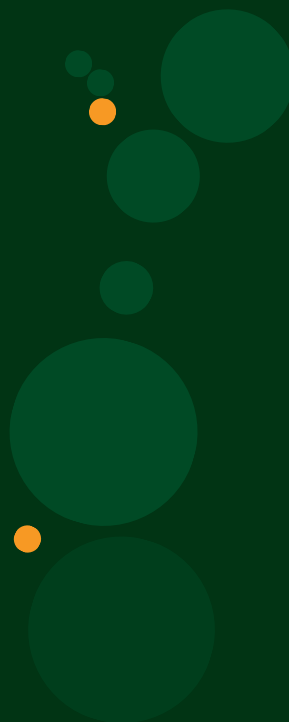
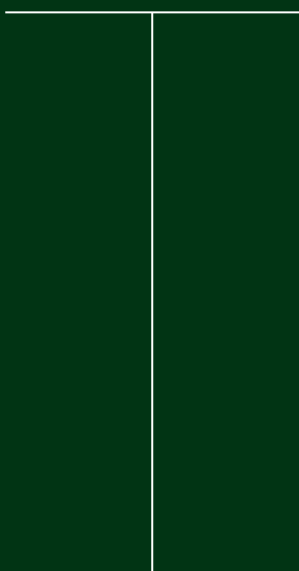


FÅ SUCCES MED DINE INTEGRATIONER

AF SEBASTIAN BENNETT FRANDSEN



Øget konkurrence, afkoblede arkitektur-strategier og regulativer stiller krav til organisationers evner til at udveksle data. Det er derfor en god forretning at investere i udvikling og forbedring af integrationsarbejdet. Artiklen belyser tre simple trin, som sikrer succes med udvikling og forvaltning af integrationer."



FÅ SUCCES MED DINE INTEGRATIONER



INTRODUKTION

Data er i dag et af de mest værdifulde og eftertragtede aktiver, men der findes dog fortsat et enormt uforløst potentiale i brugen af det. Derfor bestræber de fleste organisationer sig på at lære at udnytte data bedre. Eftersom integrationer er den mest effektive løsning til at flytte rundt på og anvende data, stiger vigtigheden af organisationers integrationskapabiliteter på dette område markant år for år. Alle organisationer tvinges altså til enten at udvikle disse kapabiliteter internt eller at gå ud og købe dem i markedet.

Hvad end man ønsker at opnå en forbedret digital kundeoplevelse, effektivitet gennem STP (Straight Through Processing) eller skalérbare processer og compliance er integrationer kommet for at blive.

Vores erfaring er dog, at arbejdet med integrationer typisk øger kompleksitet og omkostninger i forretningsdrevne IT-forandringer.

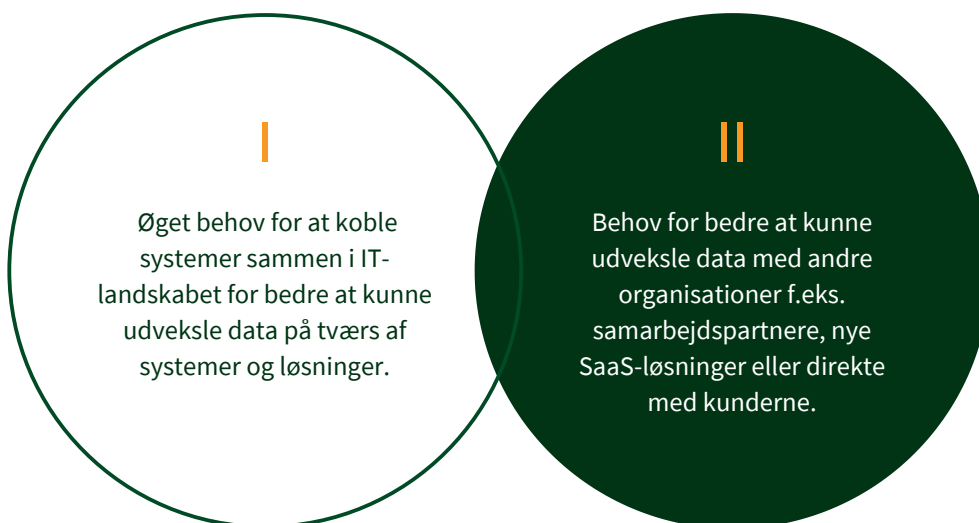


Virksomheder står over for et stigende antal udfordringer relateret til dataintegration primært på grund af den voksende mængde data, regulatoriske krav, behovet for information i realtid, øget data-kompleksitet samt distribution af data på tværs af hybrid og cloudløsninger.

Det er velkendt, at nutidens forretningsbrugere ønsker hurtig adgang til pålidelige oplysninger i realtid, som kan hjælpe dem med at træffe bedre beslutninger. En moderne strategi for dataintegration er afgørende for at understøtte denne nye generation af forretningskrav."

Forrester

SÆRLIGT TO BEHOV DRIVER GRUNDEN TIL AT MODNE ELLER UDVIKLE NYE INTEGRATIONER



Behovene kan adresseres på mange måder, men uanset tilgang undervurderes omfanget af analysearbejdet, der forudsætter udviklingen af nye integrationer, ofte.

Hvis organisationer ikke har analysearbejdet på plads inden udviklingsforløbet, kan der opstå spørgsmål om alt fra hvilke systemer, der skal integreres til, hvilket system der ejer data, hvilke data der skal overføres, samt hvad data rent faktisk betyder. Konsekvensen er typisk dårligt udviklede løsninger, overskredne budgetter og løsninger på falske problemer.

Derfor er det vigtigt, at organisationer har styr på deres behov og forretningsprocesser og ikke mindst den underliggende business case, før man kaster sig over udviklingen af nye integrationer på kryds og tværs af sin IT-arkitektur.

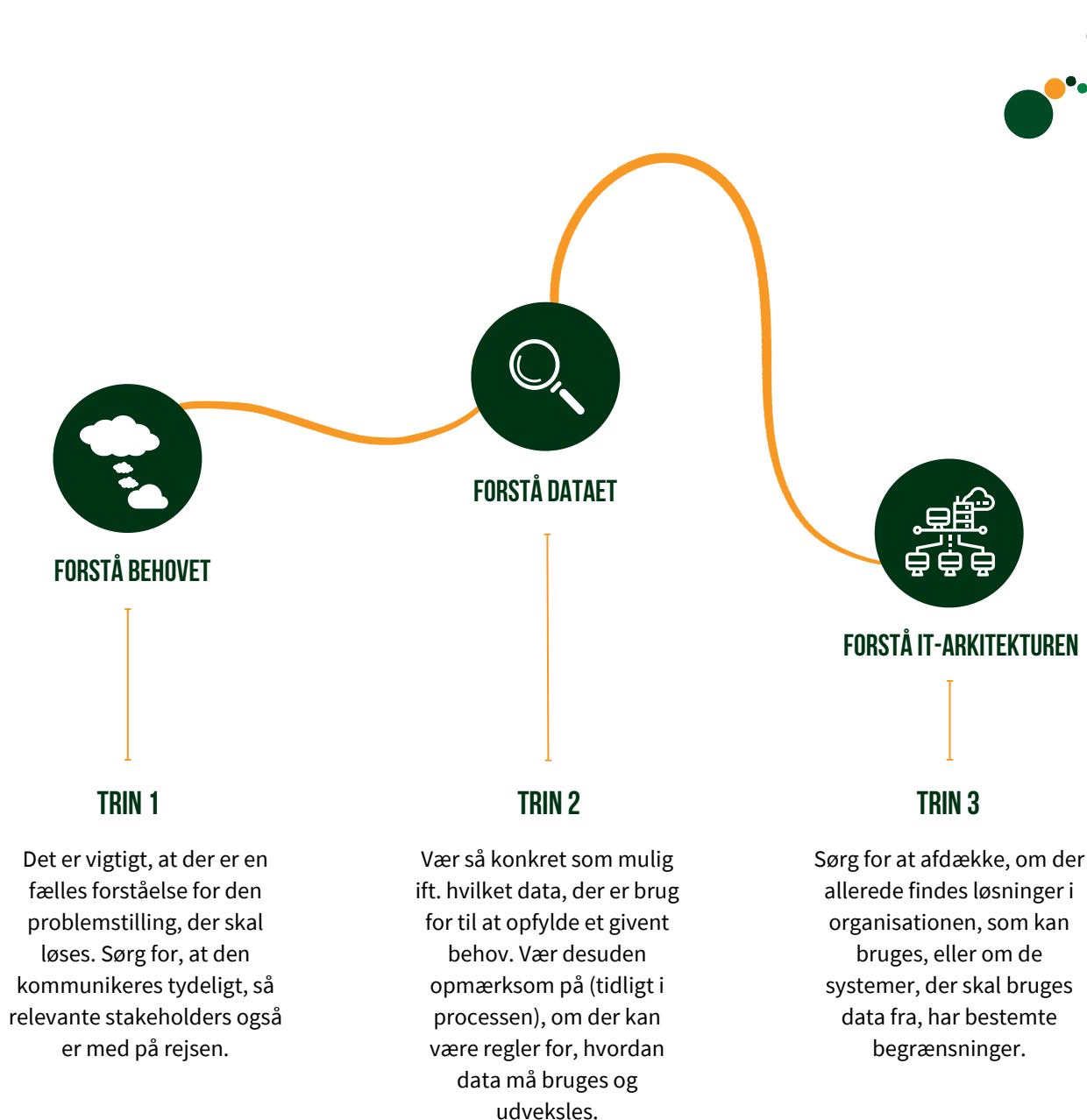
HVAD ER EN INTEGRATION?

En integration er en metode, som f.eks. kan bruges til at sammenkoble systemkomponenter eller til at flytte data fra system A til system B.

Sat på spidsen er der således ingen i forretningen, der har behov for en integration. Det egentlige forretningsbehov er nærmere den ekstra funktionalitet, der bliver tilgængelig ved at sammenkoble systemer, eller de nye måder at bruge data på, når de er blevet tilgængelige i system B. Integrationer er altså "blot" et middel til et mål.

Dog er udviklingen af integrationer ikke altid ligetil – det kræver et solidt kendskab til sine forretnings- og databehov samt sit IT-landskab.

KOM GODT FRA START MED INTEGRATIONSUDVIKLING MED TRE SIMPLE TRIN





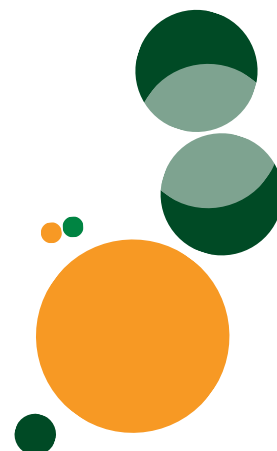
Det kan virke banalt, men én af de største udfordringer ved integrationsarbejde er, at integrationsløsningen (servicen) ofte ender med at få skylden for dårlig datakvalitet eller uklare/uafklarede krav til enten kilde eller det modtagende system. Det skyldes, at der typisk mangler overordnet arkitektur-ejerskab – en form for integrationsarkitektur product management, hvor ”nogen” har ansvaret for den samlede arkitektur herunder integrationsarkitekturen.

Det kan være en overvældende opgave i mange organisationer, da man både kæmper med monolitiske legacy-systemer (operating backbone) og ”digitale ambitioner”. Men det kan lade sig gøre, og det starter med at blive enige om rammerne og få begrebsliggjort integrationsarbejdet på tværs af forretning og IT.”

Jens Rasmussen
CIO, Chr. Hansen

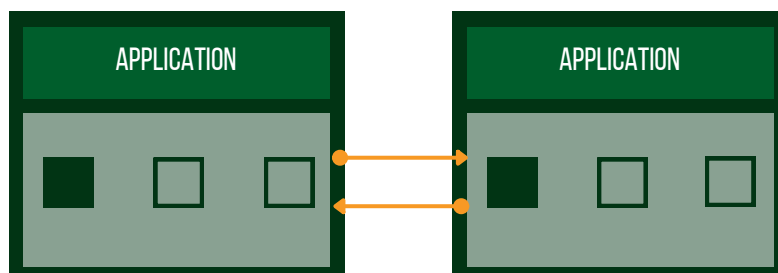
HVILKEN INTEGRATION ER BEDST?

Der findes ikke nogle one-size-fits-all integrationer, men der findes forskellige integrationsmønstre og stilarter, som er gode til at løse specifikke problemstillinger.



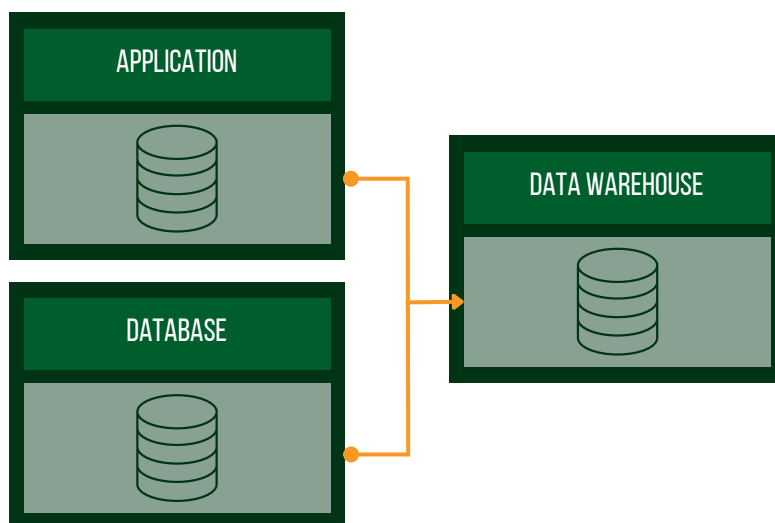
EKSEMPEL 1

En integration mellem to systemer som skal sammenkoble og konfigurere forskellige komponenter til at fungere som ét samlet system. Dette kan have mange forskellige formål, f.eks. at forbedre funktionaliteten eller ydeevnen i systemet.



EKSEMPEL 2

En anden slags integration er en dataintegration. Her er formålet ofte at sammenkoble og kombinere data fra forskellige kilder såsom databaser, filer eller andre systemer. Dette kan gøres for at opnå en bedre forståelse af data. Dataintegration kan også involvere bevægelse, transformation og konsolidering af data, så data kan anvendes til et specifikt formål.



TRIN 1

FORSTÅ BEHOVET

Inden man begynder at bygge sin integrationsløsning, er det vigtigt, at man har styr på hvilket forretningsbehov eller problem, der skal løses. Ellers risikerer man at forsøge at løse de forkerte problemstillinger – og ender måske med en løsning på et falsk problem.

KEND ORGANISATIONENS FORRETNINGS- OG DATABEHOV

Start med at forstå og beskriv data use cases – altså hvad dataet skal bruges til.

FOR EKSEMPEL

RAPPORTERINGS USE CASE	'DATAUDVEKSLING MED PARTNER' USE CASE	AI USE CASE
Skal dataet bruges til yderligere rapportering på baggrund af flere sammenkørte datakilder og systemer?	Skal dataet udveksles med en strategisk partner?	Skal dataet indgå som træningsdata til en ny machine learning model?

Mulighederne er mange, men en veldefineret use case vil gøre det nemmere at afdække hvilke data, der er nødvendige for at understøtte løsningen, herunder forretnings- og compliancekrav. Derudover vil det også blive nemmere at kommunikere behov til relevante interessenter i sin organisation, f.eks. systemejere eller integrationsudviklere.

HVOR OFTE ER DER BEHOV FOR AT MODTAGE NYESTE DATA?

Det er ligeledes vigtigt at definere behovet for frekvens. Hvor ofte har man brug for at modtage nyeste data? Jo mere nøjagtig frekvensen specificeres, jo bedre forudsætninger er der for at udvikle en solid løsning. I praksis ses det ofte, at krav formuleres som "vi skal have data hurtigst muligt, så tæt på realtid som muligt", selvom forretningsbehovet sagtens kan understøttes af blot daglige opdateringer af data eller måske endda på ugentlig basis (batch jobs).

Der er markant forskel på løsningsdesignet for en løsning, som skal understøtte data i realtid og én, som skal understøtte daglig opdatering af data.

En realtidsløsning stiller ikke kun krav til, at det konstant er muligt at trække ny data fra kildesystemet, men også at den modtagende løsning er i stand til at konsumere den løbende strøm af data. Selvom det er fristende at modtage data få sekunder efter, det er genereret, bør det overvejes, om det er absolut nødvendigt.



KONKRET SKAL MAN:



Udarbejde en detaljeret beskrivelse af, hvilke forretningsbehov eller use case, som integrationen skal understøtte. Heriblandt hvor man forventer, at data skal overføres fra og til.



Identificere hvor ofte man har brug for nyeste data for at understøtte forretningens behov. Husk at løsningskompleksiteten stiger, hvis det skal være realtid.

TRIN 2

FORSTÅ DATAET

Når man har styr på forretningsbehovet, er det tid til at gå et spadestik dybere i forhold til sine konkrete databehov.

En solid forståelse for ens data er nødvendig, når man skal bygge nye integrationer. Forsknings- og rådgivningsfirmaet Gartner beskriver ofte, hvordan 'data literacy' er en nødvendighed ikke bare for understøttelse af integrationer men for udvikling af langt de fleste digitale kapabiliteter.

Data literacy kan beskrives som evnen til at læse, skrive og kommunikere data i kontekst samt forståelsen for datakilder, analytiske metoder og hvilken værdi, data skaber.

En høj grad af data literacy vil gøre det nemmere at identificere, hvilke konkrete dataelementer, man skal bruge fra bestemte datakilder. Det vil også gøre det nemmere at have en dialog med relevante SME'er fra forretningen eller IT, som kan bidrage med yderligere indsigt i dataet.

KONKRETISÉR FORRETNINGSBEHOVENE I DATAELEMENTER

Det første skridt er at blive så konkret som muligt i forhold til hvilket data, I har behov for til at kunne understøtte jeres forretningsbehov. Skal I f.eks. bruge data om huse, så bryd det ned til konkrete dataelementer. Dette kunne være vejnavn, husnr., byggeår, areal, grund osv..

”

Data literacy er den vigtigste færdighed i den digitale økonomi.

Data kommer med et stort uudnyttet potentiale i relevante og måske endnu ikke-erkendte sammenhænge. Integrationscapabilities og -metoder er måden hvorpå, man høster værdien af sine dataaktiver”.

Jens Rasmussen,
CIO, Chr. Hansen

Hvis I allerede nu ved hvilket system(er), data skal komme fra, kan det ofte være en god idé at alliere jer med enten systemejere eller dataejere, som kan hjælpe med at identificere hvilket data, der er til rådighed og eventuelt, hvad et givet dataelement hedder. Der findes måske ikke et separat dataelement for adresse, men i stedet er det kun muligt at få adressen via en kombination af vejnavn og husnr. Det er et helt normalt scenarie, at data findes modelleret på anden vis, end hvad der understøtter fremtidige use cases.

Derudover er det vigtigt at være opmærksom på, om der er data, som mangler. Hvis I f.eks. ønsker at analysere ændringen i solgte huse, men I mangler data om salgspriser for solgte huse, kan dette begrænse jeres analyse.



FÅ STYR PÅ BEGRÆNSNINGER OG KRAV

Når man begynder at flytte data fra ét system til et andet via integrationer, risikerer man at afvige fra det oprindelige formål om "hvorfor og hvordan vi opbevarer data". For nogle typer data vil det ikke være et problem, da det f.eks. er ens egen data eller ikke indeholder personfølsom information. Typisk vil data dog indeholde forskellige former for følsomme informationer, som ikke uden videre må deles mellem systemer uden det fornødne samtykke.

Det ses ofte, at produktionsklare løsninger sættes på hold i sidste øjeblik, fordi der for sent identificeres compliancemæssige problemer i løsningen. Det er derfor vigtigt at afstemme behov og løsning med relevante interessenter, såsom compliance (inkl. DPO) og procesejere.

KONKRET SKAL MAN:



Oprette en liste med de konkrete dataelementer, som man har behov for samt en kort beskrivelse af hver enkelt af dem. Yderligere detaljer kan f.eks. være, hvad dataelementet hedder i det system, som man skal have data fra samt et eksempel på data.



Gennemgå datalistens og forretningsbehovene sammen med f.eks. jura, compliance eller en dataejer for at afklare, hvordan og om man kan benytte al data på listen.

TRIN 3

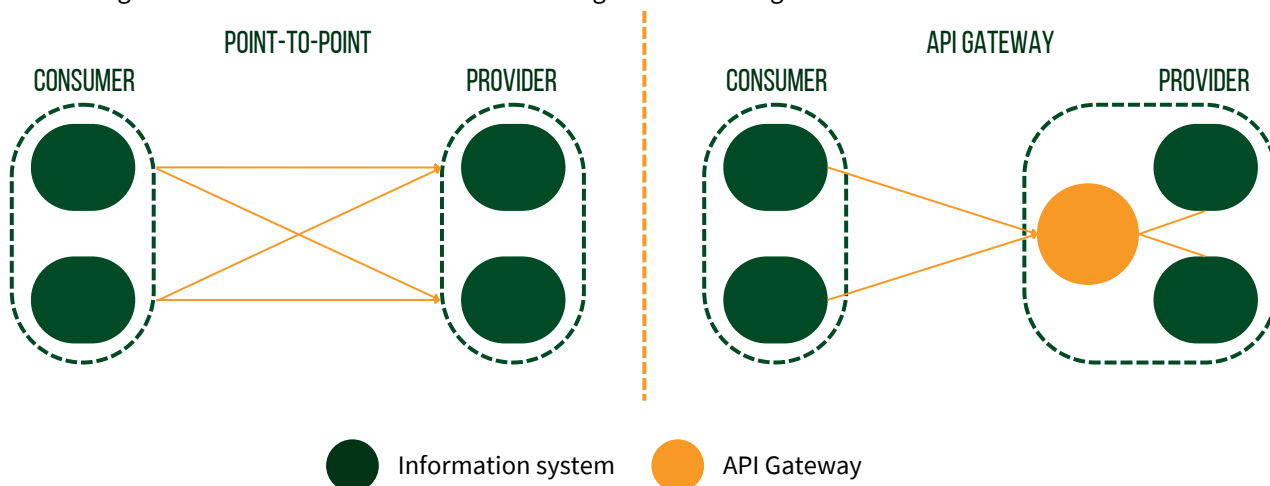
FORSTÅ IT-ARKITEKTUREN

Sidste trin går ud på at analysere og vurdere IT-arkitekturen med fokus på de komponenter, som data skal hentes fra. Alt efter behovet kan dette omfatte et eller flere systemer og kan være alt fra store monolitiske kernesystemer til domænesystemer, såsom f.eks. salg og HR til mindre skræddersyede/hjemmebyggede applikationer. Det er vigtigt at forstå arkitekturen og de respektive komponenters muligheder og begrænsninger ift. de ønskede integrationer.

API ELLER POINT-TO-POINT INTEGRATION?

Det er ikke altid nødvendigt at integrere direkte til det system, hvor data bliver skabt (kildesystem). I større organisationer med mange systemer er det ofte mere udbredt at integrere gennem f.eks. en Enterprise Service Bus (ESB) eller API gateway fremfor såkaldte point-to-point integrationer.

Hvor point-to-point er direkte integrationer mellem systemer, er en API gateway et slags mellemlid, der videresender forespørgsler til de relevante systemer. På den måde kan man nøjes med at integrere til en API gateway og dermed få "adgang" til de systemer, som er koblet på. Dette kan være en fordel, især for organisationer med mange IT-systemer, da det betyder, at de individuelle systemer ikke er lige så hårdt sammenkoblet som med point-to-point integrationer og dermed mere fleksible ift. skalerbarhed og videreudvikling.



Alternativt findes der (i nogle tilfælde) et centralt datalager i form af en data warehouse løsning eller lignende, som allerede har aggregeret data fra en lang række systemer, som måske også kan understøtte en række forskellige forretningsbehov. I sådan en situation kan det være bedre at integrere direkte med data warehouse løsningen frem for de individuelle systemer, da en enkelt integration kan give adgang til data fra flere systemer.

HVILKE INTEGRATIONSMETODER UNDERSTØTTER SYSTEMERNE?

Efter at have identificeret hvilke systemer, data skal udledes fra, er næste skridt at undersøge hvilke slags integrationsmetoder, systemerne understøtter. Nogle systemer understøtter kun muligheden for at sende og læse flade filer via SFTP, hvor andre udstiller et API, som kan bruges til at skrive og læse data til og fra. For API'er er det særligt metoderne REST og SOAP, som er udbredt, men GraphQL vinder også frem. De er beskrevet nedenfor.

REST Representational State Transfer API	SFTP Secure File Transfer Protocol	GRAPHQL
Et REST API består af en række endepunkter (også kaldet "URL'er"), som kan bruges til at hente eller sende data. Hvert endepunkt repræsenterer en bestemt ressource, og forespørgsler til endepunkterne returnerer et svar (response).	SFTP er en metode til at overføre filer over internettet på en sikker måde. Det gør det muligt at sende og modtage filer fra én computer til en anden. Det er en god løsning, hvis man har brug for at sende store filer, da den kan håndtere store mængder data.	GraphQL fungerer ved at lade klienter sende specifikke forespørgsler via en specifik syntaks (dvs. et sæt regler for, hvordan forespørgsler skal skrives) til et API, som derefter returnerer præcis de data, der er blevet bedt om i et standardiseret format.

Uanset om det er REST, GraphQL, SFTP eller noget helt fjerde, er det vigtigt at bekræfte hvilke data, der kan tilgås via den givne integrationsmetode.

Hvis data skal tilgås via et givent systems API'er, er det vigtigt at gennemgå dokumentationen for API'erne for at sikre, at de rent faktisk udstiller det data, som man har behov for at tilgå. Mange organisationer dokumenterer deres API'er i OpenAPI-standarden vha. værktøjer som f.eks. Swagger, hvilket gør det nemt at læse dokumentationen og forstå hvilket data, der tilbydes.

VÆR OBS PÅ!

Det er vigtigt, at man ikke begrænser sig til kun at kigge på det system, som man skal have data fra, men også undersøger det system, som man vil sende data til. Det ses f.eks. ofte, at en end-to-end arkitektur nemt kan integrere til en veletableret API gateway for til hver en tid at kunne hente de data, der er behov for, men at der til gengæld opstår en flaskehals ved de modtagne systemer, fordi de kun kan modtage data via en bestemt metode og kun kan læse det ind én gang i døgnet.

Derfor bør disse afklaringer drives i tæt samarbejde med f.eks. integrationsarkitekter eller udviklere, da det påvirker, hvordan det endelige løsningsdesign for integrationen kan se ud. I værste fald kan nogle af systembegrænsningerne betyde, at man ikke kan opfylde det oprindelige forretningsbehov.

KONKRET SKAL MAN:



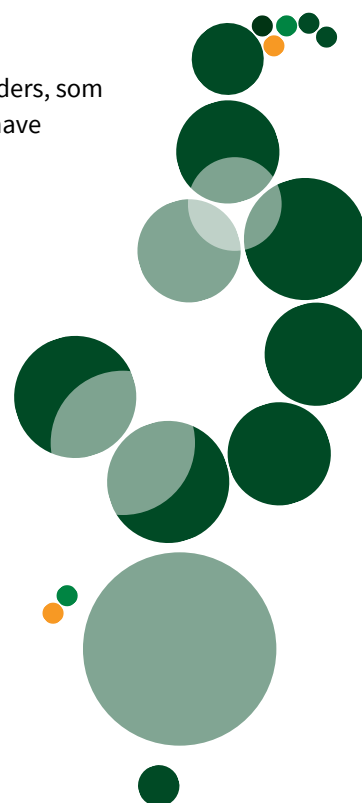
Undersøge hvilke konkrete systemer eller løsninger i IT-landskabet, der kan levere den nødvendige data. Dette behøver ikke altid være direkte fra systemet, hvor data bliver skabt, men kan også være fra centraliserede løsninger, såsom et data warehouse.



Identificere hvilke integrationsmetoder, de nødvendige systemer understøtter. Udstiller systemet f.eks. et REST API, eller er det kun i stand til at sende filer via SFTP?



Sørge for at have et tæt samarbejde og dialog med relevante integrationsarkitekter, udviklere, systemejere og andre stakeholders, som kan hjælpe med at tydeliggøre, hvordan systemlandskabet kan have indflydelse på den endelige løsningsopbygning.





OPSUMMERING

TRE SIMPLE TRIN GIVER SOLIDT FUNDAMENT TIL INTEGRATIONSUDVIKLING

Der bliver et større og større behov for at have data lettilgængeligt. Hvad end dataet skal bruges til at realisere forretningens use cases, sikre højere automatiseringsgrad eller udnytte potentialet ved kunstig intelligens, stiller det krav til, at man som organisation forbedrer sine evner til at udvikle og arbejde med integrationer.

Det kan virke banalt, men første skridt på vejen er ofte at få etableret veldefinerede rammer og metoder for arbejdet med integrationsudvikling. Gennem artiklen er der givet et bud på dette via tre relativt simple trin, som belyser, hvordan man som organisation kan komme godt fra start.

De tre trin indebærer, at man 1) forstår sit databehov, 2) forstår selve dataet, og hvordan denne "ser ud", og 3) forstår sin IT-arkitektur, herunder hvor data kan udledes fra og til.



Trinene er som nævnt relativ simple, men har man styr på dem, giver de et solidt fundament til selve integrationsudviklingen og i sidste ende til en bedre og mere effektiv udnyttelse af organisationens data.

AF SEBASTIAN BENNETT FRANDSEN
 SOLUTION ARCHITECT



Trustworks er et IT Management konsulenthus, hvis formål er at rådgive vores samarbejdspartnere indenfor IT-arkitektur, forretningsanalyse og projektledelse.

Trustworks har omfattende erfaring med at understøtte strategisk udvikling og digitalisering i den finansielle sektor, grøn omstilling, offentlig arkitektur og pharma – alt fra idé til omstilling og implementering.



TRUSTWORKS

RETHINKING IT AND BUSINESS