Five Key Points About Business Process Change –
Avoiding the Common Pitfalls

An executive briefing presented by
Adept Events and Clariteq Systems Consulting
for University Medical Center Groningen

Alec Sharp
Senior Consultant
Clariteq Systems Consulting Ltd.
West Vancouver, BC, Canada
asharp@clariteq.com

umcg

///AdeptEvents

www.adeptevents.nl

CLARITEQ

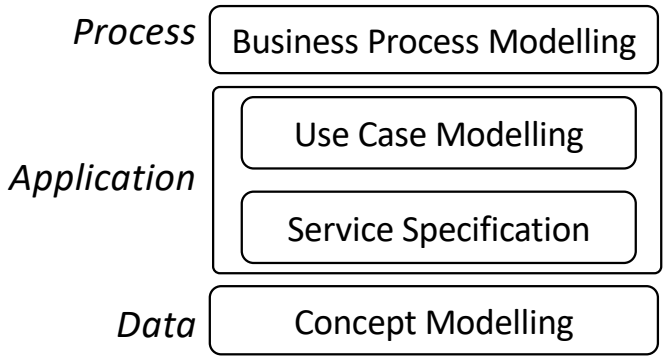
www.clariteq.com

Instructor / course developer background…

Alec Sharp, Clariteq Systems Consulting – *asharp@clariteq.com*

- 40+ years experience as an independent consultant:
  - Business Process Change – discover, model, analyse, and design/redesign processes
  - Application Requirements Specification
  - Data Modelling and Management

    +
  - Facilitation & Organisational Change
  - Project Recovery

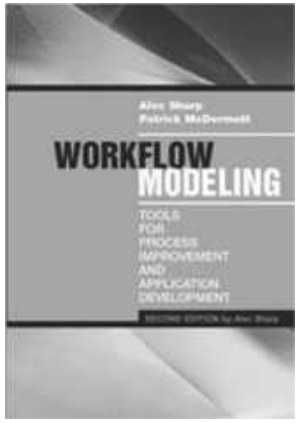


- Consulting, teaching, speaking globally (pre-pandemic)

- Awarded DAMA's global Professional Achievement Award for contributions to "human-friendly" data modelling

Check out the nice reviews on Amazon - http://amzn.to/dHun1o

- Author of "Workflow Modeling"
  - best-selling book on process modelling & improvement
  - second edition – a complete re-write



Small, husband & wife company, global clients

ABB (ASEA Brown Boveri)
Aflac
American Honda
AMP (Australia Mutual Provident)
BackOffice Associates
Bank of Finland
Bellrock
Brisbane City Council (Australia)
Canadian Natural Resources Ltd.
City of Seattle
Civica UK
Clearwater Paper
Corvias
Dell
DHL Express
Dutch National Bank
Ericsson
Essity
Eurojust (European Justice Comm.)
European Central Bank
Fortum
Helse Vest - Norway
HM Land Registry - UK
Home Depot
Idaho Transportation Dept.
Intel
ISO New England
ING Bank
JP Morgan
Kal Tire
KONE
LGM Financial Services
Liberty Mutual
Livestock Improvement Corp.
MacDonald Dettwiler
Manitoba Public Insurance
Marathon Pipe Line
Microsoft
Ministry of Defence - UK
Ministry of the Interior - Slovakia
MTS Allstream
Nexen
Novo Nordisk
Nusenda Credit Union
OP Bank
Partner Reinsurance
Ritchie Brothers
Phillip Morris
Roche Diagnostics
Salt River Project
Saudi Aramco
Serco
Shell
Sparta Consulting
State Street Bank
SunGard
Synechron
Sysdoc
Talent Base
Teck
The MUSIC Group
The Seattle Times
UK Government
University Med Ctr Groningen
Washington Gas & Light

– Higher Education –
Carnegie Mellon University
Cornell University
Douglas College
Gonzaga University
Humboldt State University
The Jackson Laboratory
The Ohio State University
Portland State University
Salt Lake Community College
Southern NH University
University of Arkansas
University of British Columbia
University of the Fraser Valley
University of Maryland
University of Utah
University of Washington
Utah Valley University

Five central ideas

1. It's essential to have clarity on what a *business process* really is

2. Existing performance measures are often *functionally aligned* and work *against* business processes

3. Enterprise system implementations *must* include a *business process* perspective

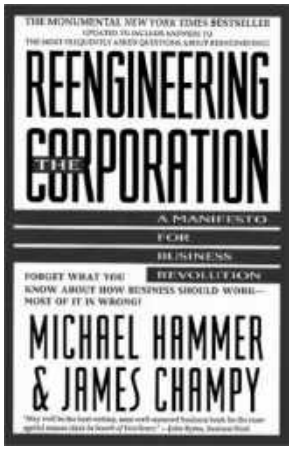
4. Success with business processes depends on taking a *holistic view* in which six *enablers* are considered

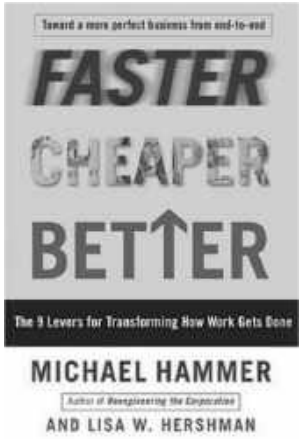
5. Business processes can't be great at everything – a single *differentiator* or *strategic discipline* should be chosen

1. Confusion – what **is** a "business process?"

1. It is essential to have clarity on what a *business process* really is
2. Performance measures may be *functionally aligned* - work *against* business processes
3. Enterprise system implementations must include a business process perspective
4. Success with business processes requires a *holistic view* in which *six enablers* are considered
5. A business process can't be great at everything – a single *differentiator* must be chosen

In the early 1990s, Michael Hammer popularised the focus on *business process*





Introduced core terminology:

- end-to-end, cross-functional, functional silo, ...
- even *business process*

Still, people and organisations miss the point...

Lesson #1 – Never assume everyone agrees what a "process" is

We need some help with our *Product Lifecycle Management* process.

Not a single process – it's a *family* of multiple business processes (a *process area* or *process domain*)

I spend all day writing business processes, like the process to *Revise Product Brochure Image.*

Not an entire process – it's a *procedure* providing instructions for a single task (SWI – standard work instructions)

A whole *spectrum* of interpretations of *process.*

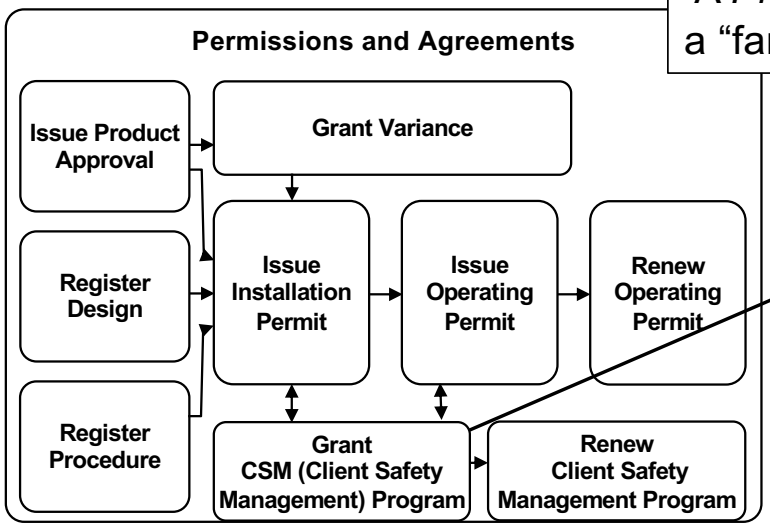
Seek balance – a "business process" lies between the extremes

Most people hear *process* and think *procedure!*

The key issues – granularity and orientation

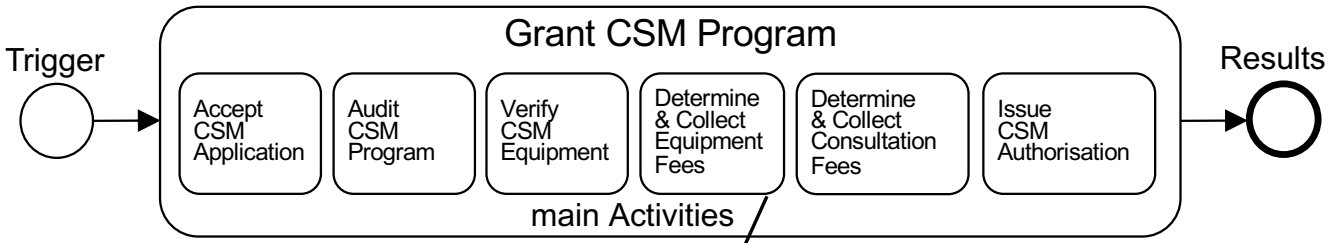
A collection of processes vs. a process vs. a procedure

A *Process Area* or *Process Domain* –
a "family" of related Business Processes:



An *end-to-end process* – "Grant CSM Program,"
from *application* to *authorisation,*
involving many departments, external organisations,
participants, and procedures.

Business Process Scope Model (TRAC) – pure "what"…



Business Process:
A sequence or set of *activities*
that delivers significant *results*
for the process' customer
and other *stakeholders*

"how" to complete a task…

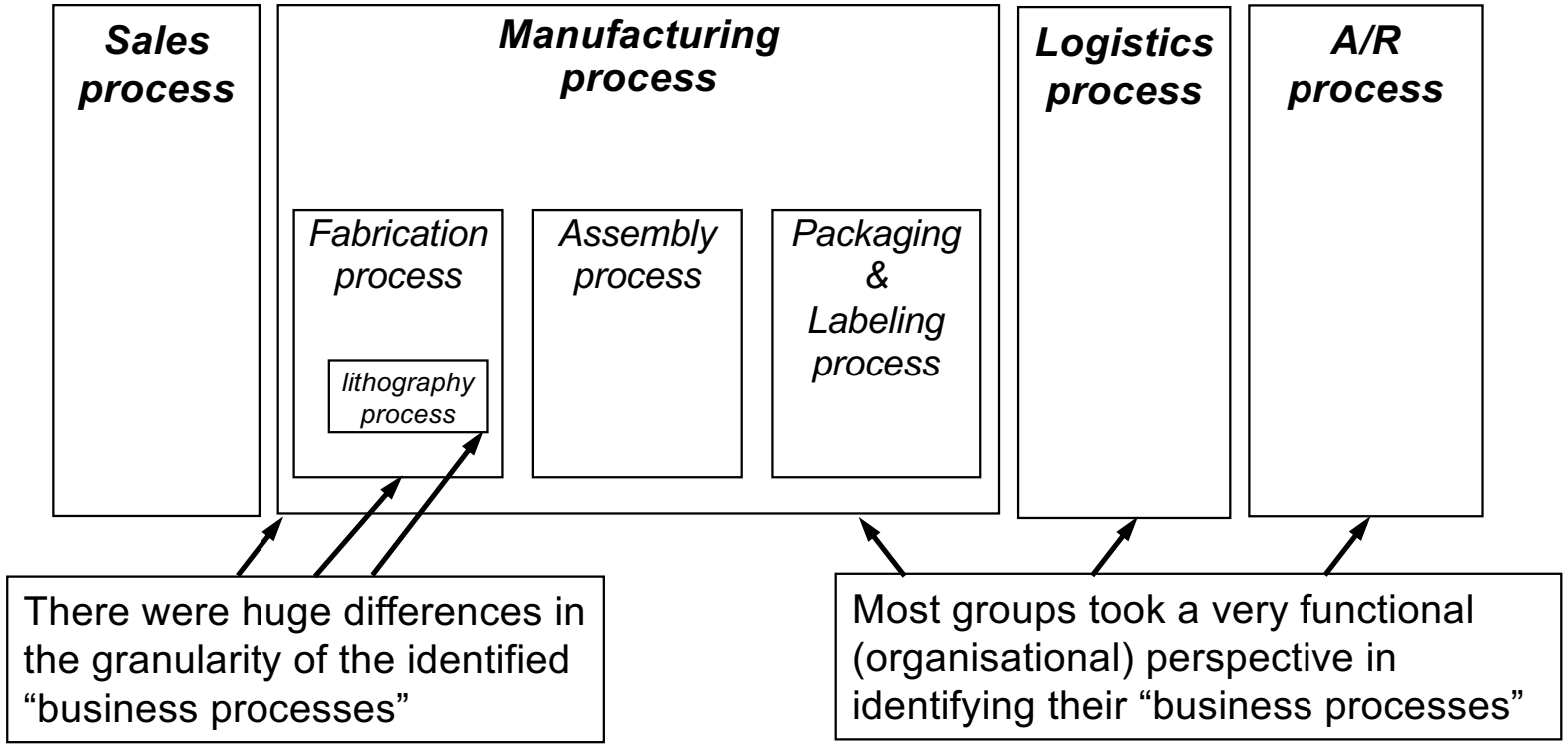
Procedure:
A set of step-by-step work instructions
(a job aid) for a specific task or activity
that will yield identical results every time

Procedure – Calculate Unit Registration Fees:
For each Unit:

- Determine Unit Type and Unit Risk Factor;
- Apply Registration Fee from Reg. Fee Table;
- Identify additional Inspection fees from…

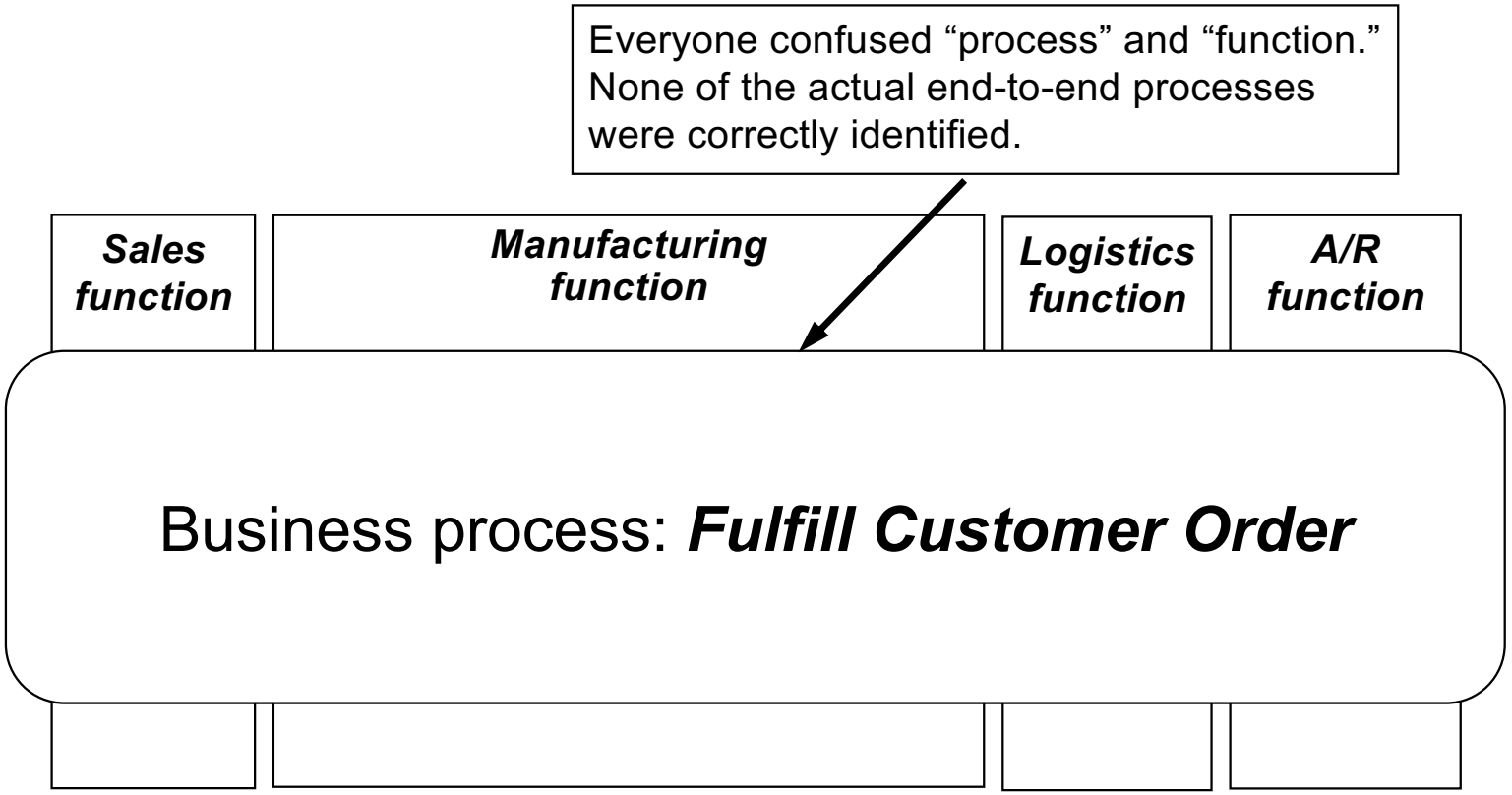
A real life (and expensive!) example

As part of a massive system implementation, a global manufacturer identified the *business processes* that were expected to improve:



The problem? *These aren't processes – they're functions!*

The "real" business processes were missed



"Business Process" =
end-to-end, cross-functional, business process.
"Larger" than people think – from *initial* trigger to *final* results.

What are the boundaries of the process?

Trigger

Order received? *No.*

Before that…

- Contract is Finalised
- Price & Schedule are Negotiated
- Specifications are Confirmed

And before that…

- Demand is Signalled. *Yes.*

Result

Order is Shipped? *No.*

Order is Received? *No.*

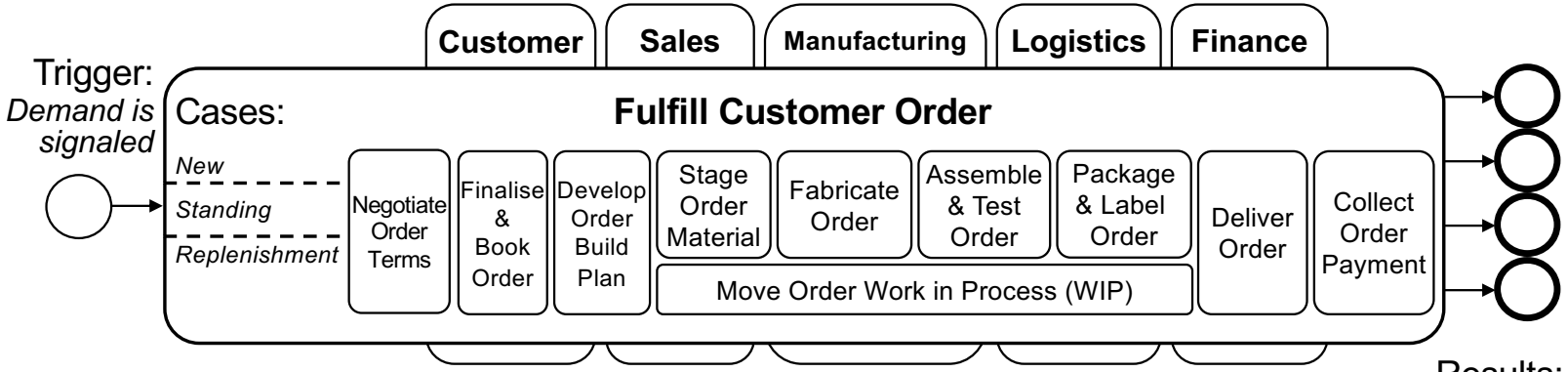
Order is Received, Tested, and Accepted? *Yes.*

Any other results?
Yes, for other stakeholders.

Always trace to the *earliest* trigger,
and to the *final* results for *each* stakeholder.

Process Scope Model – "what" first, "who and how" later

I build a *Process Scope Model* & a *Process Summary Chart* on ~100% of *Project Recovery* assignments -



"TRAC" –

*1 – **T**riggering event or events*

*2 – **R**esults: final outputs*

- result(s) received by the process' primary customer
- result(s) for other stakeholders (performers, owner, supplier, regulator, …)

*3 – **A**ctivities: 7 +/- 2 phases, milestones, or sub-processes*

- a phase achieves a significant intermediate result
- simply ask the participants for ~5 to 7 milestones within the process

*4 – **C**ases*

- main variations, e.g. "*new* order" vs. "*standing* order"
- verb – *qualifier* – noun

essence** of the process **("what")

5 – Functions or Organisation Units

6 – Actors and responsibilities

7 – Systems, data sources, other mechanisms

***as-is elements** of the process, for clarification **("who and how")** (6 and 7 not shown)*

Results:

Customer:
Goods received, tested, & accepted

Owner:
Payment received

Performer:
Commission credited

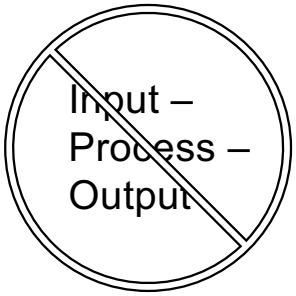
Industry Association:
Order stats reported

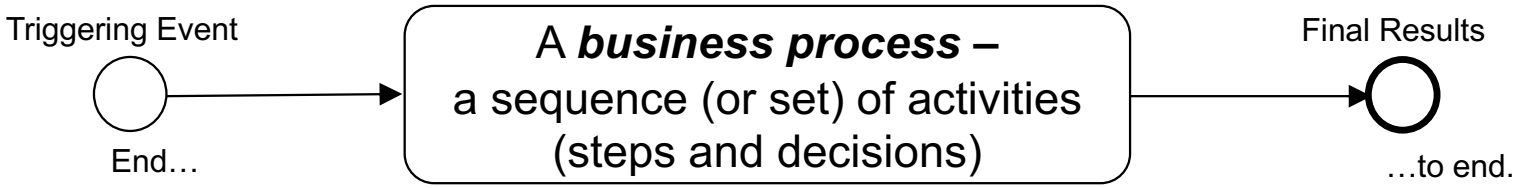
Always construct a *Process Scope Model* & a *Process Summary Chart* before diving into Workflow Modelling / Swimlane Diagramming

The essential framework

Business Process:

- a sequence (or set) of *activities* (steps and decisions,)
- initiated in response to a *triggering event*,
- that achieves a defined *result* for each process stakeholder





- Three types of events:
  - Decision-based (action)
  - Time-based (temporal)
  - Data-based (conditional)
- The *earliest* triggering event

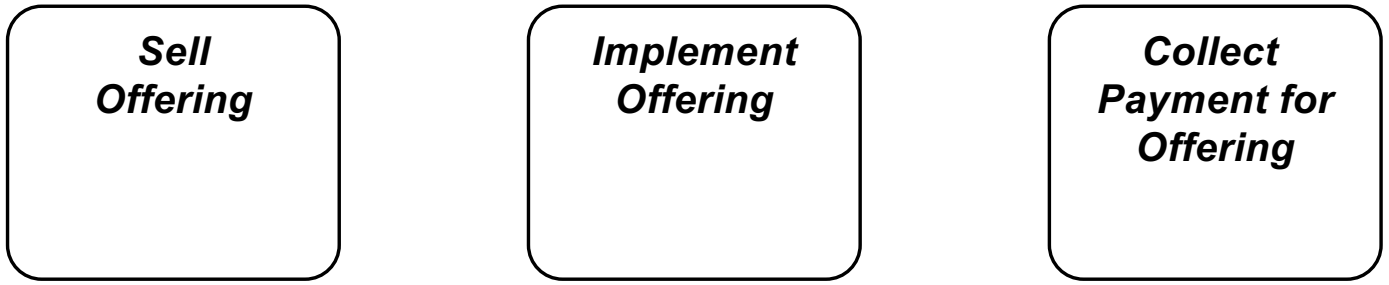
- Important processes are virtually always cross-functional and involve multiple actors / roles
- May be a defined *sequence*, or a more ad hoc *set* of activities
- First, identify *"what"* it includes – Trigger, Results, Activities, Cases ("TRAC")
- Later, we add *"who and how,"* then map the process flow, if there is one

- Three types of results:
  - A service
  - A good
  - Information
- The *final* result

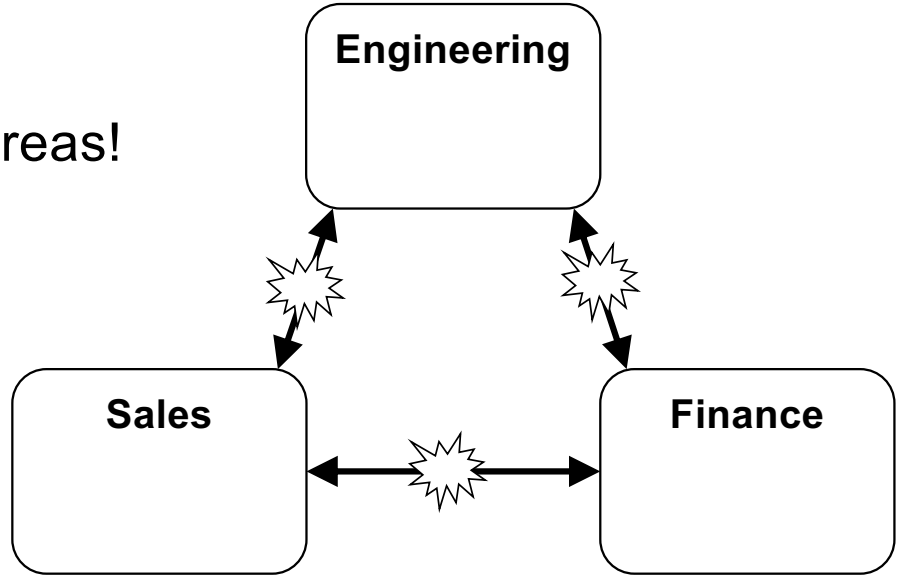
"What" before diving into the "who and how"

A recent Business Process example

A regional telecommunications provider (the "Telco") thought they had three main Business Processes, and efforts to improve them were failing:

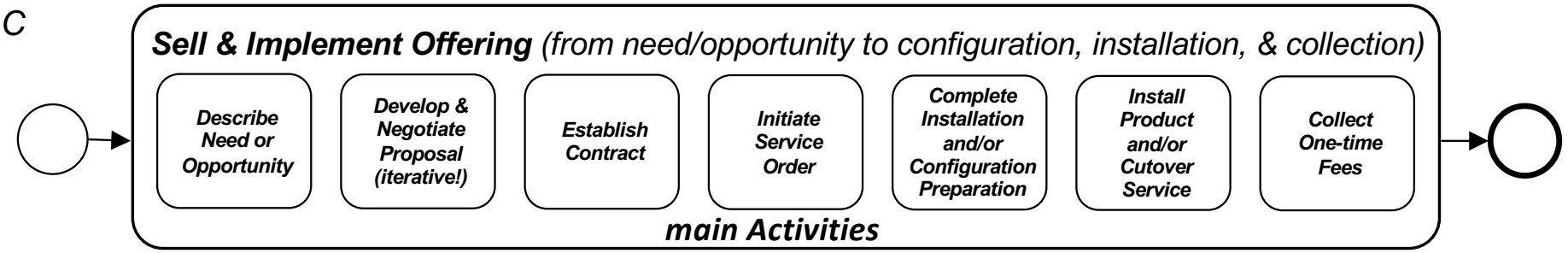


The outcome…
conflict between functional areas!



Process Scope Model showed ONE process not THREE

TRAC



Triggering Event:
- *Prospect / Customer* expresses *need*
- *Telco (Inside Sales, Marketing, Sales Rep, …)* recognizes *opportunity*

Cases:
- BU with or without Telco Internet, no cabling *(our focus)*
- initial installation
- service only
- product only
- mixed

Other factors:
- TBD

Results:
Customer:
Product / Service is *installed and operational* per original or amended contract terms
Telco:
- Ongoing source of *revenue* in place
- One-time *fees* collected

Employee:
- *Commission* or *referral credit*

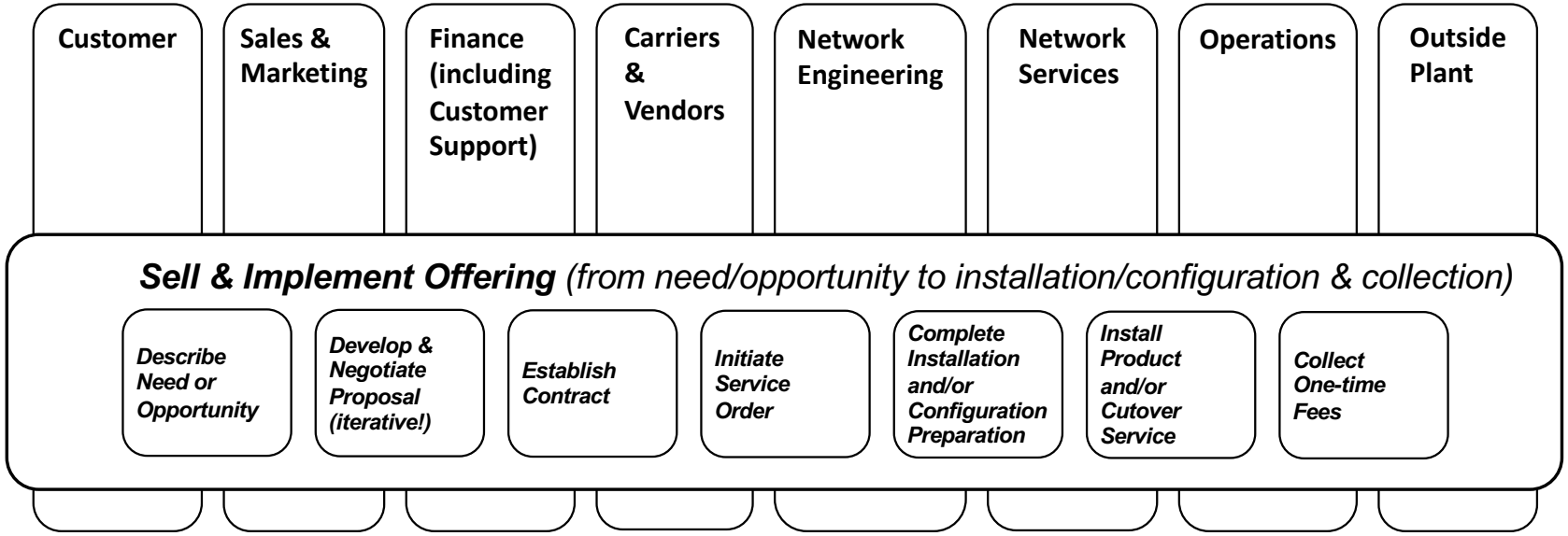
Agent:
- *Commission*

The "token," a Service Order, is changing state from *need/opportunity* to *configured, installed, & collected*.

The Business Process could be named "Fulfill Service Order" but the client wanted to name it "Sell & Implement Offering."

"We're all in this together!"
An end-to-end, cross-functional Business Process is a great lens to view organisation conflict and disfunction!

Process Summary Chart – my favourite diagram!



Process Summary Chart (a.k.a. ”Process vs. Function Chart”) adds “who” at the organisational unit or functional level.

Nothing else clarifies "Process" vs. "Function/Organisation" as well.

Great for putting details of Activities or Functions in context, e.g. …

Multiple roles by organisation for "Sell & Implement Offering"

Customer

Roles:

- Office manager or Owner (Smaller)
- IT (Larger)
- C-level (CIO, COO, CFO…)
- Third party IT vendor or agent
- Customer Project Coord.

Sales & Marketing

Roles:

- Senior. Account Execs
- Strategic Rel'nship Managers
- Account Rep 1
- Inside Sales Rep

Finance (including Customer Support)

Roles:

- Sales Admin
- Order Writer
- Billing Rep.
- Customer Support Rep.
- Director of Customer Support
- Receiving and Posting Payments (what role does this?)

Carriers & Vendors

Roles:

- Port Out Specialist (for CS Record) CSR/LSR
- IT Person
- Local government
- "Call before you dig"
- Customer Project Co-ord (int/ext consultants or phone vendors)

Network Engineering

Roles:

- System Admins (assign IP)

Network Services

Roles:

- BU Tech (survey)
- Switching Specialist (NS Spec)
- Network Services Coord / Provisioner

Operations

Roles:

- Sales Engineer
- CLEC Technician
- Material Manager
- Materials Specialist
- Project Manager
- Customer Training & Support
- Install Supervisor

Outside Plant

Roles:

- Drop Crew
- Lineman (not usually)
- Engineering Supervisor
- Outside Records Specialist

It was a shock to senior leadership to see how many roles were involved, often overlapping or unnecessarily

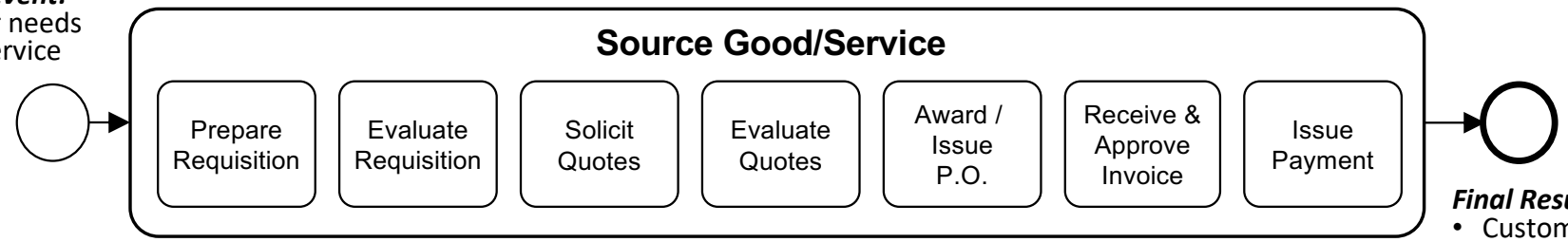
Another fast Augmented Scope Model example

Cases:
- $5000 - $25000 Goods
- $25000 - $50000 Goods
- $5000 - $25000 Services
- $25000 - $50000 Services

Assume everything <$5000 is purchased with a PCard

This example adds detail by major Activity (or subprocess/phase/milestone)

Triggering Event:
- Customer needs Good / Service



Final Results:
- Customer has received Good/Service:
- Vendor has been paid
  - via A/P
  - via PCard

Prepare Requisition

Develop scope of work / specs

Investigate potential vendors (and price?)

Solicit vendor quotes (just to get an idea)

Obtain approval (Department)

Verify Item and Account (General Accounting)

Submit requisition (visible to all)

Evaluate Requisition

Confirm completeness – get clarification this is actionable (scope sufficient)

Assign (or re-assign Buyer as necessary)

Identify MBE/SB opportunity
(competitive)
(co-op)
* sole source or co-op, vendor(s) known

Determine methodology
- sole source
- co-operative (piggyback on contract)
- competitive
- emergency

Solicit Quotes

Determine (additional) potential vendors

Solicit quote (including Bid Due Date)

Post quote (solicitation documents) in "the binder"

Resolve vendor queries

* Up to $200K, we control who gets solicitations; above, no control – it's "publicly advertised."

Over $200K there would be 20 more activities, and could be multiple award.

Evaluate Quotes

Receive quote (mail, fax, e-mail, …)

Confirm completeness

Verify suitable price, terms, and conditions (generally, low bid for equivalent)

Clarify (not negotiate) with vendor

Optional:
- Evaluate equivalency (for alternate)
- Confirm equivalency w. Customer

Identify vendor

Award / Issue P.O.

Generate Purchase Order

Notify Requestor

"Transmit / deliver" P.O.
* Pain point – we aren't sure when the vendor receives the P.O.

Receive & Approve Invoice

Receive Good/Service
* Invoice could be attached

Accept Good/Service

Issue invoice (vendor)

Issue Payment

Receive invoice:
- from vendor
- from the department the vendor sent it to

* Vendor complains invoice is "lost"

If >$5000, match
- invoice
- PO
- receiver

If <$5000, match
- invoice
- PO

* Could invoice $4K on $40K PO

Batch invoices for GAD

Issue Payment (Magic Happens Here)

Receive payment

* If multiple line items, different line items could go to different vendors;
* If multiple vendors, line items are not split.

Naming conventions will make life easier

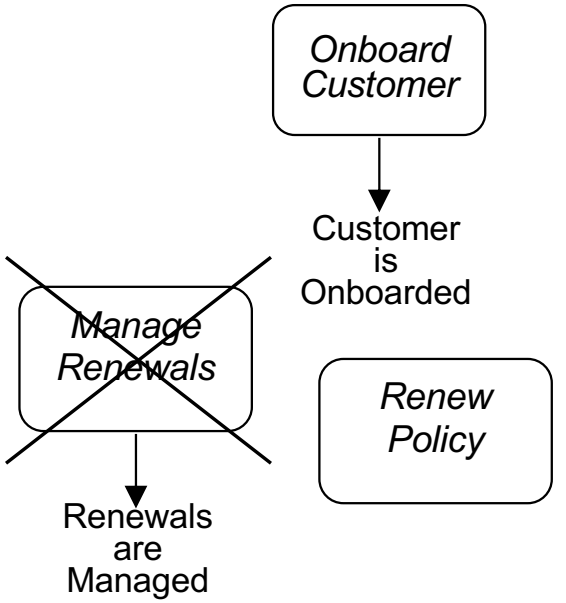
1. The process name ***must*** indicate the expected result
   - Name potential process in "verb – noun" format
   - Restate that name as a result ("noun is verbed")
   - Ensure this is the intended result of the process: *discrete*, so results are *identifiable* & *countable*

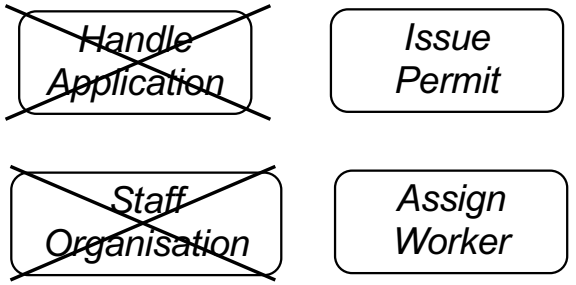
   - ***No mushy verbs:*** manage, monitor, administer, handle, track, support, maintain, etc.
   - ***Active verbs only:*** *Evaluate* Prospect, *Onboard* Customer, *Fill* Customer Order, *Resolve* Customer Issue, …

   - Applies to business processes, phases (subprocesses,) activities, steps, …

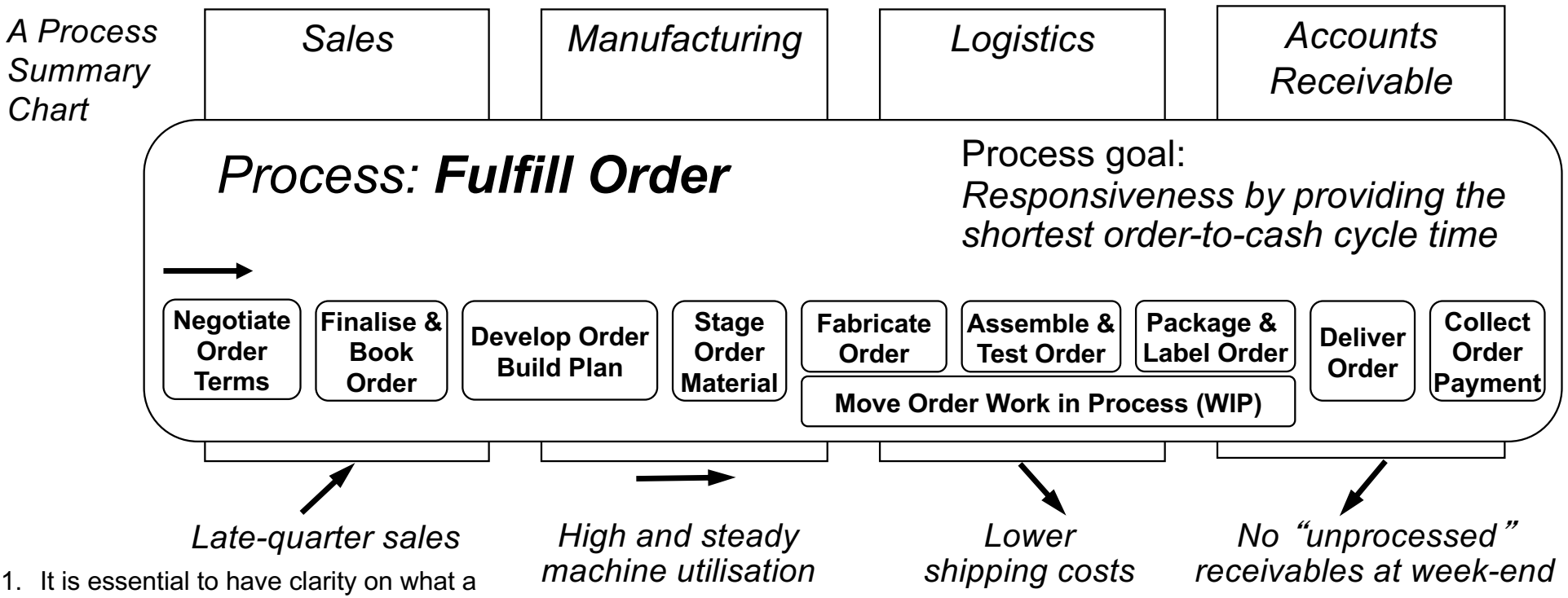
2. Name process from customer's perspective (what do they want from the process?)

3. Name process in the singular





2. A common obstacle – misaligned performance measures

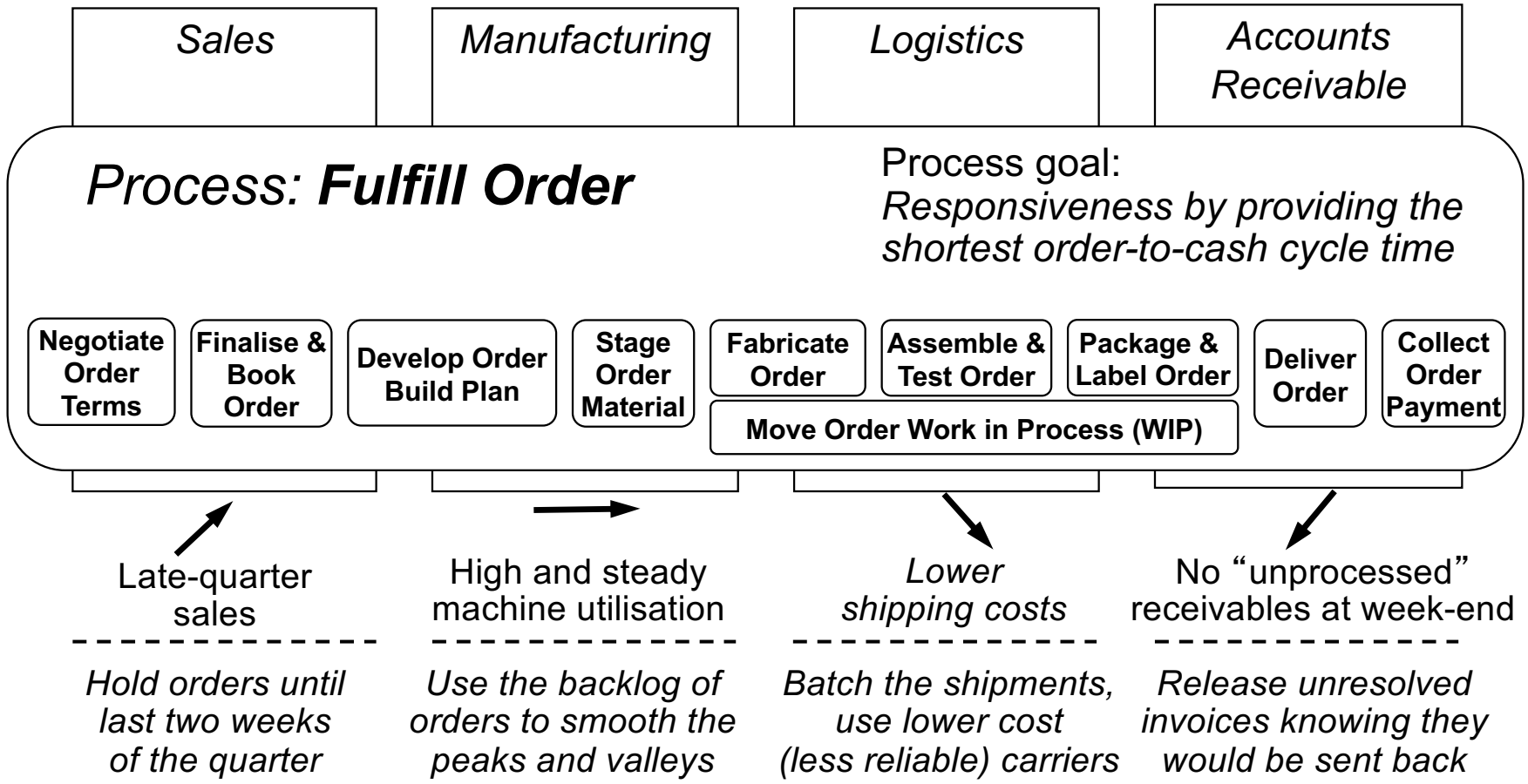


1. It is essential to have clarity on what a *business process* really is
2. Performance measures may be *functionally aligned* and work *against* business processes
3. Enterprise system implementations must include a business process perspective
4. Success with business processes requires a *holistic view* in which six *enablers* are considered
5. A business process can't be great at everything – a single *differentiator* must be chosen

But… performance measures were established *functionally, before* awareness of the *end-to-end process*

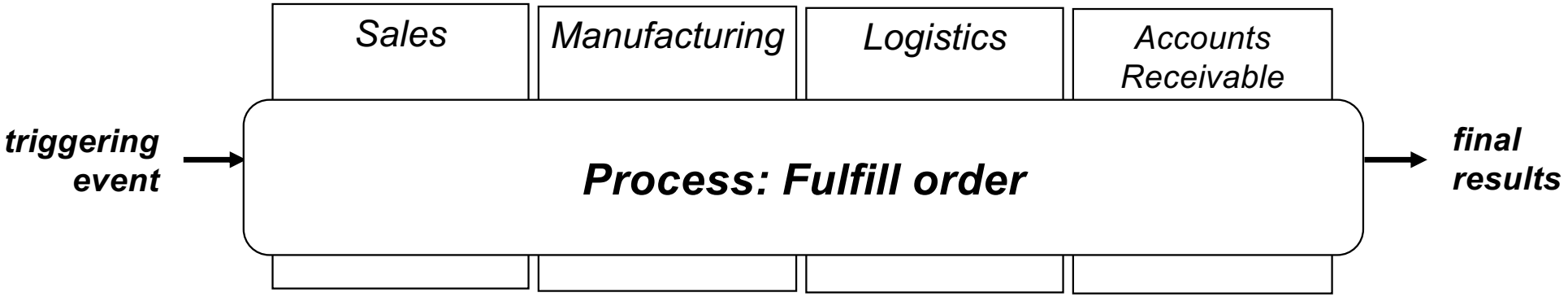
Discuss –
What are the likely impacts of these performance goals?
What will the different functions do to meet the targets?

Misaligned performance measures



Poor performance because each function was
working hard to meet *uncoordinated, functional targets*

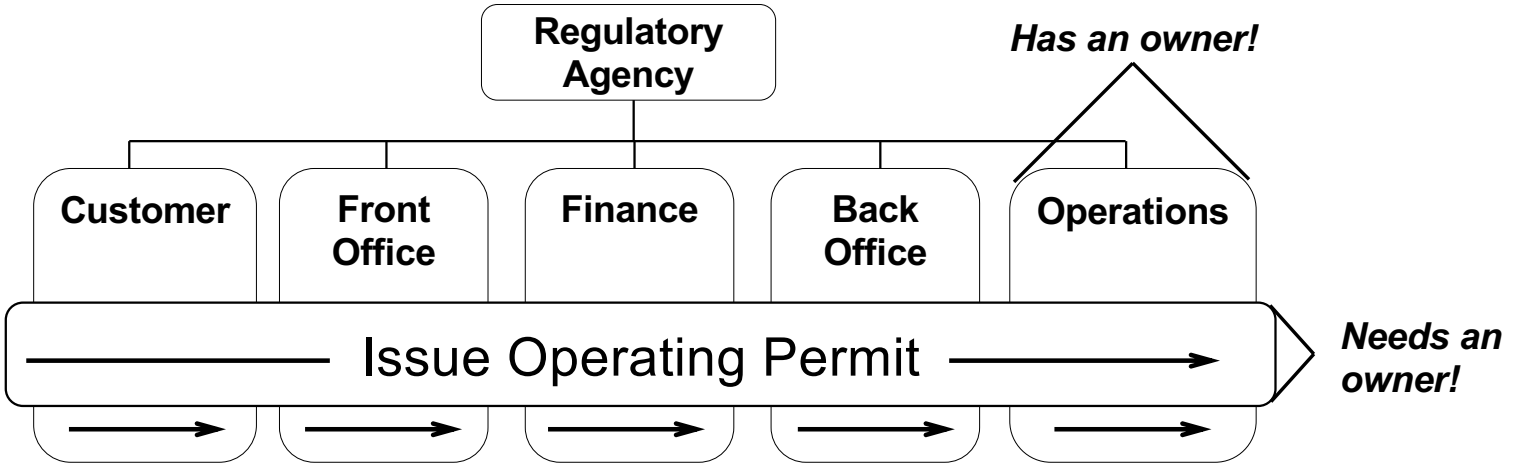
This doesn't mean functions are bad!



Process	*Function*
• End-to-end business processes deliver ***essential results*** by aligning the work of multiple functions • Must be identified and managed as a whole	• A centre of expertise – an efficient way to provide resources across multiple processes • Specialised skills, knowledge, tools • Organisational design usually based on functional specialties • We prefer not to use the somewhat negative term "functional silos"

Ultimately, business processes are all about alignment

Processes and functions – three key points



- The first step in managing processes is to *determine what they are* – they don't identify themselves
- Performance goals for the functions must *align with* (or be *balanced against*) the performance goals of the process
- Processes need an *owner / steward* to set direction, ensure alignment, and resolve conflict

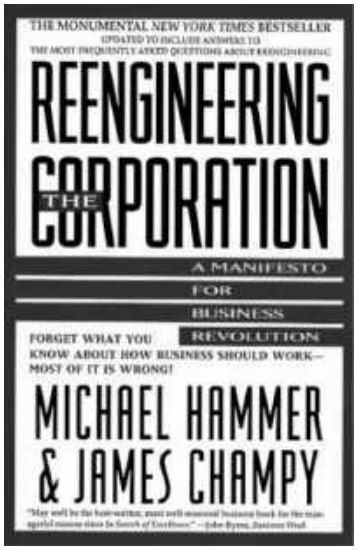
It takes concerted effort – nothing happens by accident

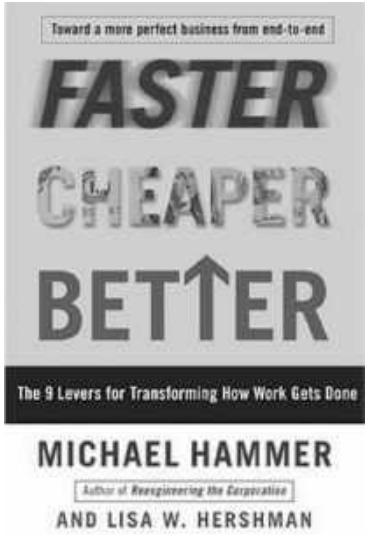
3 – Processes and information systems

1. It is essential to have clarity on what a *business process* really is
2. Performance measures may be *functionally aligned* - work *against* business processes
3. Enterprise system implementations must include a business process perspective
4. Success with business processes requires a *holistic view* in which *six enablers* are considered
5. A business process can't be great at everything – a single *differentiator* must be chosen

"Success with SAP Implementation"

Study by the late Michael Hammer, "godfather of BPR"



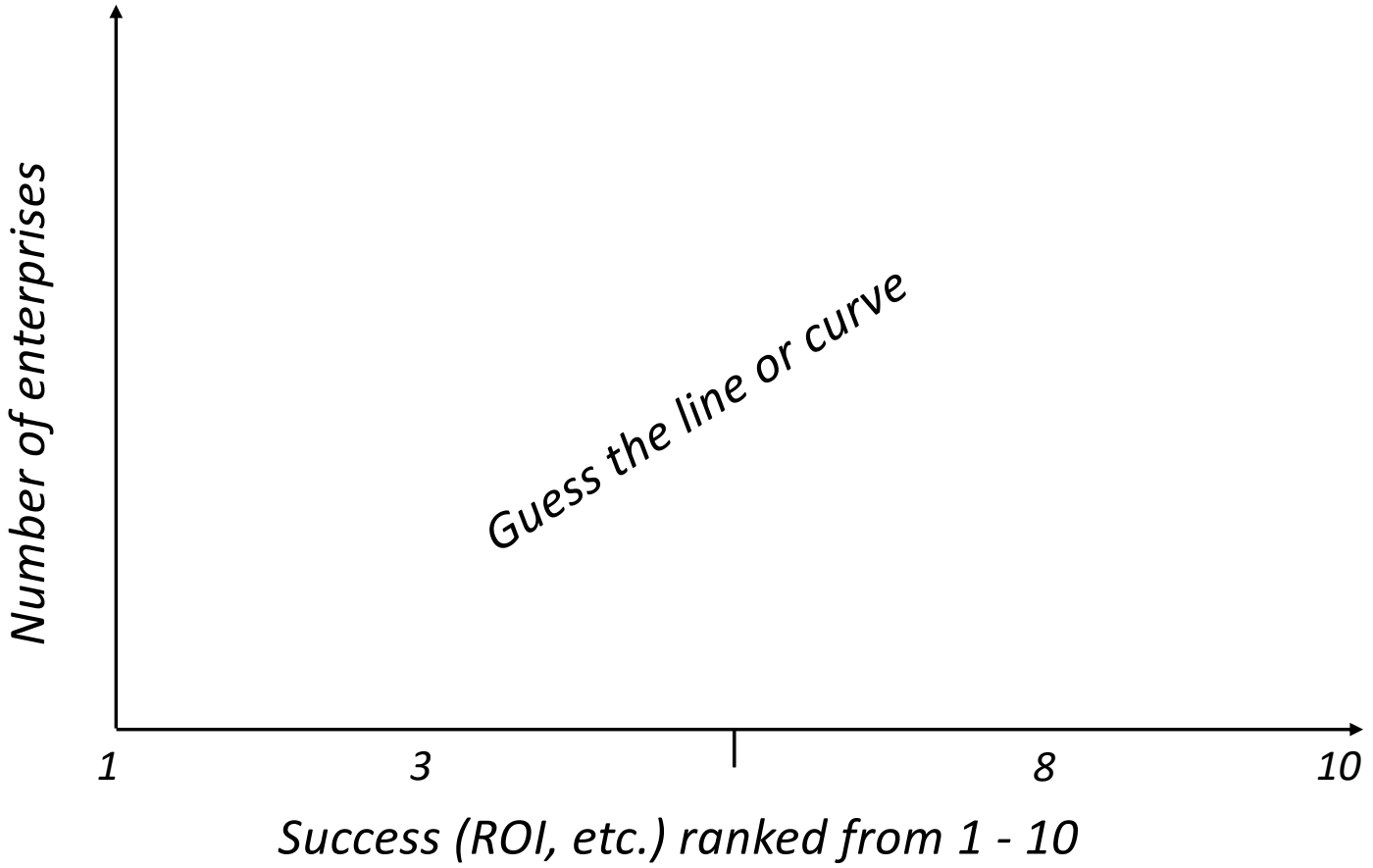


Observed that success of SAP implementations varied *wildly*

Worked with ~80 companies to assess their degree of success with SAP implementation

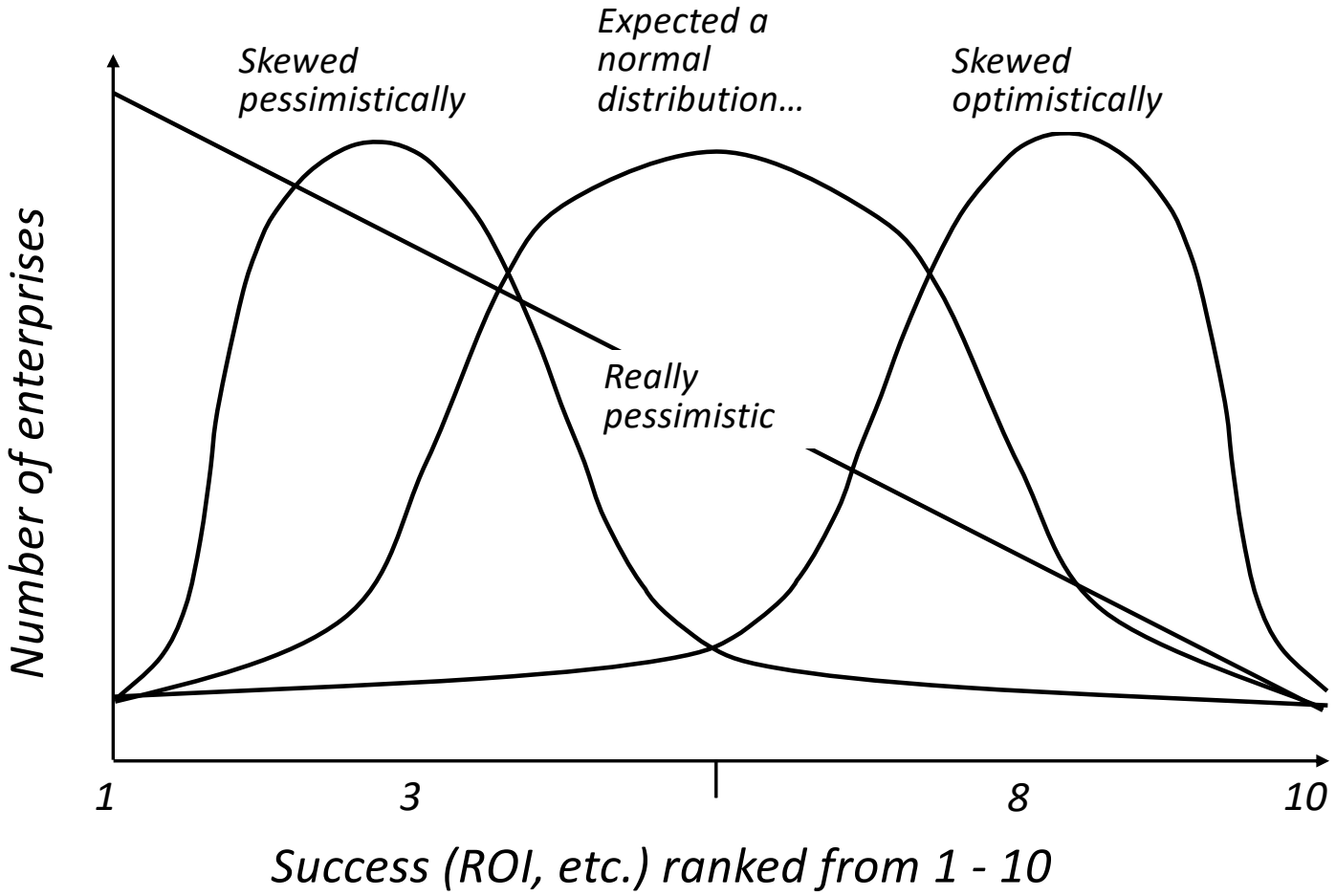
Success with SAP implementation

Hammer plotted the number of companies for each “success” ranking

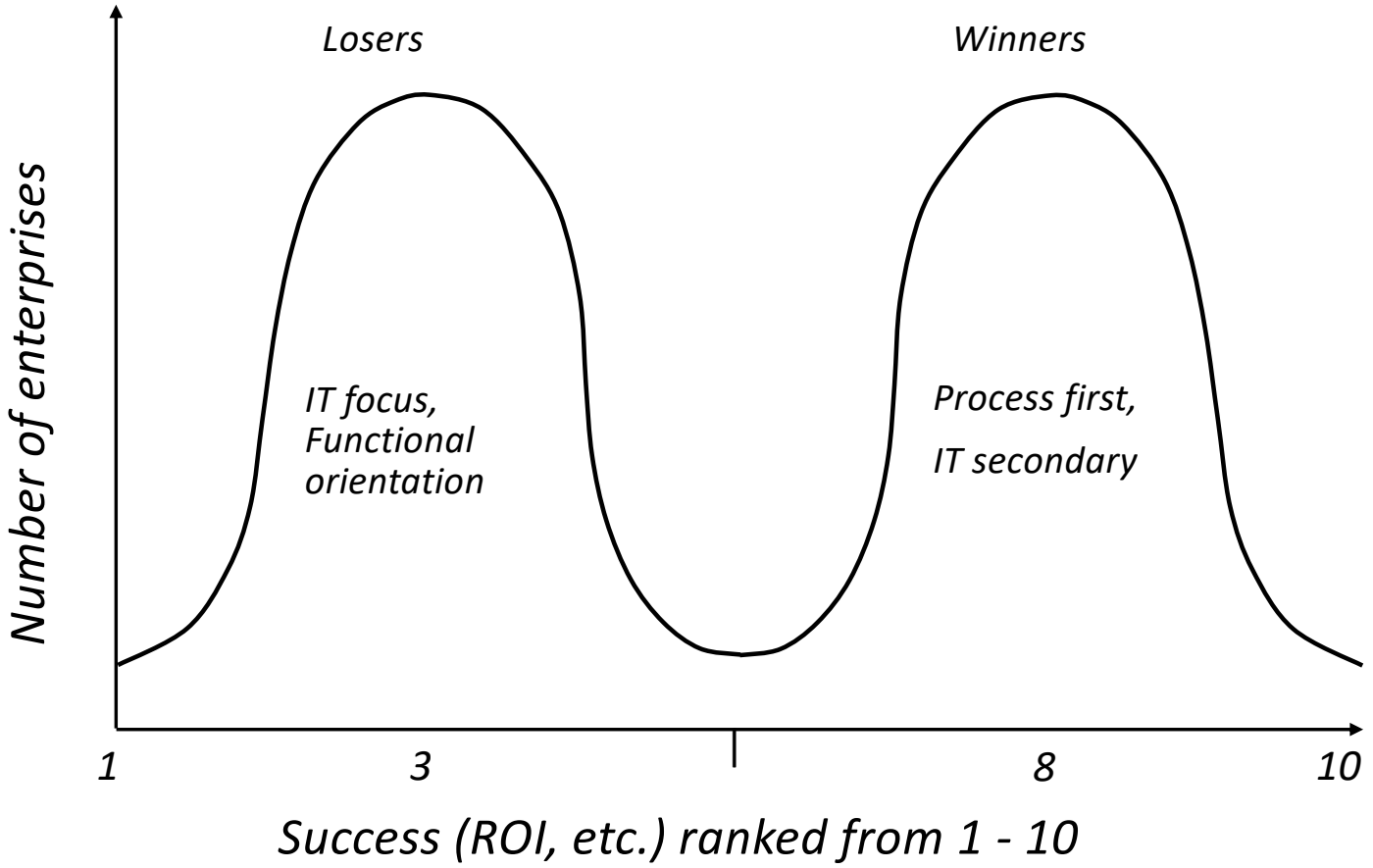


24

Hammer not sure what the outcome would be



The surprising result



Returning to an earlier example

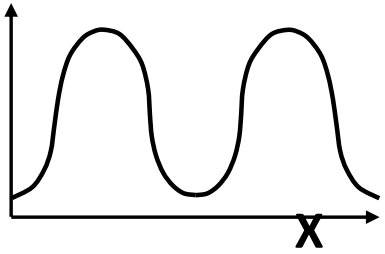
Global manufacturer implementing SAP

Four primary modules:

- *Sales*
- *Manufacturing*
- *Logistics*
- *Finance*

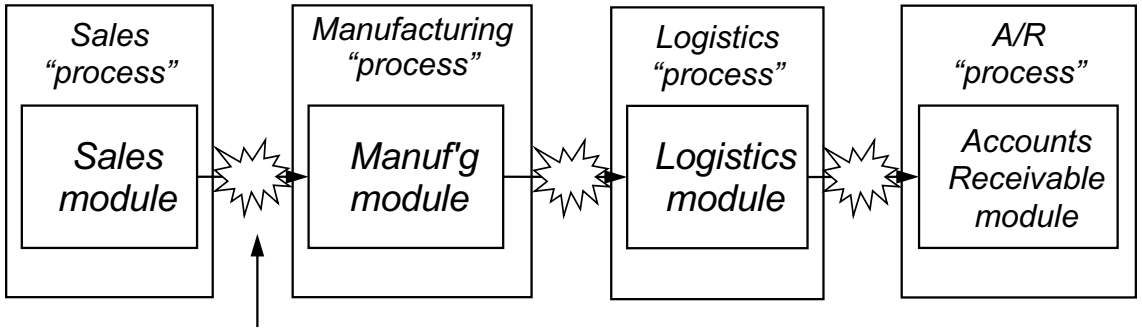
Determined to *do it right:*

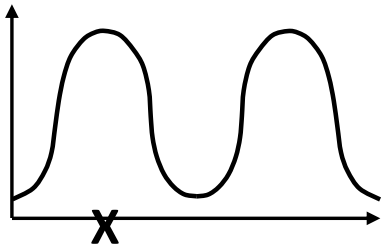
"This will be a *process-oriented* implementation!"



Impact of confusing function and process

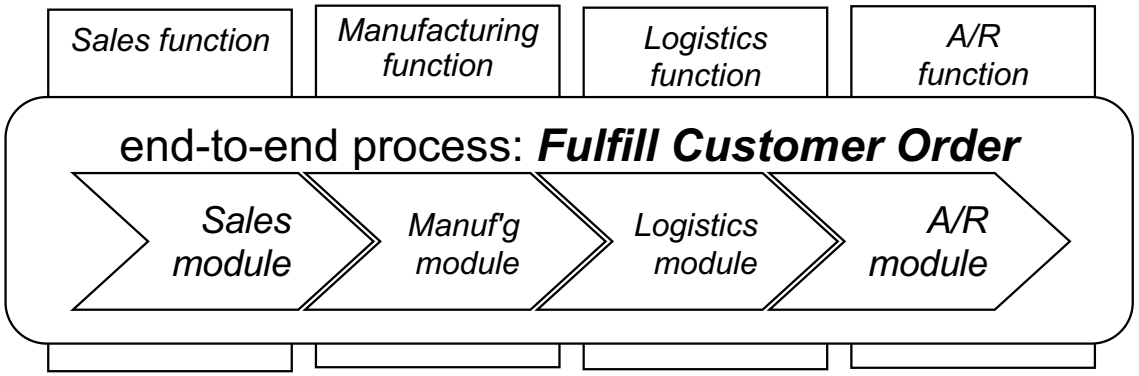
Implementing SAP without clarity on "process":

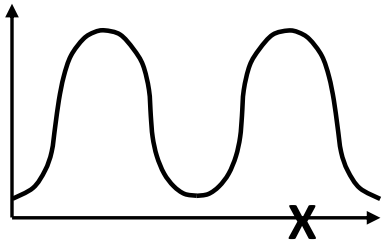




Conflicts: timing, coding, terminology, data formats, *performance targets*, …

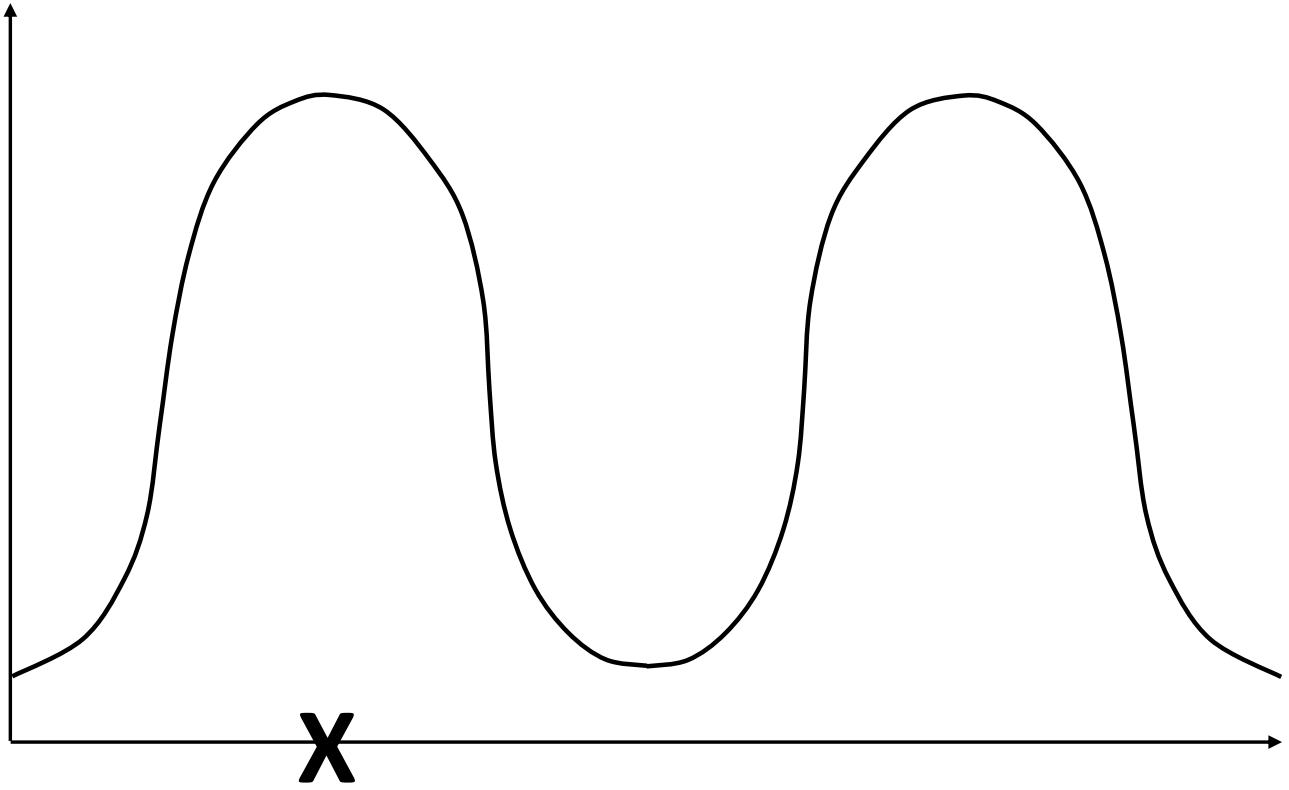
SAP re-implemented in a process-driven configuration:





Same software, radically different outcomes

Staying "right" in an "entropic" environment

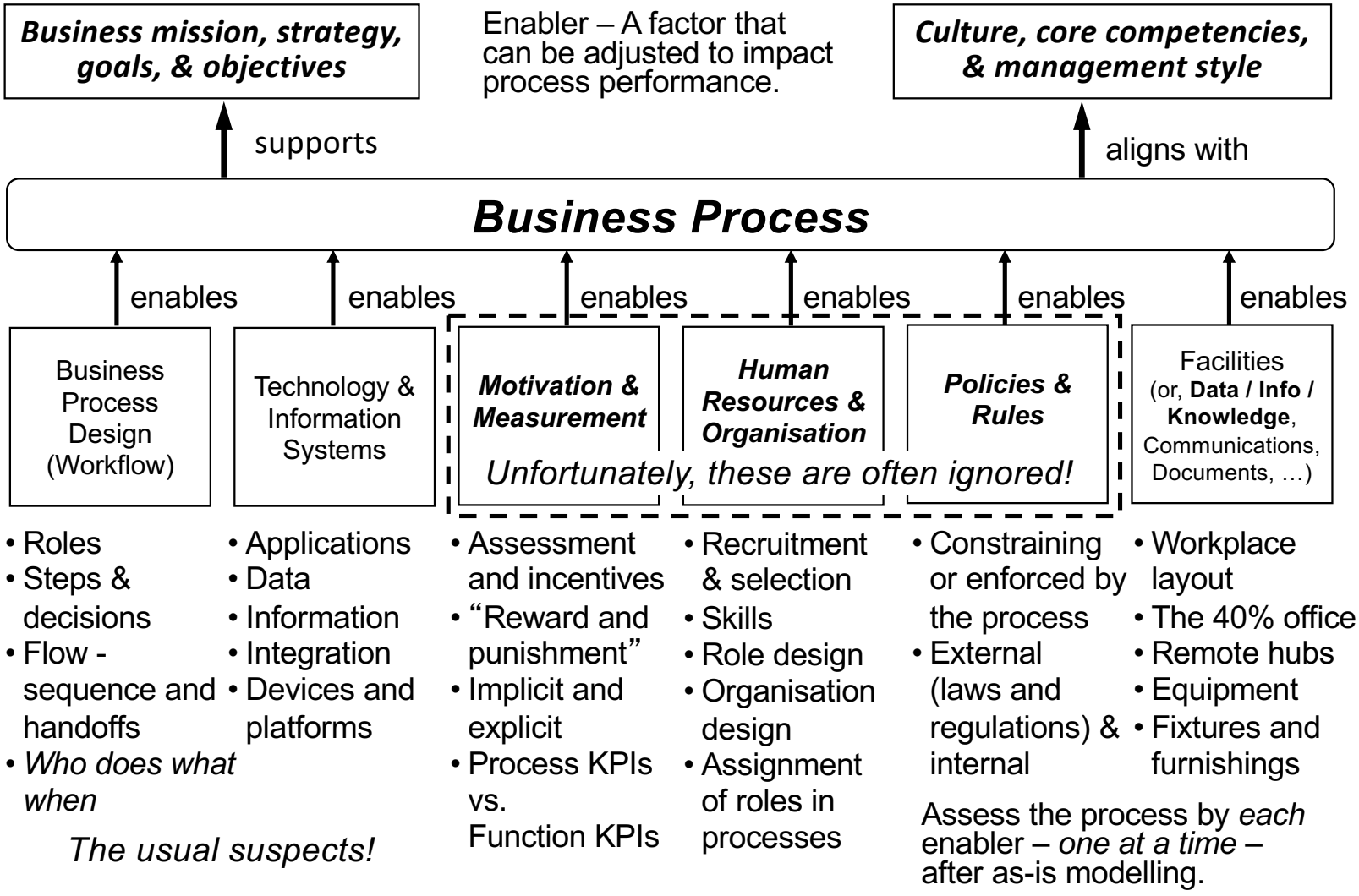


There will always be a pull back towards functional comfort

- *ongoing management of the process is critical!*
- *all enablers must be addressed for a sustainable process*

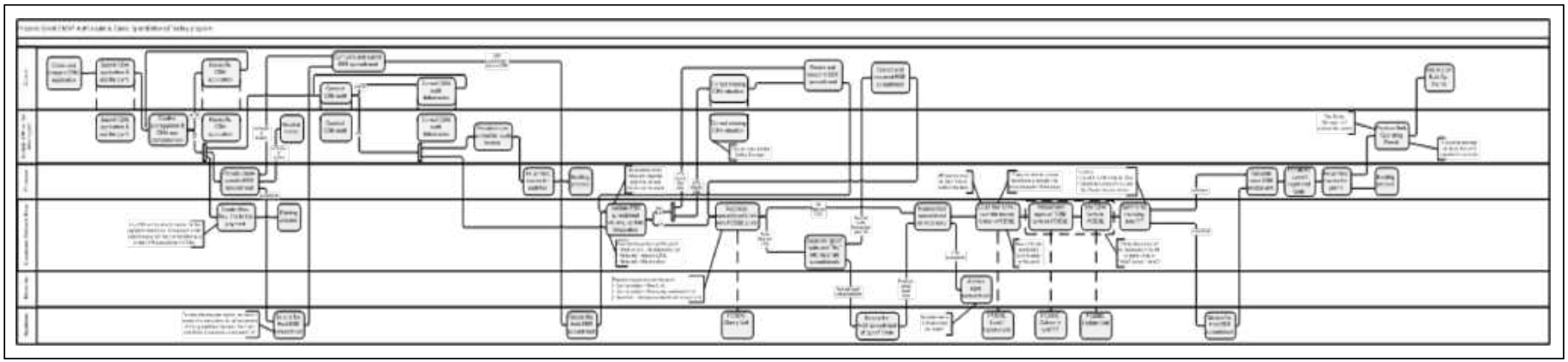
4. A holistic view for process analysis and design

1. It is essential to have clarity on what a *business process* really is
2. Performance measures may be *functionally aligned* - work *against* business processes
3. Enterprise system implementations must include a business process perspective
4. Success with business processes requires a *holistic view* in which six *enablers* are considered
5. A business process can't be great at everything – a single *differentiator* must be chosen



We model the as-is process to support assessment by enabler

As-is modelling maps *reality* – *who*, does *what*, *when*.



This supports a *fact-based* assessment of the *as-is* process by enabler.

Process Workflow Design: Is each step adding value, placed at the right point in the process, sequential or parallel as appropriate, performed by the best role, etc.?	***Information Systems & Technology:*** Are the process, the steps, and the actors supported by the right systems and technology?	***Motivation & Measurement:*** How is the performance of the steps, the actors, the participating functions, and the process measured, and what are the consequences?	***Human Resources & Organisation:*** Are roles suitably broad, are organisations designed properly, and are roles & skills deployed well into the process?	***Policies & Rules:*** What policies or rules , whether internal or external, constrain or are enforced by the process, and what is their impact?	***Facilities (or other):*** Are the layout & furnishings optimal or do they impede the process? (Many clients instead use this enabler to consider data, info, and knowledge.)

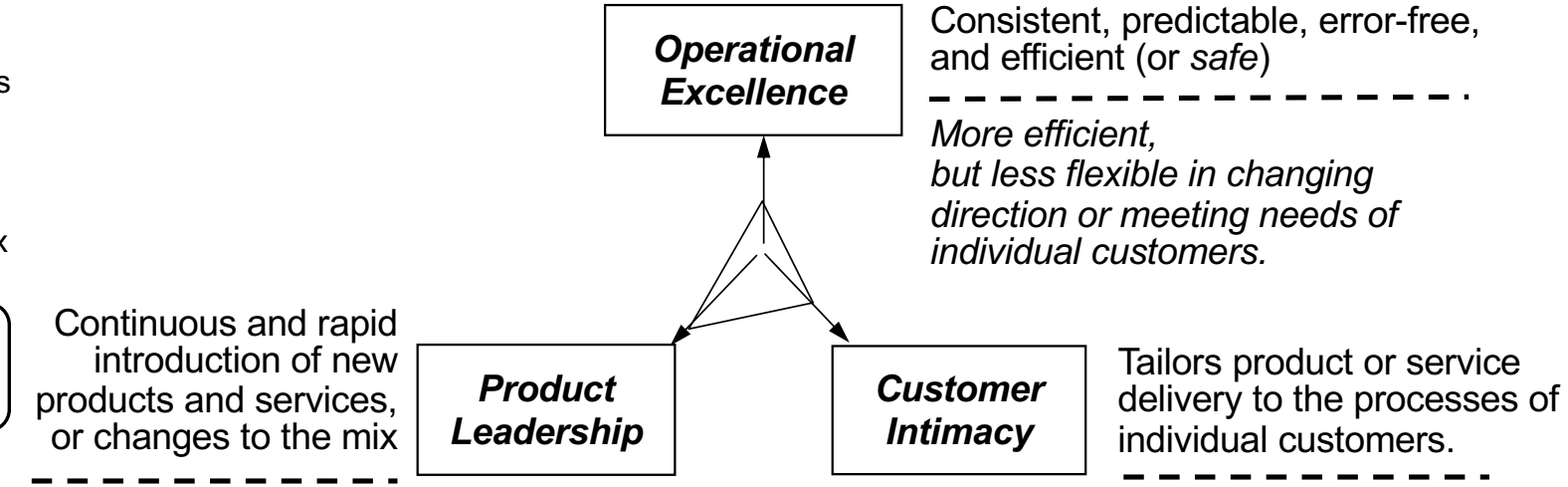
5. Process goals: know your "differentiator"

1. It is essential to have clarity on what a *business process* really is
2. Performance measures may be *functionally aligned* - work *against* business processes
3. Enterprise system implementations must include a business process perspective
4. Success with business processes requires a *holistic view* in which six *enablers* are considered
5. A business process can't be great at everything – a single *differentiator* must be chosen

As noted, this is one of the things I do on ~100% of *Project Recovery* assignments -

1. Build *Process Scope Model* & *Process Summary Chart*
2. Develop *Case for Action* – an *As-Is Assessment by Stakeholder*
3. Establish the *Differentiator*
4. *(*Optionally conduct an *As-Is Assessment by Enabler)*

Great processes don't try to be all things to all people –
strive to be ***great*** at one differentiator, and ***good*** at the other two…



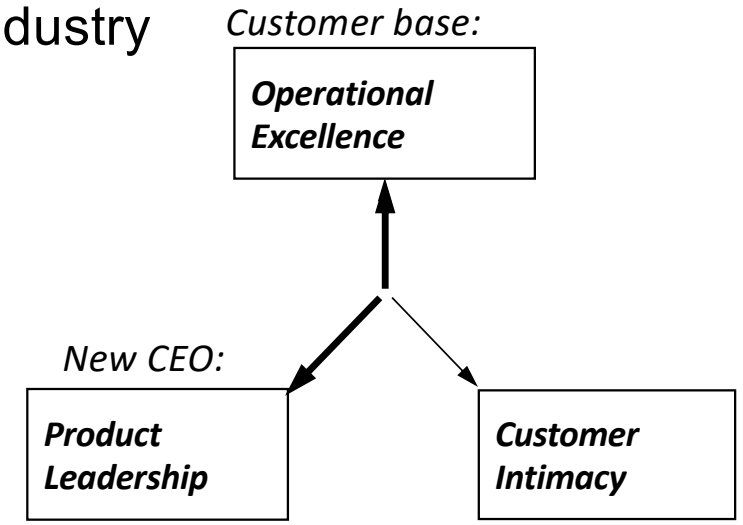
The original reference:
The Discipline of Market Leaders
Michael Treacy and Fred Wiersma
Addison-Wesley 1995

1. Concept developed for the entire enterprise, but excellent for individual processes – a "signpost" for decisions on process changes.
2. Processes in an enterprise do not all have the same differentiator.
3. The Process Differentiator can change over time – *slowly!*

Example: "differentiator confusion"

Getting it wrong can be *expensive…*

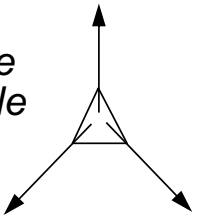
- Insurance company recruits CEO from high tech industry
- New CEO decides "innovation is everything" – $100M spent on process redesign and system development in support of "innovative car insurance products" – *Product Leadership*
- Total failure – customers wanted affordable, easy to understand, easy to buy insurance – *Operational Excellence (Op Ex)*



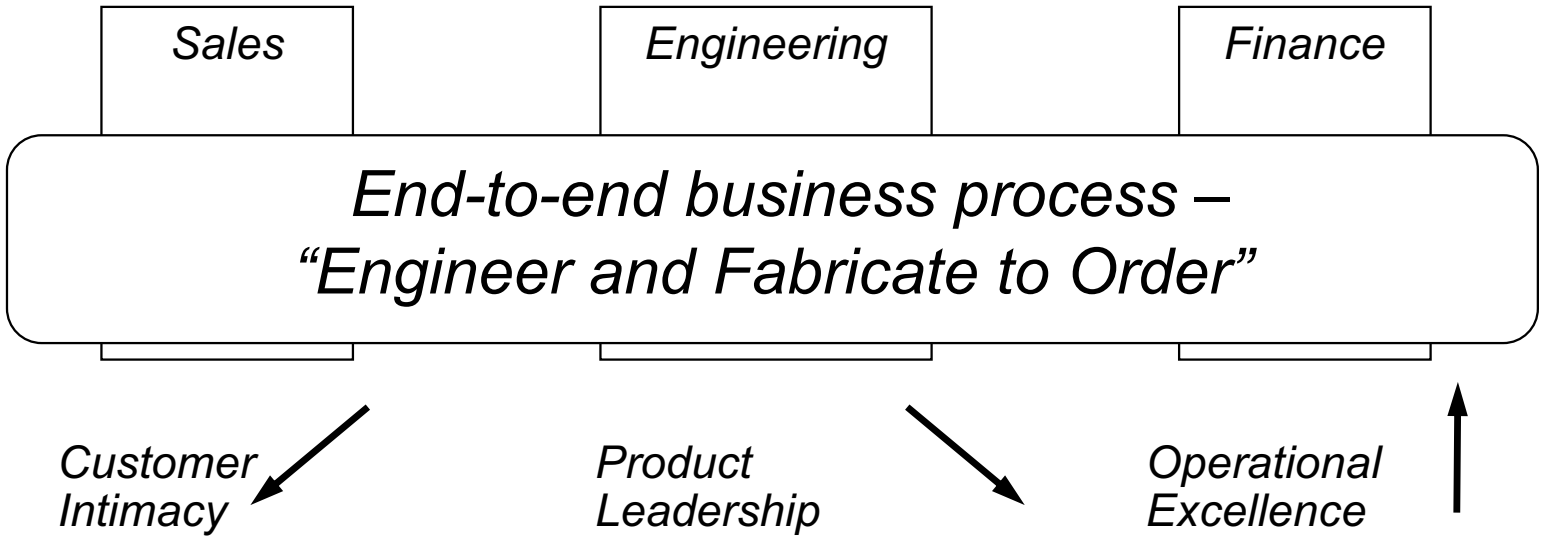
Three common differentiator problems

1. Focus on the wrong differentiator – *customer alienation*
2. No differentiator or trying to excel at *multiple* differentiators – *stressed workforce* and *lower performance*
   - *Operational excellence* – "We must be the low-cost provider!"
   - *Customer focused* – "We must do what it takes for each client!

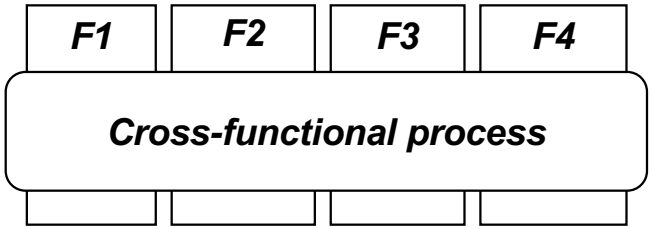
Stuck in the Bermuda Triangle



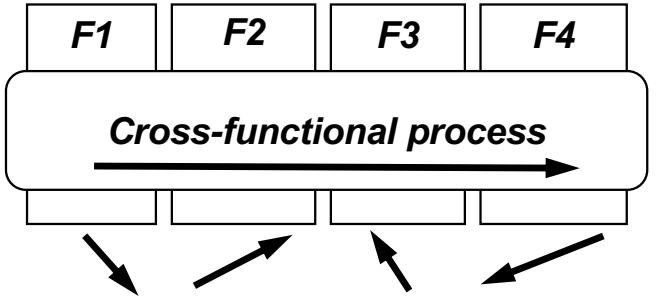
3. Conflicting differentiators within functions of a process – *lower performance*



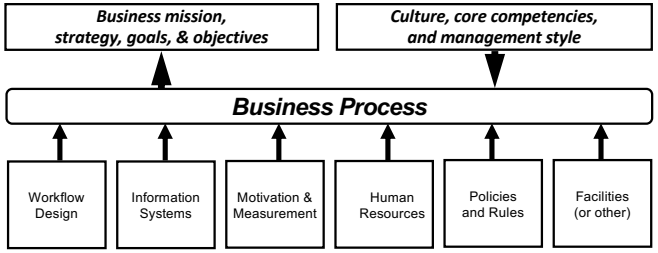
Five key points about Business Processes



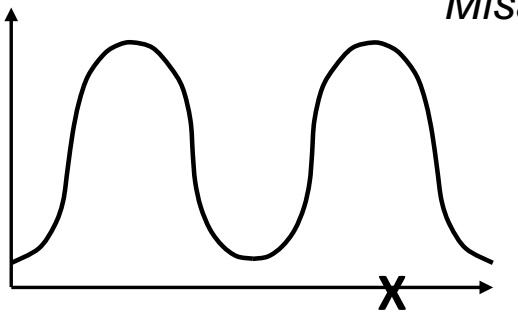
Processes:
"large" and X-functional



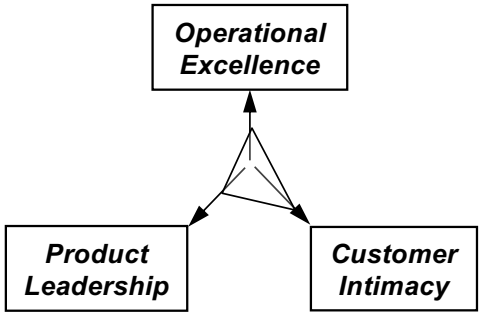
Misaligned measures



Holistic method



System success



Differentiator

Thanks again!

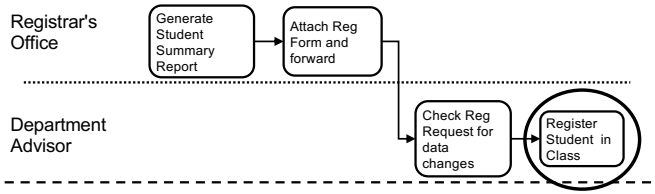
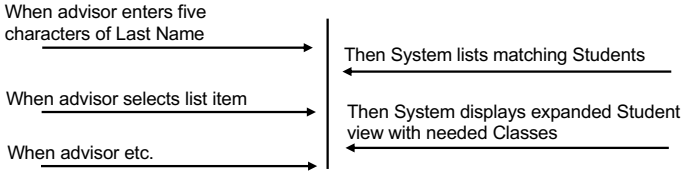
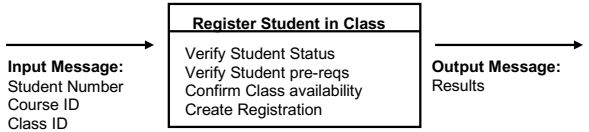
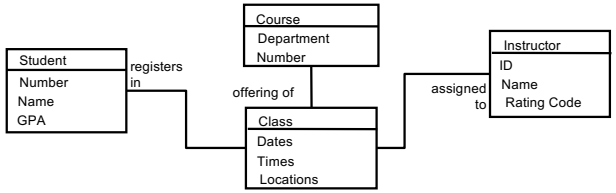
Alec Sharp, West Vancouver, BC, Canada

If you have questions or comments…
don't be shy, get in touch!

- e: asharp@clariteq.com
- t: @alecsharp
- ig: @alecsharp01
- m: +1 604 418-3352

Some extras…

Business Process – part of a proven framework for Business Analysis

	Framework Layer	Technique sample	What it covers
Goals	*Business Objectives*	The university is initiating the "Strategic Enrollment" program to raise Student graduation rates in part by ensuring Classes are available for Student registration when needed.	✓ ***Project Charter:*** documents the rationale, objectives, scope, and success measures for the project
Process	*Business Process*		✓ ***Process Model:*** shows "what" in a Scope Model, then "who & how" in a Workflow Model – the steps done by the actors in the process
Application	*Presentation Services (user interface)*		✓ ***Use Case:*** describes how an actor would like to interact with a system to obtain a service, typically to complete a step or task within a process
	Business Services (rules & logic)		✓ ***Service Specification:*** describes a service – a package of rules and logic – that is triggered to complete or respond to a business event
Data	*Data Mgmt. Services (databases)*		✓ ***Concept Model:*** depicts the things and the facts about things the organisation needs to record; the things (the Entities) are what processes and solutions act on.

This is not a fixed sequence!

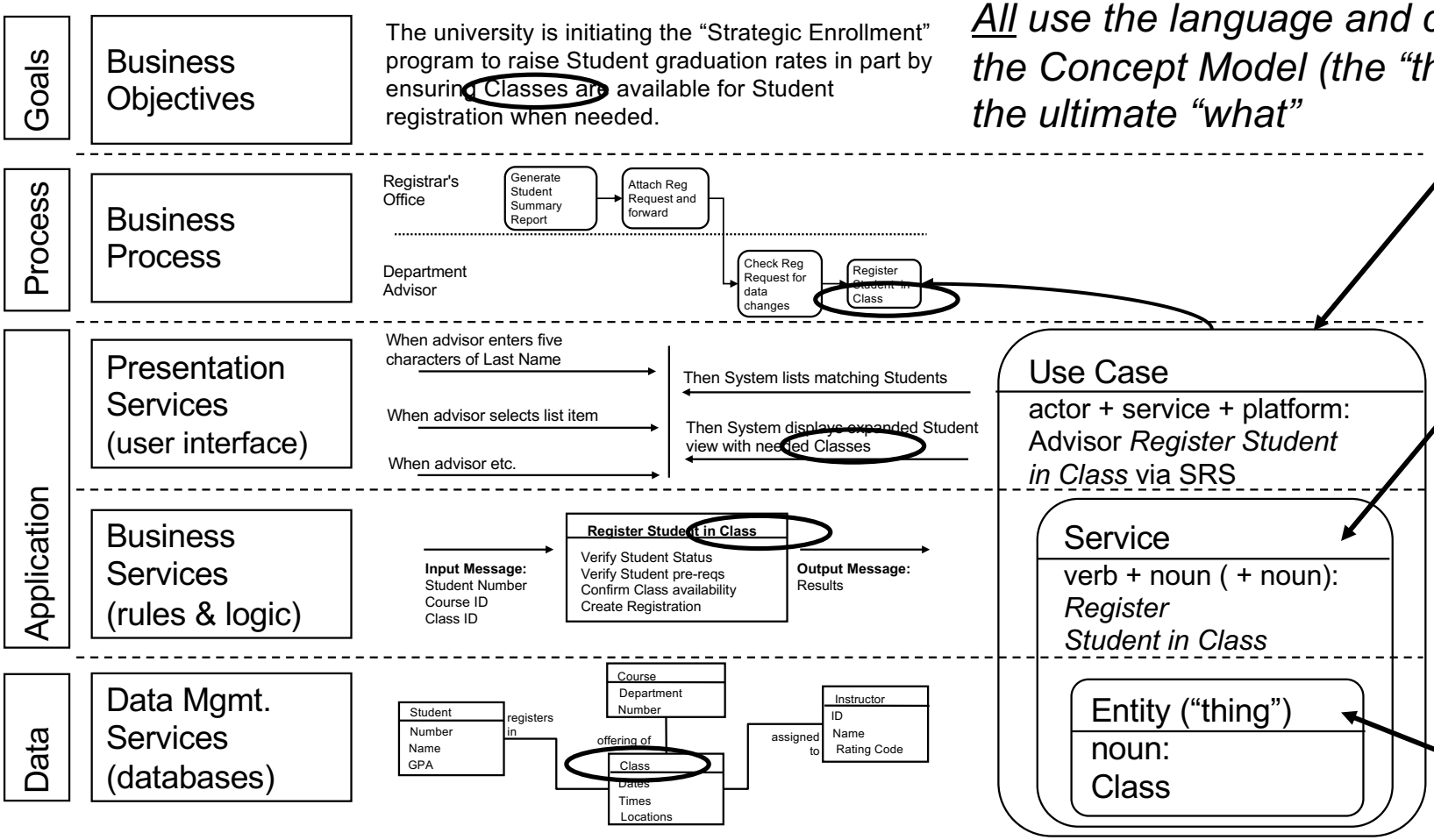
Business Process – gives great *context* for *Business Analysis*

Use Cases and Services – where we capture *Functional Requirements*

Concept Model – a great *platform* for *Business Analysis*

Only four types of models vs. 14 in the UML! (Unified Modelling Language)

Key point! Everything relies on the concept model



All use the language and constraints of the Concept Model (the "thing model") – the ultimate "what"

Use Cases/User Stories:
- Who (Actors) needs access to the Services, and how (Platform)?

Verb-Noun pairs:
- The *Services* (event-handlers) that are at the heart of a *Service Oriented Architecture.*
- *Also* "building blocks" of Business Processes

The core *Nouns* in your enterprise. Also known as *Business Objects.*

Bonus – great starting point to discover your Events/Services and Use Cases/User Stories

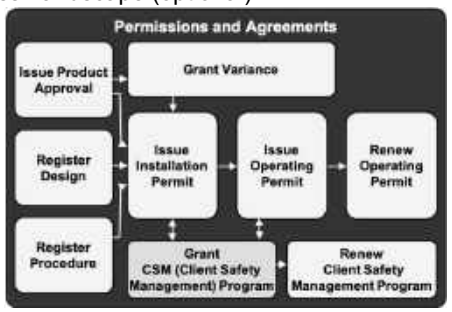
Another key point! Different levels of detail for different purposes

Different models and levels of detail for different audiences and purposes.

Also applies to Use Cases, Services, and Data Models

Scope – for Planning

Process Landscape (optional):



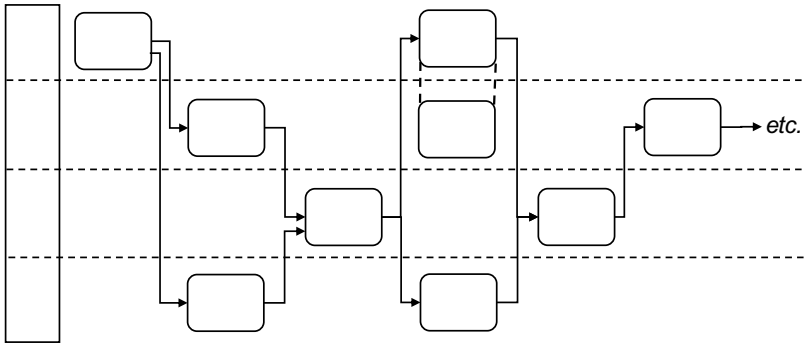
Process Scope Model:

Process Summary Chart:

Boxes

Concept – for Understanding

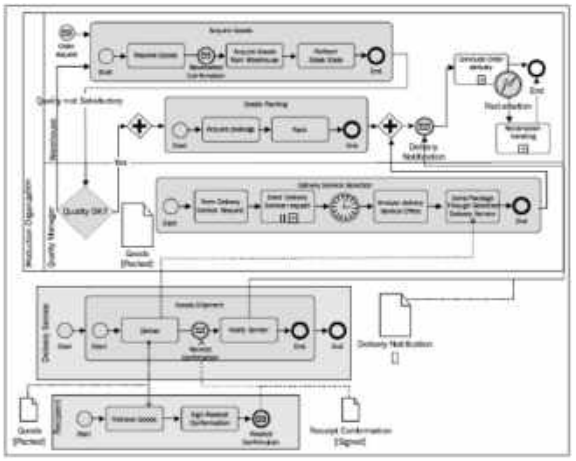
- Augmented Scope Model showing next level activities: *who - what - how*
- "Business-friendly" (just boxes & lines) flow models to maximise communication and participation
- Two levels – *Handoff* and *Service*



Boxes & Lines

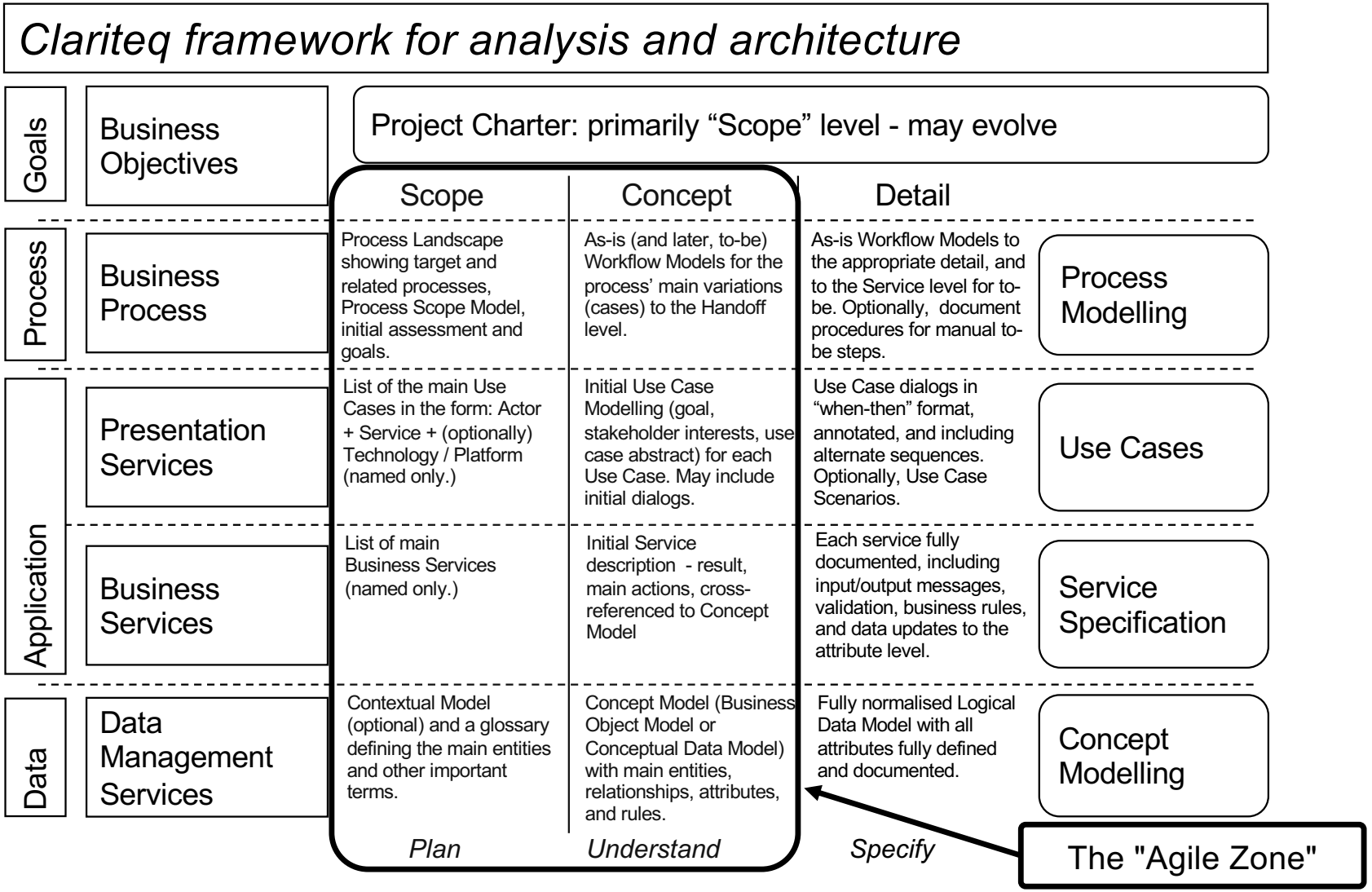
Detail – for Specification

- Detail for technical design, perhaps using full BPMN



Boxes, Lines, & MANY Symbols

Specifics on progressive detail for all techniques



Our three-phase methodology – proven, practical, & agile

1 *Establish Process Scope and Objectives*

2 *Understand the As-Is Process*

3 *Design the To-Be Process*

Goal or issue, not rigorously specified

Identify & scope the process with a Scope Model & a Process Summary Chart; Optional - build a Concept Model

*Complete **initial** as-is process assessment, and to-be objective setting, by stakeholder*

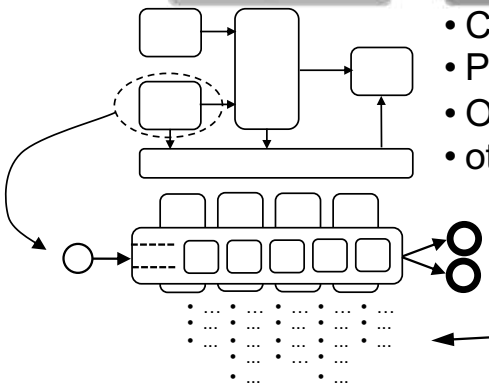
*Perform more **detailed** as-is process analysis:*
- Augmented Scope Model
- Optionally, draw workflow

*Complete **final** as-is process assessment by enabler, and generate to-be improvement ideas*

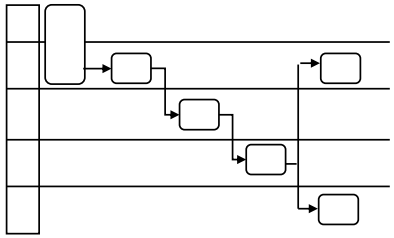
*Refine to-be improvement ideas, determine 5–10 **key** features of the to-be process*

*Assess each to-be feature **by enabler** to determine changes to make it sustainable*

Design to-be process:
*1 - **essential** activities first*
2 - "who & how"
3 - transport & protocol

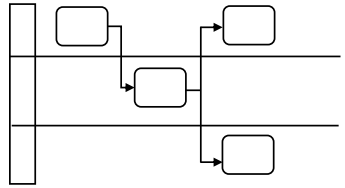


- Customer
- Performers
- Owner
- others…



- Process
- IT
- M&M
- HR
- P&R
- Fac. or…

- Select key to-be *Features*

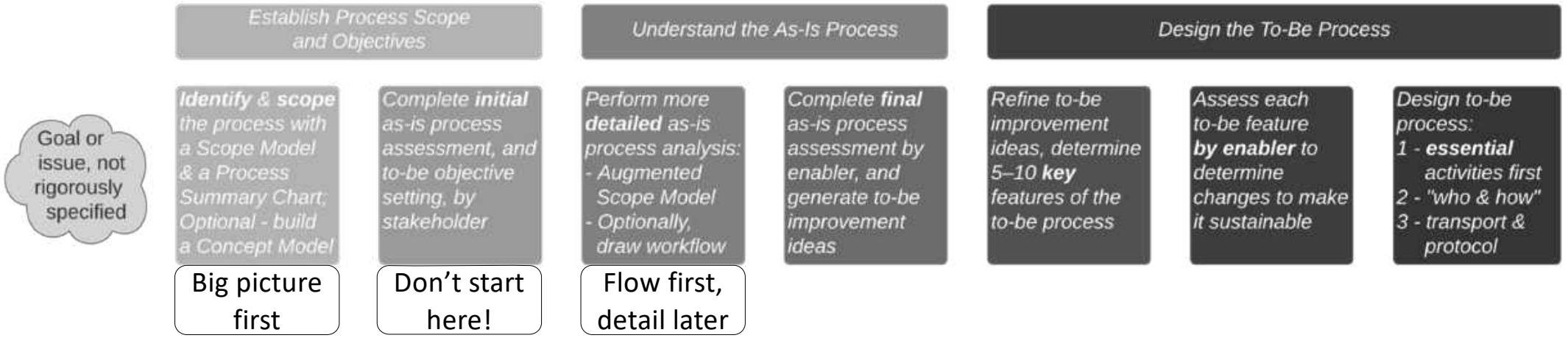


- ID processes & draw *Process Landscape* (Optional – only if you have a large scope)
- ID **T**rigger, **R**esults, main **A**ctivities, **C**ases (**TRAC**) & draw *Process Scope Model* – focus on *what, no* reference to *who* or *how*
- ID involved functions & mechanisms (*who* and *how*) & draw *Process Summary Chart*
- Conduct *stakeholder-based assessment*

- Develop *as-is* models:
  - *Augmented Scope Model* – add ~5 – 7 more detailed Activities for each main Activity
  - (Optional) as-is *Workflow Models* – *only* enough detail to understand process behaviour
- Conduct *enabler-based assessment* and identify *potential improvements*

- Assess each key *Feature* by enabler
  - Identify and sequence *essential* activities
  - Develop *to-be Workflow Models* depicting the future *who* and *how*
  - *…on to requirements definition and implementation*

Our methodology – three responses to three common difficulties



My *hardest* assignments

1 – Premature diagnosis of the situation

Don't start with a problem statement!
There will be some goal or issue, but don't formalise it ***yet***.
And remember… it may not be a "process" issue.

Why not?

2 – Failure to identify true end-to-end processes

Rigorous techniques to identify real business processes – a Process Scope Model and a Process Summary Chart make scope and context visible.

3 – A rapid descent into unhelpful detail

Clarify the big picture, then take a *controlled* descent with well-defined levels of detail.

Our methodology – two points highlighted by clients

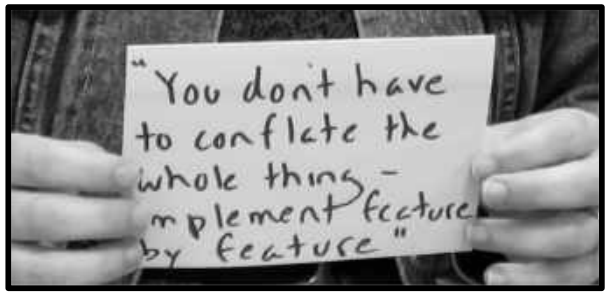
Establish Process Scope and Objectives		Understand the As-Is Process		Design the To-Be Process		
Identify & scope *the process with a Scope Model & a Process Summary Chart; Optional - build a Concept Model*	*Complete **initial** as-is process assessment, and to-be objective setting, by stakeholder*	*Perform more **detailed** as-is process analysis: - Augmented Scope Model - Optionally, draw workflow*	*Complete **final** as-is process assessment by enabler, and generate to-be improvement ideas*	*Refine to-be improvement ideas, determine 5–10 **key** features of the to-be process*	*Assess each to-be feature **by enabler** to determine changes to make it sustainable*	*Design to-be process: 1 - **essential** activities first 2 - "who & how" 3 - transport & protocol*
Start with *what*	*Inclusive* assessment	Based on *reality*	Awareness of *all* factors	Addresses *our* goals	We can *do* it!	Feature-based approach

Goal or issue, not rigorously specified

Builds support for *change*

Not a "big bang" – an effective, implementable, sustainable business process

"We like the way support for change is built in *throughout* your approach, not bolted on at the end."



Feature-based approach makes it *Agile* / *iterative.*

And *fast!* – up-front work avoids endless rehashing later