Using Data Lifecycle Management to Improve Operational Performance

Webinar

Piet Knijnenburg

30th Oct 2018

interfacing.

25 years of experience in the field

Global client base in 46 countries

International offices & partner network

Recognized as a Leader by

KMWorld Trend-Setting PRODUCTS OF 2018

Gartner®
2018 Gartner® Enterprise Digital Twin Leader

SR 2018 Silicon 70
The Most Representative 70

100 COMPANIES That Matter in Knowledge Management 2018

Gartner®
2017 Enterprise Business Process Analysis

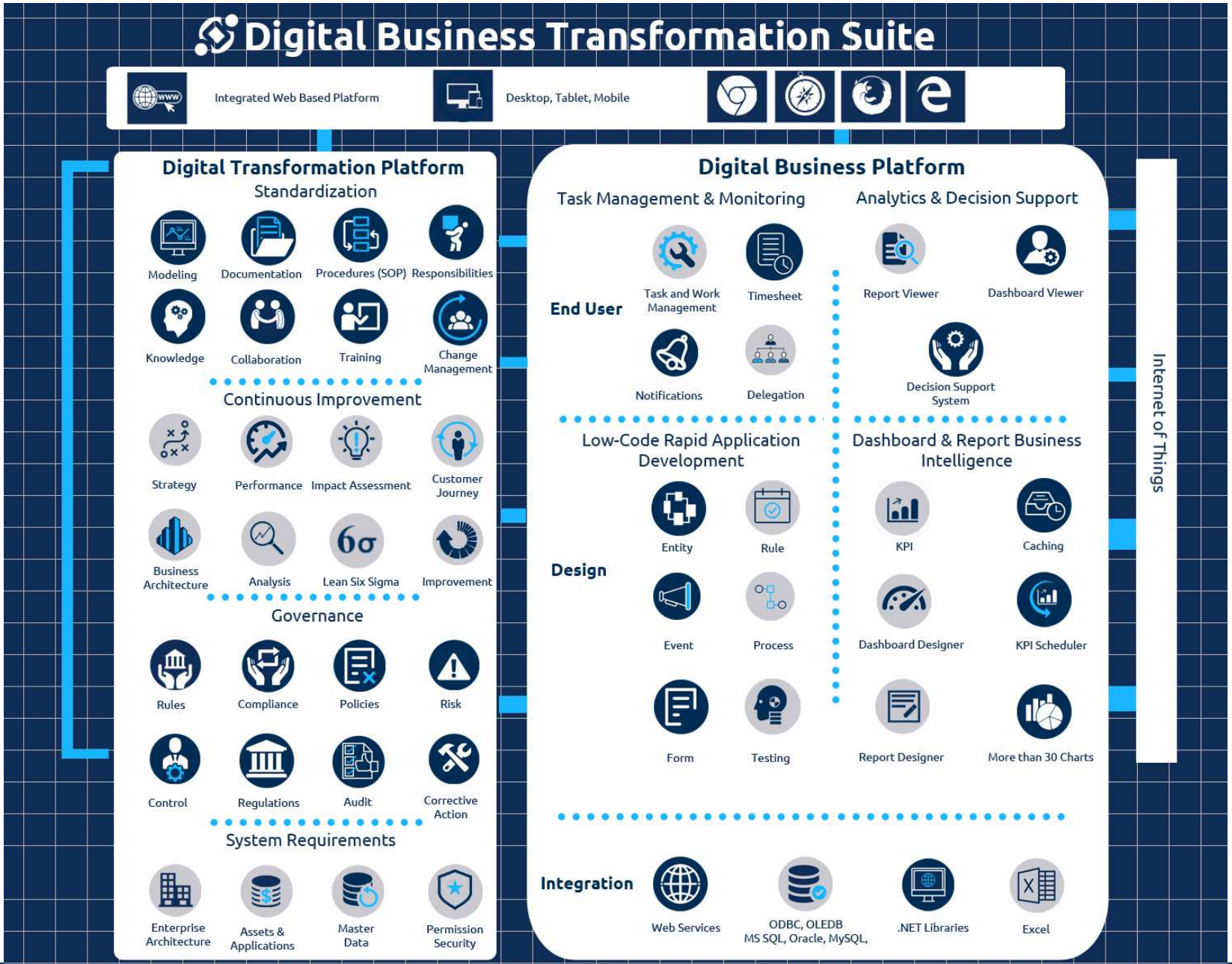
CIO APPLICATIONS TOP 25 GOVERNANCE, RISK & COMPLIANCE (GRC) TECHNOLOGY PROVIDERS - 2017

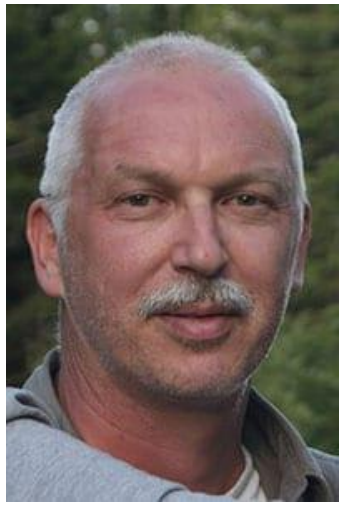
CIO Review 20 MOST PROMISING BPM SOLUTION PROVIDERS -2017

Gartner®
2017 Business Operating Systems

FROST & SULLIVAN
Product Leadership Award

KMWorld Trend-Setting PRODUCT OF 2017





Piet Knijnenburg

Author, Integration Architect

Piet Knijnenburg has worked for 35 years for the IT departments of very diverse organizations in government, manufacturing, insurance, banking, army and retail.

Operational Performance

IT Support of Operational Processes

- Direct support (primary processes)
  - Control or Execution of processes
  - Registration of processes
- Indirect support (secondary processes)
  - Business analytics
- Agility

Data Lifecycle Management

- Which entities, elements and attributes
- Relations
- Lifecycle & Ownership (of each)
  - Authoritative system
  - Role of participants
- Structure & Representation
- Consumers & Usage
- Rules & quality measures
- Policies & Incentives
- Compliance & Governance, Data Protection

Business Process Coverage

Engineering	Manufacturing	Sales	AfterSales	HR	Finance	BI	IT
Concept to Product	**Build to Completion**	**Order to Cash**	**Deliver to Support**	**Hire to Retire**	**Record to Report**	**Business Intelligence**	**IT Support**
Product Research	*Production Planning*	*Customer Management*	*Asset Management*	*Talent Selection & Acquisition*	*Accounts Payable*	*Operational reporting*	*Architecture*
Product Engineering	*Production Control*	*Dealer Management*	*Field Service Management*	*Time and Expenses*	*Accounts Receivable*	*Dashboards Scorecards*	*Analysis, Design & Build*
Product Lifecycle	*Plant Management*	*Sales Order Management*	*Claims Management*	*Develop & Retain Talent*	*Financial Reporting*	*Business Analytics*	*Operational Management*
						Management reporting	

Acme Corporation business model

A Collection of Applications? **or** **An Applications Network ?**

Difference between a collection of applications and a network of applications

Composition

Communication

Organization

Coordination

"Things Work Better When They Work Together on Purpose“

A system is as good as the cooperation of its parts

or

An application network is as good as the integration of its applications

Integration of applications

is at least as important as the applications themselves

Don’t focus on things, focus on how to make them work together.

Don’t Focus on Applications or Data Need

Focus on applications themselves leads to a wrong integration approach: getting data from one application to another.

- Wrong integration pattern choices, unsuitable integration solutions
- Loss of (context) information
- Consistency issues, error handling issues
- No reusability, many similar interfaces (design & runtime!)
- Poor performance, uneven workloads
- Unnecessary data streams
- Unclear, ambiguous or inconsistent meaning of data
- Poor business process support, outages

Cost of Unsuitable Integration Solutions

- Lack of Consolidation:
  - Unnecessary load (CPU, network)
  - Unnecessary maintenance
  - Unnecessary monitoring

- Incidents:
  - Often due to contingency, cause sometimes hard to find
  - Impact often much larger than infrastructure issues
  - Remedy often much more difficult

- Unprecise fit to business processes

Reasons of Unsuitable Integration Solutions

(re)use unfit existing legacy integration

"getting data from A to B" syndrome

design/build endpoints before service contracts

correct integration perceived as
- too difficult
- too expensive
- too time consuming

integration pattern not supported by application

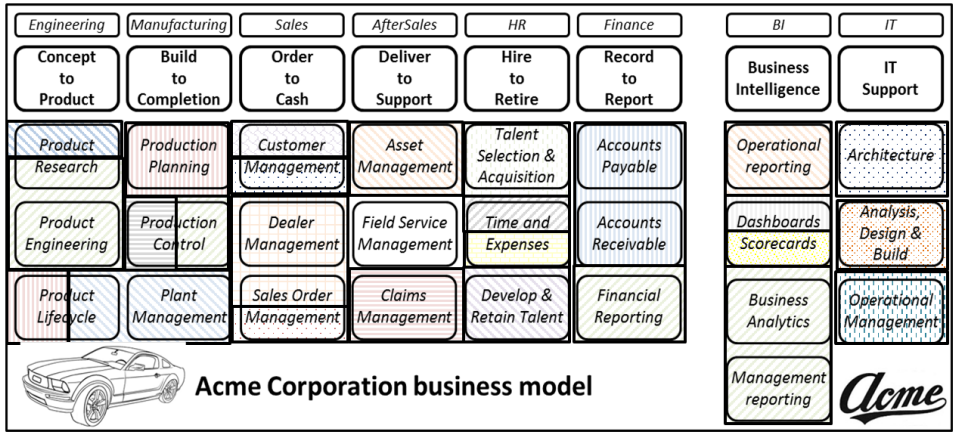
security requirements

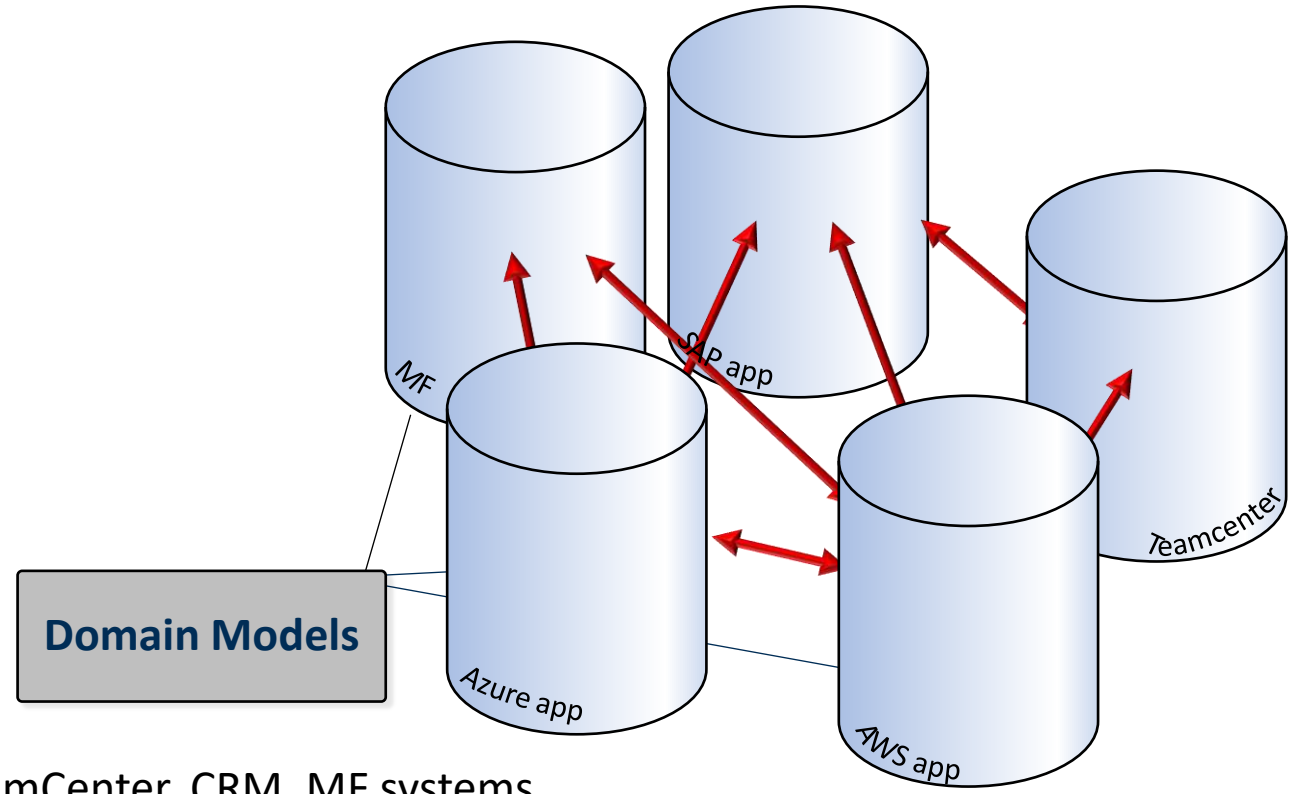
In an Applications Network . . .

Integration is **NOT** about getting data from one application to another. Instead, it is . . .

Integration is the support of your business processes across a heterogeneous landscape of applications

Collection of Applications



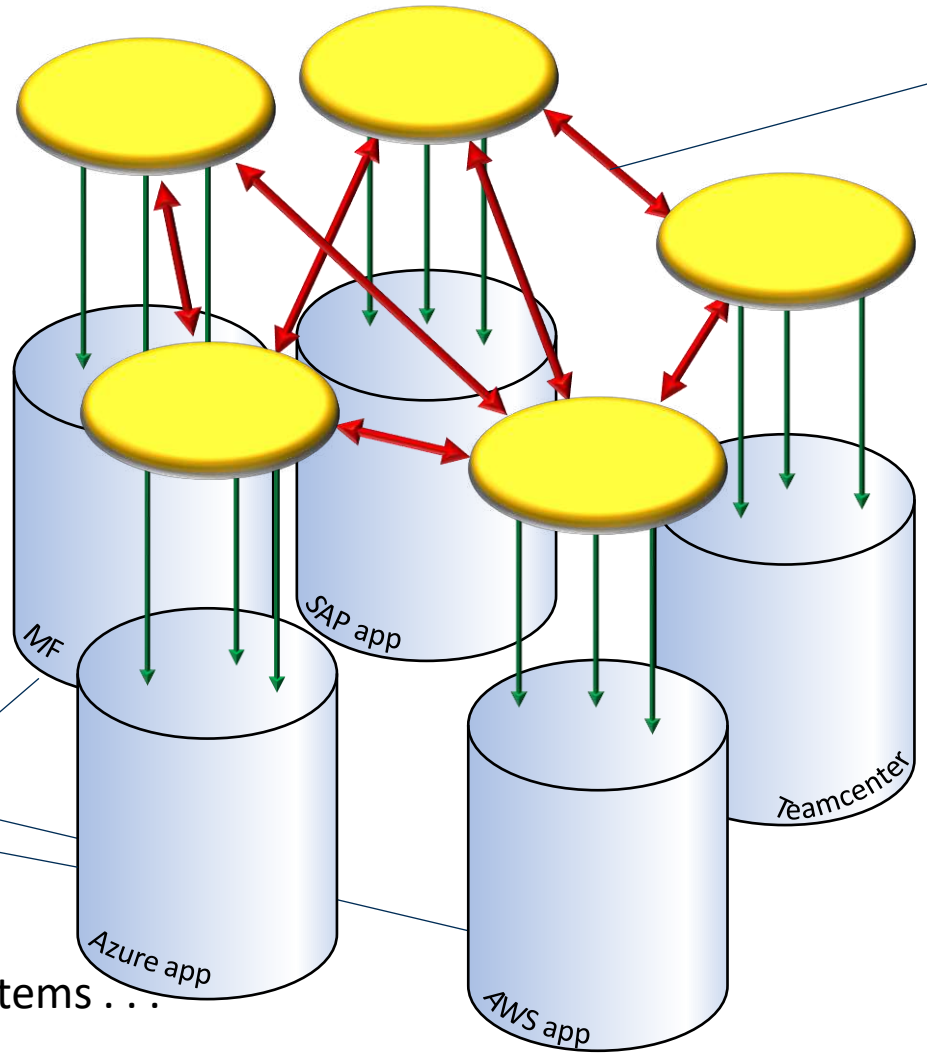


SAP, TeamCenter, CRM, MF systems . . .

Applications Network

App Network

App Adapters



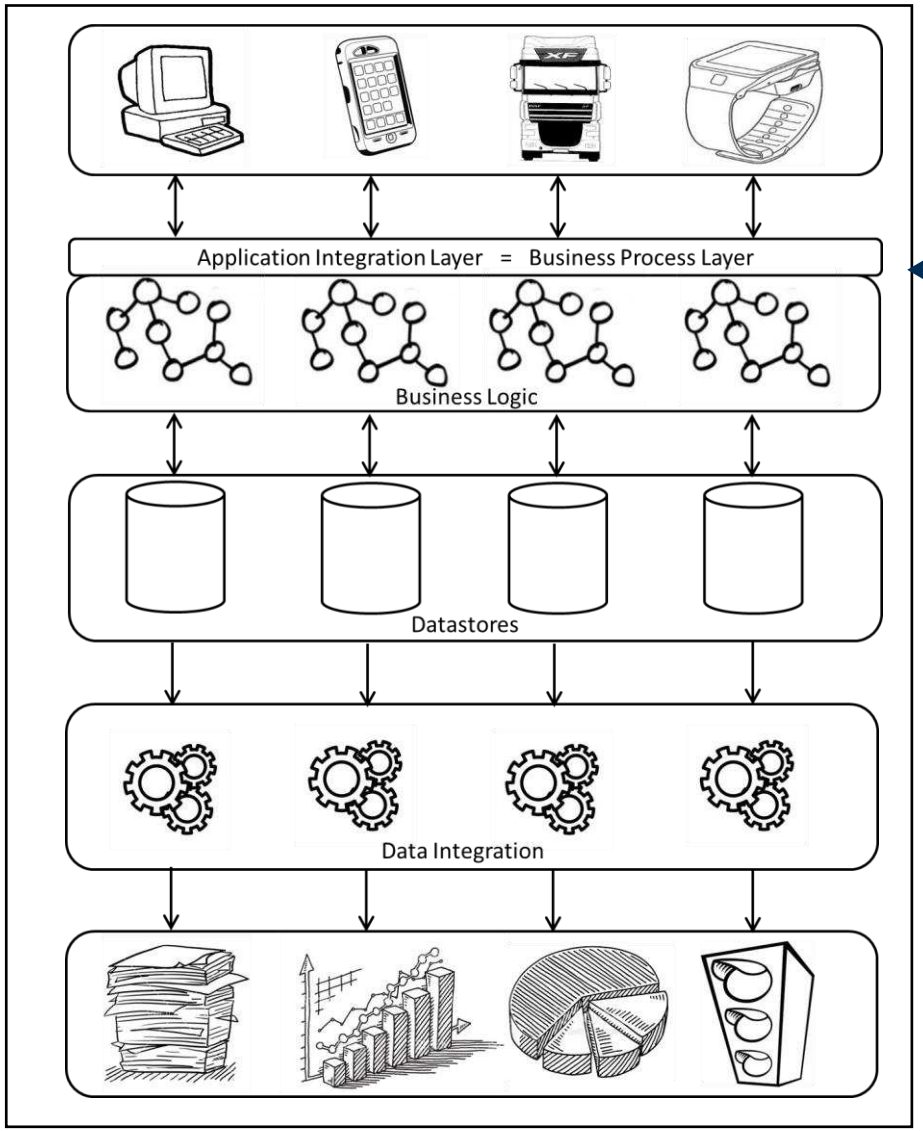
Enterprise Model

Domain Models

SAP, TeamCenter, CRM, MF systems . . .

Integrated Integration

Primary Business Processes



Process integration svc's
- Sharing process information between your processes

Data integration svc's
- Gathering data over your processes

Secondary Business Processes

Primary & secondary processes have **very** different non-functional requirements

Data Gathering Services are not Business Services

Data Gathering Services:

Disregard business process context

Miss out on business process context

Expose domain models across domain boundaries

OK for **non-primary** business processes, such as ETL for data warehousing / business intelligence

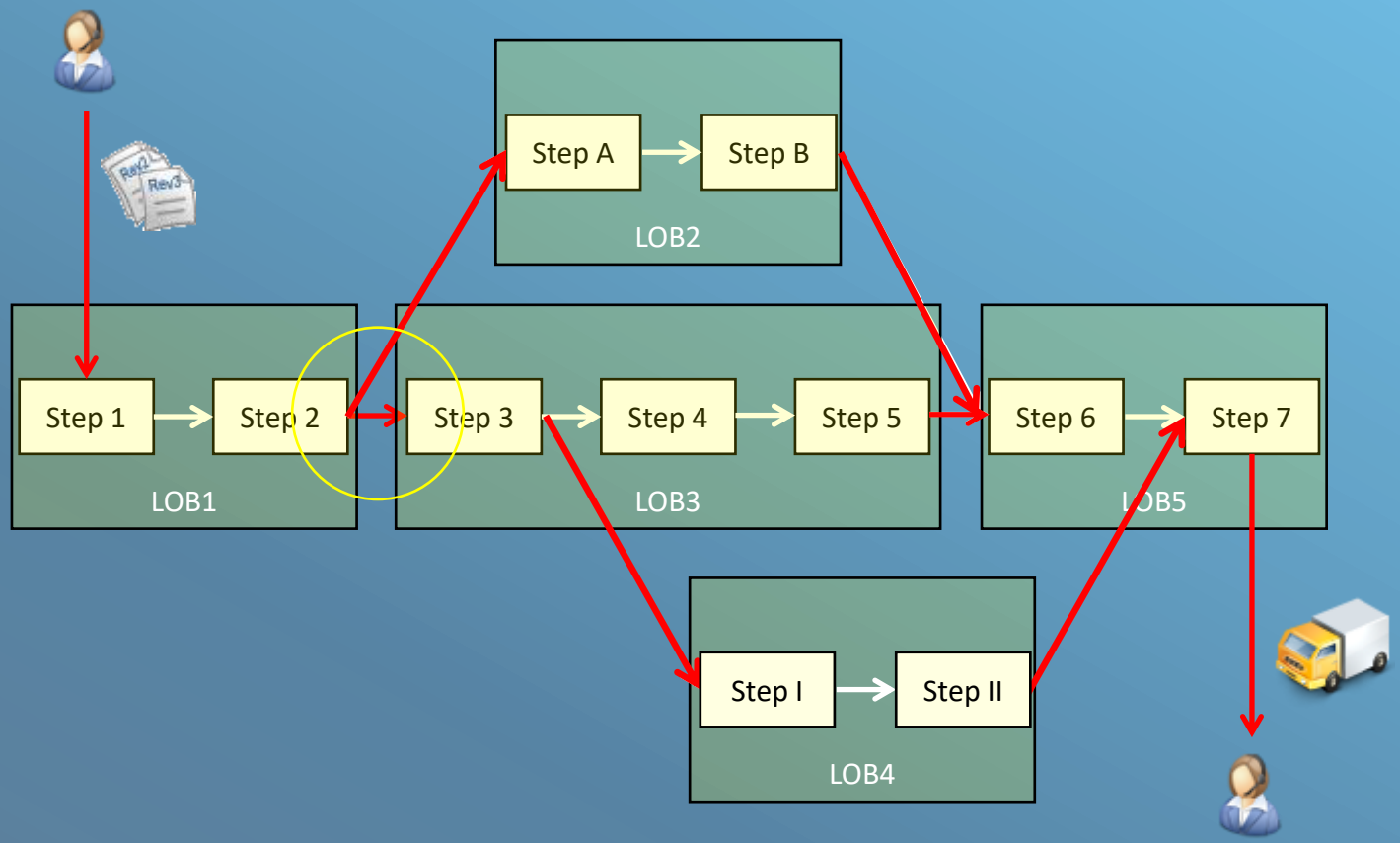
Dangerous for **primary, operational** business processes

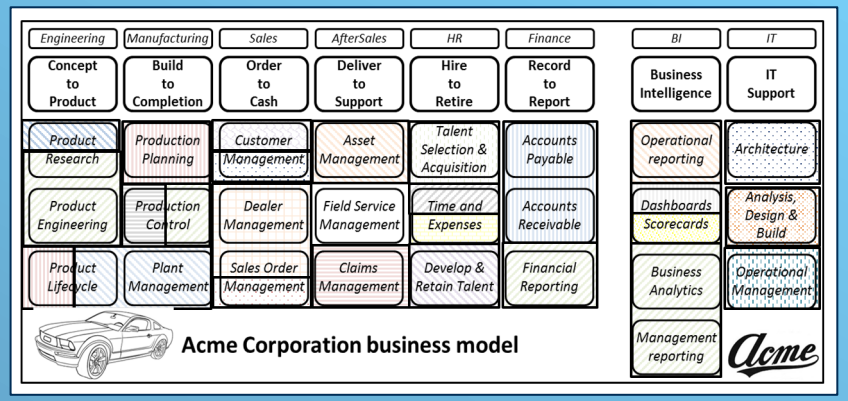
- data may be inconsistent, untimely, irrelevant, ambiguous

a.k.a. Snapshot “integration”

- Database replication
- Export & Import (full, delta)

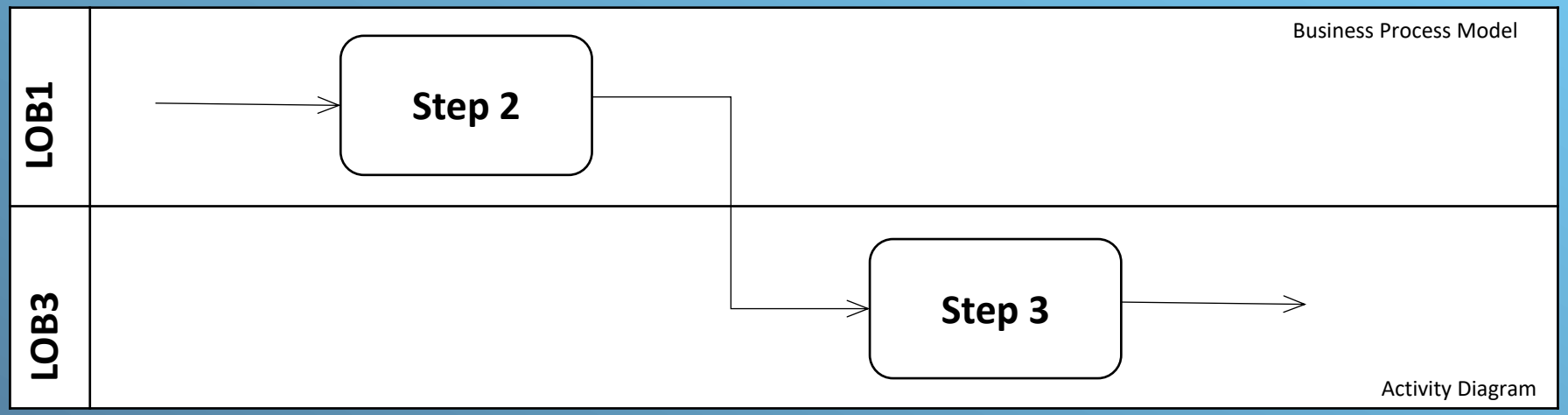
Applications Network and the Business Process



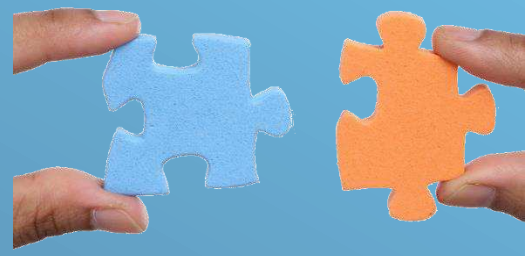


→ The fabric of an applications network consists of integrations that implement the business process information flow between the LOB systems.

Anatomy of an Interface



- Define the event(s) that trigger the information stream
- Define the information stream
- Choose the integration pattern
- Define the adapter and mapping at the endpoint of step 2
- Define the adapter and mapping at the entrypoint of step 3



Business Service Design: Required Information (1/2)

1. Business Process Context

- What is(are) the business event(s) / trigger(s) ?
- Which business process(es) is(are) involved?
- Where do these processes take place?
- How many / how often
- Roles, RACI, what roles are involved?
- No business trigger, no business service!

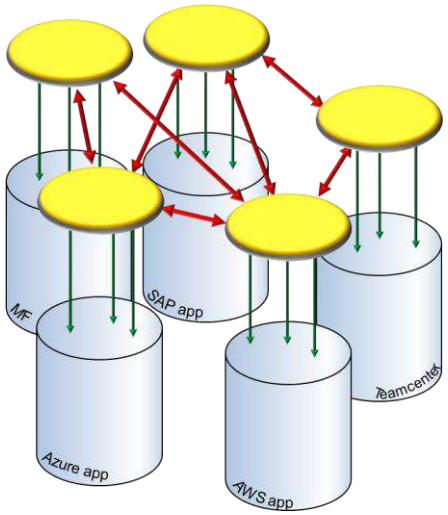
Business Service Design: Required Information (2/2)

2. Information Stream Context

- Which entities, elements and attributes
- Lifecycle & ownership (of each)
  - Authoritative system
  - Role of participants
- Structure & representation
- Validate data requirements against trigger

Canonical Schema

- Data Contract(s) (Canonical schema)
  - **NOT** a reconciliation of the involved endpoints
  - But a representation of the data involved in the business service (irrespective of IT implementation)
- **NOT** to be forced onto the domains
- Loose-coupling on data level



Datasource Roles in the Data Lifecycle

Who, what, when, where decides over the creation and disposal of an entity, element or attribute?

- Authoritative source (master)
- Local copy (slave)
- Delegated authoritative source (manages on behalf of slave)
- Local Authoritative source (middleman)
  - is a client of an external authoritative source
  - is an authoritative source to other local systems

Business Service Design: Message Nature

Derive message nature

- (OOP) Objects
- Commands
- Events
- Request messages
- Submit messages
- Acknowledgements
- Response messages
- Business Objects
- “Records”
- Composite records
- Error messages

Never exchange objects

- Allowed within a solution
- Not between solutions
- Use messages instead

Commands

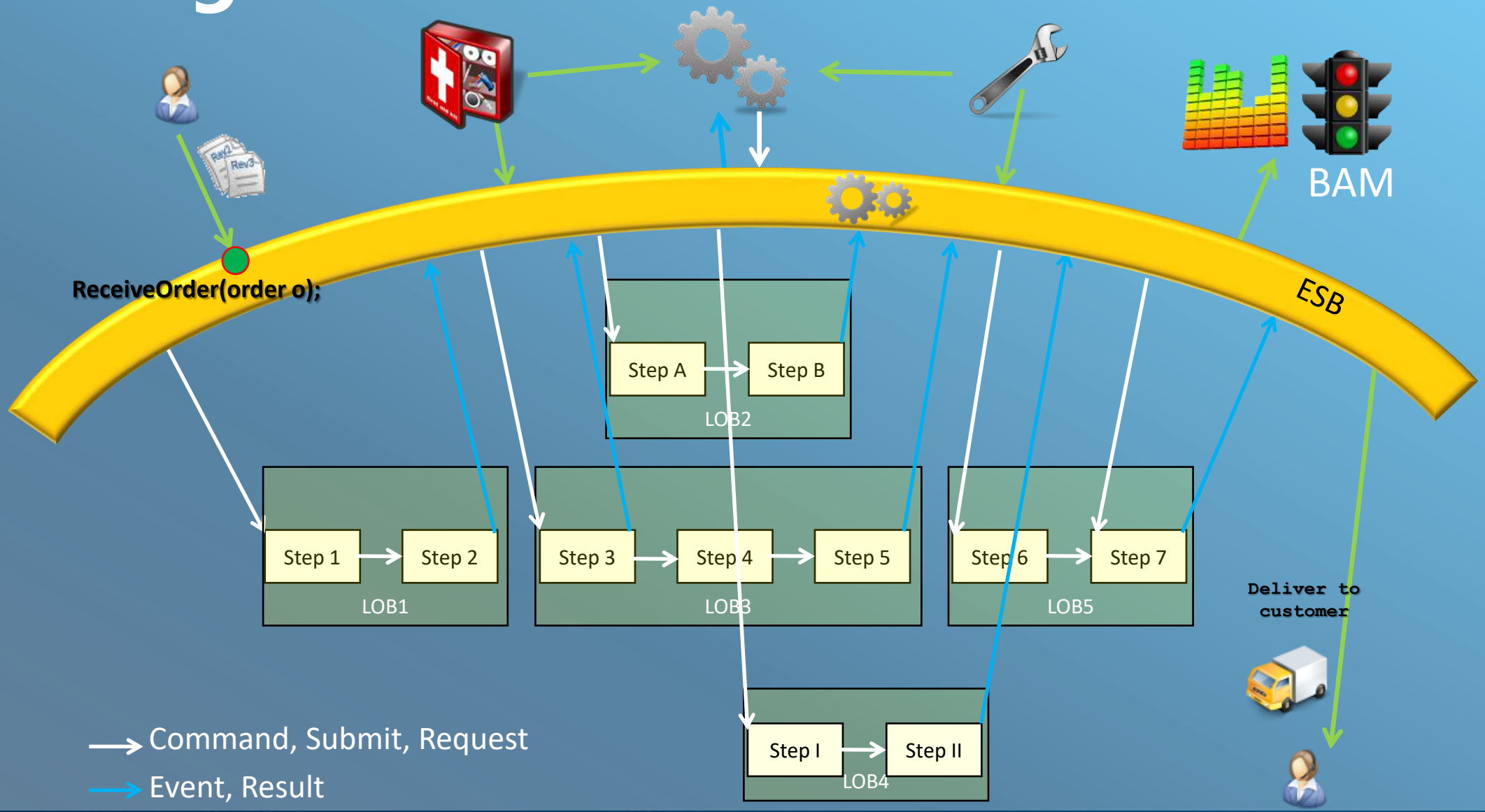
- Do not send commands (from applications)
  - It assumes that the sender knows the receiver
  - It assumes that the sender has authority over the receiver
  - It assumes that the sender must coordinate the workflow
  - It assumes that the sender has knowledge of the receiver
  - It assumes that the receiver knows the sender
  - It assumes that the receiver must obey the sender
  - Do not send commands (from applications)

- Leave sending of commands to controllers (i.e. workflow engine, service bus)

Events

- Instead send events
  - the sender does not need to know the receiver(s)
  - the sender has no authority over the receiver
  - coordination the workflow lies outside the sender
  - the sender needs no knowledge of the receiver
- Send only events on business processes that you are responsible for (only from authoritative systems)
- Name: noun + past participle verb (i.e ClaimApproved)
- A receiver of an event does **NOT** send a response
- TIP: Let a controller (Choreography / Orchestration / Workflow engine) receive events and send commands

Integration and the Business Process (EDA)



Request & Submit messages

- “Request” implies that a response is expected
- “Submit” implies that no (immediate) response is expected
- Send “Request” & “Submit” msgs to authoritative systems
- Authoritative system decides how to handle
- Name: imperative verb + noun + “Request” (i.e. RegisterClientRequest).

Acknowledge & Response messages

- Response implies a reference to a request
- Acknowledge message has no functional content
  - i.e. http status 200 or <SOAP-ENV:Body />
  - no more than a message delivery notification
  - **N O T** 100% reliable
  - usually implicit and embedded in protocol
  - synchronous ACK, asynchronous processing
- Response message has functional content
  - **not** necessarily synchronous !

Models, records & Composite Records

- For data synchronization services, not integration services
- Business objects (cmp objects) are objects (as recognized in the business process model) with their dependents and dependencies.
  - i.e. Invoice + invoicelines + 3rd party + items
- Records: the (partial) content of a table
- Composite records: collect required data from various tables and flatten into one container
  - Composite records are (mostly) point-to-point solutions

All Messages

- Make sure no knowledge of service partner is needed
- Clients either use proprietary format (let middleware perform transformations)
- or uses standardized messages
  - Standardization (i.e. XML dateTime, ISO, NEN, DIN etc…)
  - Qualification (i.e. UoM="pce")
  - Enumeration
  - Use an EDM (Enterprise Data Model) for "canonical messages"
- Make sure the message contains only information pertinent to its trigger or request
- Avoid "CRUD" names
  - i.e. use Register- instead of Create- or Update-
  - use Remove- instead of Delete-

Basic Scenario's Overview

Msg Type	From	To	MEP
Objects	use only internally		
Events	Auth.Sys.	Clients / Controller	FF / PS
Commands	Controller	any one executor	FF
Requests	Client	Auth.Sys.	RR / SR
Responses	Auth.Sys.	Client	RR / SR
Submits	Client	Auth.Sys.	FF / SR
Ack's	Auth.Sys.	Client	implicit
Business Objects	Auth.Sys.	BI Client	FF*
"Records"	Auth.Sys.	BI Client	FF*
Composite	Auth.Sys.	BI Client	FF*

FF*: also MFT

Some Simple Rules of Thumb

- If the data contains an element or attribute called “status”, you are probably engaging in snapshot integration. Use (normalized) events instead.
- Never use an operation code to steer the inner workings of a service. Normalize!
- If neither partner in a desired integration solution is authoritative, question the validity of the sln.
- Avoid sequence dependency.
- Avoid stateful services.

Further reading

Piet Knijnenburg

pietknijnenburg@gmail.com

www.interfacing.com

sales@interfacing.com

+1 (800) 561 5005