) when you choose Depends on::Used by (parent), but it's possible to add a fifth class (service instance or application service) manually.

Figure 57 – Relating your Information Object (Type) to the CIs where the data is stored directly from the relationship manager on the Information Object(Type) form.

You will now be able to keep track of both upstream and downstream relationship from an Information Object (Type) viewpoint.

Figure 58 – Showing how the service instance "ServiceNow Discovery PROD" is related to both the parent business application "ServiceNow Discovery" and the Information Object (Type) "Core CIs in the CMDB". We also see that the same Information Object (Type) has an upstream relationship to both "ServiceNow Discovery" and another business application called "IT Service Management".

By tying the service instance record to underlying technical CIs, you will be able to see what is planned, ongoing and past changes, incidents and other activities on those that may have an impact on your Information Object (Type). This will allow you to place policies on those that are important to control to ensure that they are maintained and protected according to the information's criticality, integrity and availability needs.

This is the necessary starting point for most of your information management processes.

Step 3: Showing who takes care of your discoverable information assets and who consumes them

Now it's time to show who's accountable for maintaining and providing the CIs and assets that your Information Objects (Types) are depending on. This is done by creating Technology Management Offerings, the ones that maintains and secures the CIs and assets, and the Business Offerings that describe how these can be consumed and the whole user or consumption experience. If you have the Strategic Portfolio Management product, you may want to leverage the Service Builder feature for this, which provides a user-friendly step-by-step guidance. Or you can do the same adding services and offerings from the list view and use the Application Service Wizard:

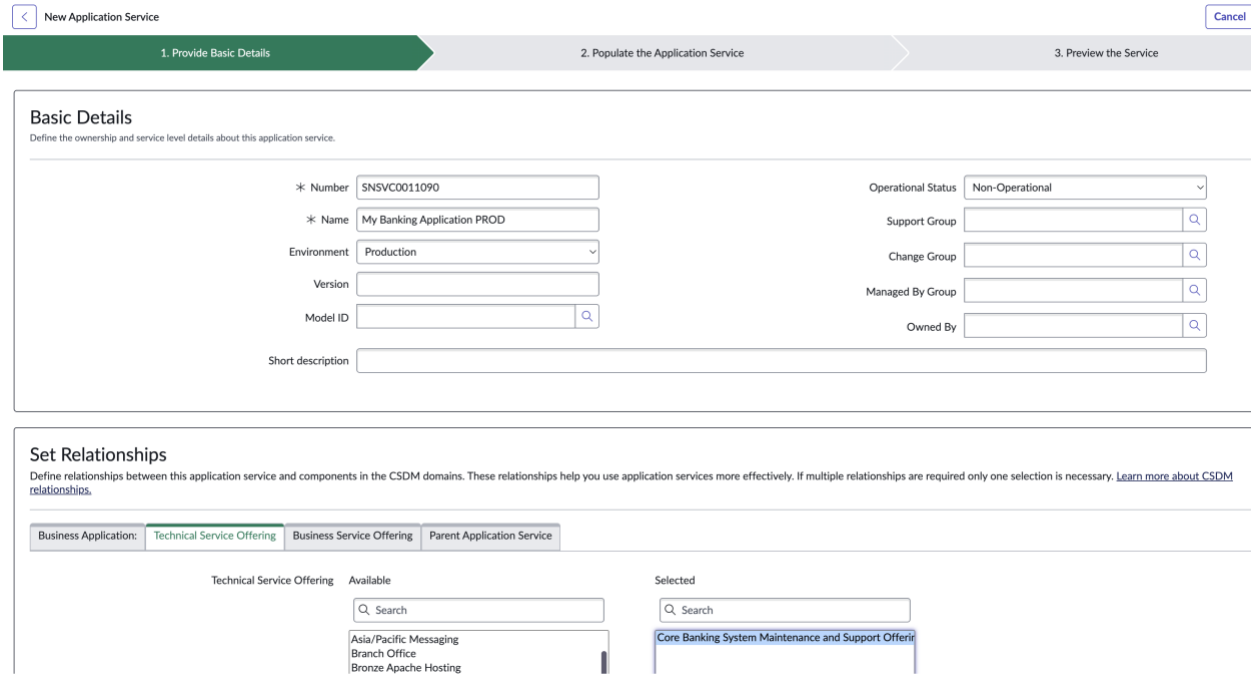

Figure 59 – Creating the relationships between service instances that your Information Objects (Types) are stored within and the different offerings that maintain and provide them using the Application Service Wizard.

A tip on how to differentiate between various teams (sys_user_group):

- The support group on the **business offering** handles cases/interactions/calls coming from business users. If there seems to be an outage, bug or other technical blocker, an incident is created and placed in a support group of a technology management offering.
- The support group on the **technology management offering** handles incidents that need a technical root cause analysis and potentially also result in a work item for engineers and a change request.

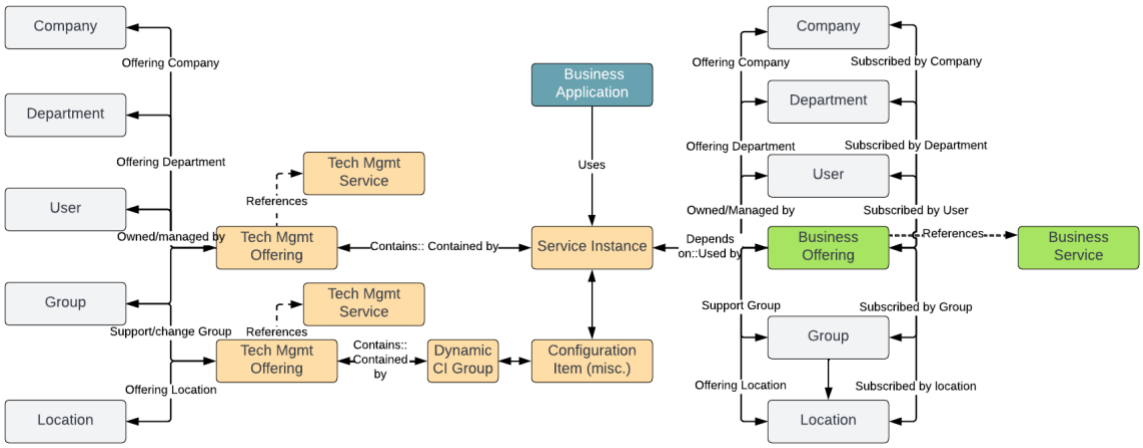

Figure 60 – How various levels of accountable/responsible or consuming people, teams, organization units and companies are referenced from the different service offerings.

Defining control objectives and controls using IRM

This section is for customers who use Integrated Risk Management or are considering acquiring this product suite. You will need to know some of the features in this suite to fully understand it.
If you have implemented UCF (Unified Control Framework) policies, you will have the opportunity to kick start the use of control objectives for governance of information assets and products, as well as the processes that have these as input or output and the services that offer information to systems as well as end users. But you also want to define control objectives on the storage of information content (information at rest), and the transfer of information (information in transit). Control objectives on this will be possible to measure via scripted controls that run against observable Configuration Items (CIs) such as databases, servers, networks and identity and access management tools such as Single Sign On solutions and Privileged Access Management systems. Thereby using live, empirical data collected through the actual IT service management processes such as Access Management and Change Management.

- ISO27001 defines Information Management through its Information Security Management System (ISMS), focusing on safeguarding information assets by identifying risks and implementing security controls to protect confidentiality, integrity, and availability (CIA) of information. It emphasizes a risk-based approach to managing information security risks and compliance especially aligned with GDPR requirements for data protection and privacy.
- ITIL approaches Information Management in the context of IT Service Management, providing practical processes and workflows to manage and deliver IT services efficiently, ensuring that information and services support business needs.
- COBIT provides a governance and management framework that includes control objectives for managing information and technology resources holistically, covering risk management, compliance, performance measurement, and alignment of IT goals with business strategies, integrating information security as part of overall IT governance.
- GDPR mandates organizations to protect personal data through appropriate technical and organizational measures, often relying on frameworks like ISO27001 and others to implement these requirements effectively. GDPR focuses on the lawful and secure handling of personal information, requiring risk identification and mitigation for risks such as unauthorized access or data breaches.

Handling Policy Exceptions

Once you have defined Policies, Control Objectives and Controls, you will soon find that there are deviations taking place in your organization. This is normal but should be controlled and monitored to avoid introducing exploitable vulnerabilities and thus risks. There may be plausible reasons for these deviations, as well as good ways of mitigating the risks – or simply accepting them for a short period of time.

Setting up the Policy Exception process will allow you to have an overview of the number of accepted deviations, as well as having a means of reminding the persons and teams behind these that they must resolve the underlying issues and become compliant.

This is where the criticality of the Information Object (Type) at stake becomes a valuable context. If the confidentiality level is very high, and/or the need for stable availability is also very high, a policy violation that potentially increases the risk of intrusion or unavailability should not be accepted unless the mitigations is trustworthy and sufficient. And the Information Owner should be part of the decision making.

This is hard to achieve unless the people in charge of the CIs in question know what Information Objects (Types) that are related to these. Creating a relationship between the Information Object (Type) and the database or other storage medium will allow transparency and allow for more elaborate controls that dynamically checks for criticality of the content in addition to the normal security policies that applies a to all CIs.

Reporting on Information Security policy violations

There are various forms of potential violations to your information assets. A common categorization of these are Confidentiality violations, Integrity violations and Availability violations, abbreviated as 'CIA'. Based on what kind of policy that is violated, there will be need for action from different stakeholders. If there's a confidentiality or integrity violation, the information owner needs to know about it, alongside the service owner who needs to maintain the service quality and the product owner who needs to know if any improvements to the product is called for to remediate and avoid it from happening again. But also, those responsible for safeguarding the assets and access privileges to these, needs to act. Lastly, if the data cannot be trusted, as will be the result of an integrity breach, the consumers of the information must be notified. If the violation is causing loss of availability to data, the consumers, service owners and product owner needs to be notified.

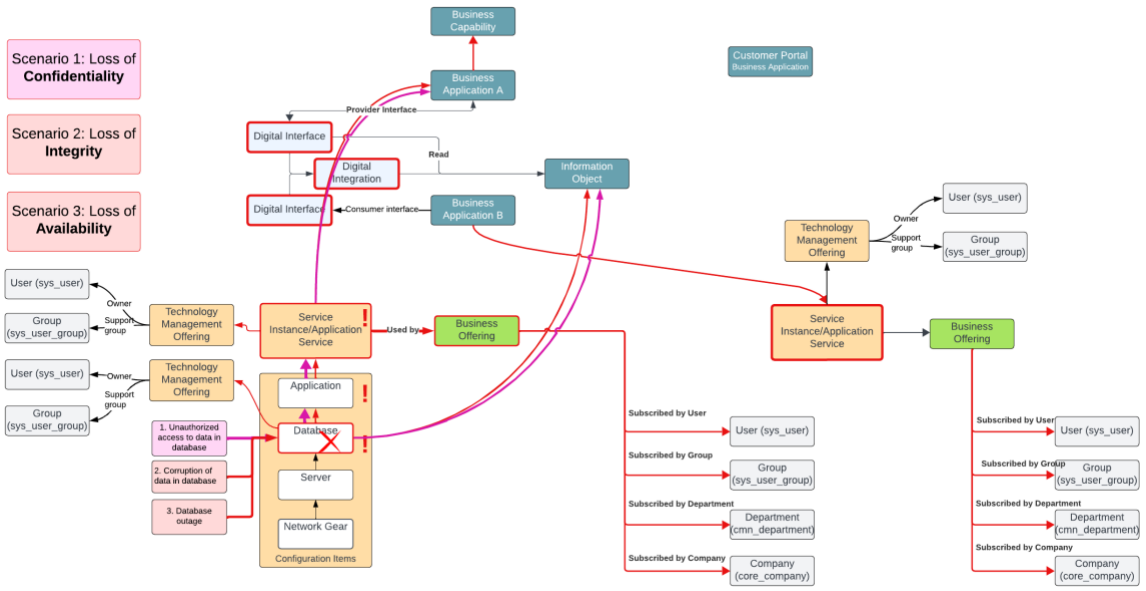

Figure 61 – Using various relationships to the impacted CI to determine who should be notified and who is responsible for fixing the issue. Although the Database team is responsible for troubleshooting, also the impacted Application team needs to be participating in case there's a need for any testing and mitigation on their side.

If Application B contains a multitude of Information Objects (Types) coming from various source systems, potentially not all consumers of it may be impacted. That is why the Information Object (Type) element can be important to model. When notifying the consumers that an outage potentially impacts their service, adding the Information Objects (Types) that are impacted as part of the message can be a good idea.

How prepared are you if something bad happens to your data?

We have now gone to great lengths to find out what data we have, what it is used for, how sensitive it is, and how we protect it, accordingly, using every trick in the book. Nevertheless, even the best companies can be affected. Sometimes, a once-in-a-millennium flood simply strikes. Unfortunately, resilience cannot be measured simply by compliance or maturity. Your most important company data may not be protected by any law or standard.

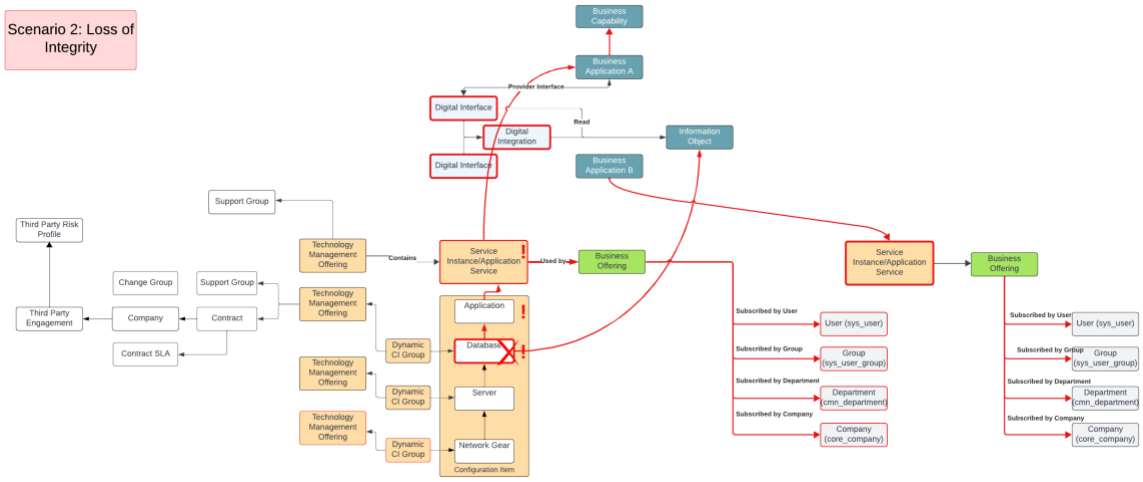

Figure 62 – You may have to involve a Third-Party vendor to resolve a data integrity issue. Your contracts should reflect the same service level requirements as those you have agreed with your consumers of the data.

We have seen that impacted and affected consumers may exist multiple steps away in the data lineage. But how do you ensure that your vendors take action to resolve an issue as fast as the strictest service commitments instruct? And are they following the same high availability and backup regimes as you do?

The following questions should be asked both for your internal IT organization *and* for your vendors.

That is why we have listed a few questions here that you can ask yourself and use your answers to determine the appropriate measures to take.

How often does your organization perform scheduled backups for critical data (Data which are essential for business to function)?

- Every 3 hours or more frequently
- Every 4 hours to 1 day
- Every 2 days to every week
- No regular backup schedule

Which of the following data backup practices are implemented within your organization? Several may apply

- ☐ Our data backups are encrypted while data is in transit, at rest, and in use
- ☐ Our data backups are assigned to storage tiers based on data criticality
- ☐ Our backup policy incorporates business risk and compliance needs
- ☐ We have continuous replication within a single datacentre or zone
- ☐ We have continuous replication across geographies zones
- ☐ We have deduplication and file compression built-in

Which of the following most accurately describes your organization's disaster recovery (DR) plan?
No DR plan formalized

- Formal DR plan in place, but may be outdated and lacks staff training or exercises
- DR plan ready to implement, but staff training and exercises are inconsistent

- Comprehensive DR plan with tested procedures, regular staff training, and periodic DR exercises

How often does your organization conduct Disaster Recovery (DR) testing?
Less than once per year
- About once per year
- About twice per year
- About once per quarter
- More than once per quarter
- My organization does not formally test our recovery capabilities

Which of the following most accurately represents your organization's ability to meet RTOs (Recovery Time Objectives) and RPOs (Recovery Point Objectives) for workloads using critical data?
- Formal RTOs and RPOs are not defined
- RTOs and RPOs are defined but not tested or tracked
- RTOs and RPOs are defined, recovery solution(s) can meet them on normal days
- RTOs and RPOs are defined, recovery solution(s) can meet them even on days with many severity 1 incidents/outages

How fast can your organization scale its secondary infrastructure (i.e., DR site/cloud/alternative infra/hot site) in cases of outage?
- Less than 1 day
- About 1 day
- Between 2-6 days
- Between 1-2 weeks
- 2 weeks or more

Which of the following most accurately describes your organization's ransomware response plan?
- No ransomware response plan formalized
- Formal ransomware response plan in place, but may be outdated and lacks staff training or exercises
- Ransomware response plan ready to implement, but staff training and exercises are inconsistent
- Comprehensive ransomware response plan with tested procedures, regular staff training, and periodic ransomware response exercises

Which of the following most accurately describes your organization's data monitoring, dashboarding, and reporting capabilities?
- Comprehensive regulatory/compliance reporting (e.g., for GDPR, HIPAA, SOC)
- Reporting on incident types, trends, and response times
- Insight/frequent review of capacity demand

Which security use cases related to cyber incident response and data backup and recovery processes based on your data criticality are implemented at your organization?

- Real-time logging and advanced forensic capabilities for a detailed reconstruction of a ransomware attack timeline
- Third-party ransomware response team retained for detection, response, and recovery
- System restoration from clean backups as a part of cyber-incident response plan
- Network monitoring tools to log all security-related events (e.g., SIEM) in place

Which security use cases related to cyber incident response and data backup and recovery processes are implemented at your organization?

- Defined ways to understand context and usage pattern of data being made resilient
- AI data change pattern analysis to detect corrupted/infected data
- Prediction of potential outages/recovery failures
- Proactive analysis of potential data impact (e.g., of migration, updates)
- Data protection solution utilizes an AI assistant

All the measures listed here can be planned and reviewed using the ServiceNow GRC modules. The BCM module is also very helpful in this regard.

servicenow

Big thanks to:

Averell Pscheidl, for providing an angle to this guide and coaching on content
Mark Bodman for inspiration guidance on the CSDM
Scott Lemm for providing detailed models and clarifications
Barry Kant, for providing feedback on models and description
Bruno de Graeve, for counselling on the Digital Integrations Management chapter and on stakeholder perspectives

Table of figures

Nomenclature		
Name	**Short definition**	**Table Name**
Business Capability	Business capabilities refer to the specific abilities or competencies that an organization possesses to achieve its goals and objectives. These capabilities are the essential building blocks that enable a company to deliver its products or services, operate effectively, and compete in the market.	cmdb_ci_business_capability
Business Service	is a service type that is published to business users, and it typically underpins one or more business capabilities	cmdb_ci_service_business
Business Service Offering	consist of one or more service commitments that uniquely define the level of service in terms of availability, scope, pricing, and other factors	service_offering (classification=business)
Technical Service	is a service type that is published to service owners and typically underpins one or more business or application services	cmdb_ci_service_technical
Technical Service Offering	is a service offering type defined as a stratification of the technical service into options including localization/geography, environment, pricing, availability, capability, support group (for incident), technical approval group (for change), and packaging options (commitments)	Service_offering (classification=technical)
Business Process	is a method of related steps that stakeholders take to achieve a business goal. The Business Process is a manually maintained configuration item that can identify criticality, both declared & determined, as well as impact to confidentiality, integrity, and availability.	cmdb_ci_business_process
Value Stream	is a sequence of activities that helps you visualize the flow of a process from start to finish, the value of each step in the flow, and the application models associated with each step in the flow.	cmn_value_stream
Business Application	represents all Software and infrastructure (For example catalogue of titles) configured to provide business functionality. Business applications are the logical representation of all instances, used to increase productivity and to provide functionality to perform business functions accurately (For example payables, receivables, general ledger).	cmdb_ci_business_app
Application Service	is a service type that is a logical representation of a deployed system or application stack.	cmdb_ci_service_auto
Dynamic CI Group	is a dynamic grouping of configuration items (CIs), based on results of CMDB Groups queries. Dynamic CI Group uses CMDB Group to identify Cis of common criteria.	cmdb_ci_query_based_service
Information Object (Type)	is part of the new information portfolio and referenced by the business application. The Information Object (Type) table may be used to identify the types of data a business application may possess such as PII, PCI, HIPAA, etc	cmdb_ci_information_object

Nomenclature		
Name	**Short definition**	**Table Name**
Digital Interface	Digital interfaces are provided as part of a business application, but they can also stand on their own. Interfaces provide a way for other business applications to interact with the applications.	sn_apm_di_digital_interface
Digital Integration	The digital integration represents the integration between two business applications. In a typical scenario there would be a consuming business application, a provider business application, and an interface that is provided by the provider business application. The digital integration is a design object used by the Enterprise Architects.	sn_apm_di_digital _integration
CI relationship	Configuration management is not effective without the use of relationships between CIs. Not all objects in the CSDM conceptual model are CMDB tables. Additionally, not all the objects have relationships. Several ServiceNow products, such as APM, have a critical dependency on the relationships listed above. If these relationships are not utilized (ex. Business Application "consumes" Application Service) then functionality such as the Technology Portfolio Management risk assessment will not function. Additionally, the relationships commonly created as part of ITOM Service Mapping & ServiceNow Discovery are considered the standard for infrastructure CIs.	cmdb_rel_ci
Reference	Most tables reference data from other tables, without generating a CI relationship. When a table has a drop-down menu or value list and could have been referencing another table, you can configure these. That way you don't have to update data two different places and can leverage the reference by adding related lists later.	(no specific table)
Products	Products are specific versions or configurations of a product used for managing and tracking through various ServiceNow platform applications. Within ServiceNow these products are recorded as Product Models. There exist multiple categories of product that extends the root table.	cmdb_model
Account	An account is a customer or a partner who sells to and supports other customers. Accounts can be customer accounts, partner accounts, or both.	customer_account
Software Component	Building block in the decomposition of Business Applications and other Digital Systems especially useful for Service Delivery Life Cycle management value streams.	Multiple tables

Anne Kristine Naess, Cand.Polit., and Senior Advisory Enterprise Architect working for ServiceNow Norway. She was prior to this a ServiceNow customer for nine years, being part of teams implementing nearly all ServiceNow's IT management and governance products as well as some business implementations.
https://www.linkedin.com/in/kristinenaess/

Rob Koeten, Engineering Fellow, Chief Architect and data model expert in ServiceNow, who co-authored the CSDM 5.0 white paper defining ServiceNow's common service data model framework. As an Engineering Fellow (a senior leadership technical role), he specializes in enterprise architecture, CMDB design, and large-scale information governance across the ServiceNow AI Platform.
https://www.linkedin.com/in/robertkoeten/

For More Information

www.servicenow.com