A Simulationist's Framework for Business Analysis

R.P. Churchill

CBAP, PMP, CLSSBB, CSPO, CSM, CSD

www.rpchurchill.com/presentations/SimFrameForBA

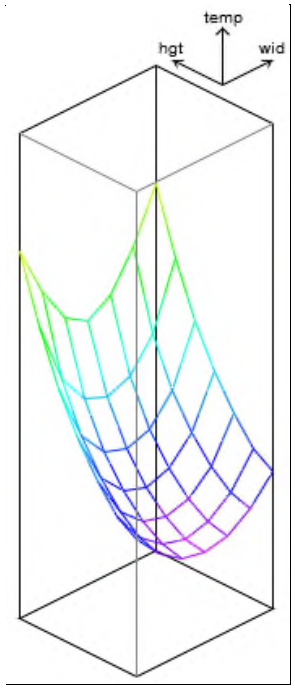

tvdays.com

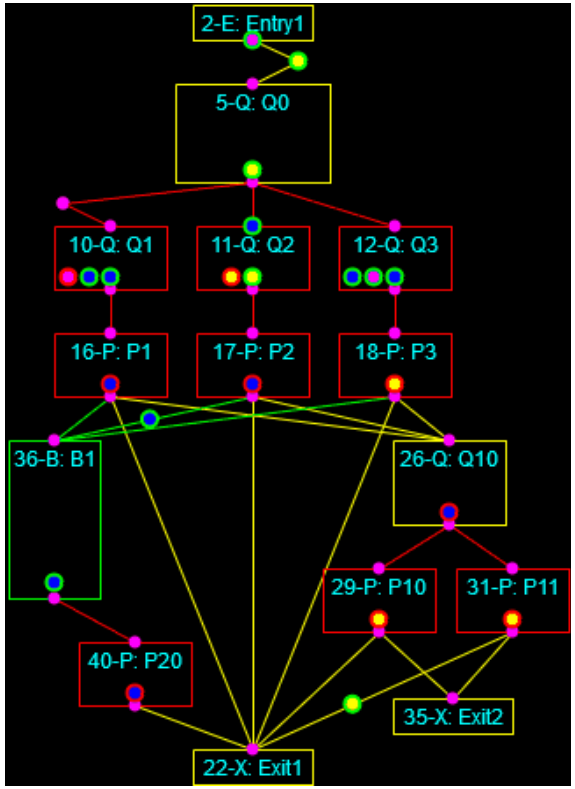
Moschitta-Hawking Scale




30 Years of Simulation

iterations: 34





Continuous simulation of the heating of a square billet and Discrete-Event simulation of a multi-phase process.

30 Years of Simulation

Industries

- Paper (Chemical/Process)
- Nuclear (Power Generation)
- Metals (Steel, Non-Ferrous)
- HVAC (Building Control)
- Insurance, Banking, Legal
- Security, Inspections
- Passenger Processing
- Medical Facilities
- Evacuations, Area Control
- Threat Response
- Logistics, Supply
- Maintenance and Reliability
- Staff Level Determination
- Fleet Management

Applications

- Design and Sizing
- Operations Research
- Real-Time Control
- Operator Training
- Risk Analysis
- Impact Analysis
- Economic Analysis
- Process Improvement (BPR)

Architectural Considerations

- Continuous vs. Discrete-Event
- Interactive vs. Fire-and-Forget
- Real-Time vs. Non-Real-Time
- Single-Platform vs. Distributed

The Framework:

- Project Planning
- Intended Use
- Assumptions, Capabilities, and Risks and Impacts
- Conceptual Model (As-Is State)
- Data Sources
- Requirements (To-Be State: Abstract)
  - Functional (What it Does)
  - Non-Functional (What it Is, plus Maintenance and Governance)
- Design (To-Be State: Detailed)
- Implementation
- Test
  - Operation and Usability (Verification)
  - Outputs (Validation)
- Acceptance (Accreditation)

BABOK Knowledge Areas vs. Bob's Framework

Bob's Technique	Business Analysis Planning and Monitoring	Elicitation and Collaboration	Requirements Life Cycle Management	Strategy Analysis	Requirements Analysis and Design Definition	Solution Evaluation	Requirements per BABOK
Project Planning	**X**			x			
Intended Use	x	**X**	x	x		x	Business Requirements
Assumptions, Capabilities, Limitations, and Risks & Impacts	x	**X**	x				
Conceptual Model (**As-Is** State)	x	**X**			**X**		
Data Sources	x	**X**					
Requirements (**To-Be** State: Abstract)	x	**X**	**X**	x	**X**		Stakeholder Requirements
Design (**To-Be** State: Concrete)	x			x	**X**	**X**	Solution Requirements (Functional and Non-Functional)
Implementation	x	**X**	x	**X**	x	x	Transition Requirements
Test Operation and Usability (**Verification**)	x					**X**	
Test Outputs (**Validation**)	x	x				**X**	
Acceptance (**Accreditation**)	x					**X**	

Project Planning

Sets up and kicks off the project using traditional project management techniques.

Considers intended management framework (Waterfall, Agile, Kanban, etc.) and special requirements for the effort (existing assets/PMO, identify stakeholders)

Intended Use

This defines the customer's goals for what the new or modified process or system will accomplish.

It may describe technical and performance outcomes but must ultimately be expressed in terms of business value.

Each goal can be described in terms of:

- Key Questions
- Application
- Outputs and Data

This information is included in the Project Charter from the PMBOK.

Assumptions, Capabilities, and Risks & Impacts

Define the scope of the project and what capabilities and considerations will and will not be included.

Describe the risks inherent in the effort and the possible impacts of risk items occurring.

Reasons to omit features and capabilities:

- Outside of natural or organizational boundaries
- Insufficient data or understanding
- Impact on results is small (benefit not worth cost)
- Components aren't active in modes being investigated

Conceptual Model (As Is State)

If an existing process is to be modified, improved, or automated, discover all operations and data items. This defines the **As Is State**. (In simulation this is known as building a **Conceptual Model**.)

If there is not an existing process, work backwards from the desired outcomes to determine what operations and data are required.

Map out the discovered process and document and collect data and parameters for each operation and communication.

The conceptual model is not a specific type of drawing, but is a representation of an existing system using any techniques appropriate for the job.

Iteratively review the maps, data, and descriptions with customers and SMEs until all parties agree that understanding is accurate and complete!

Data Sources

Data in this context refers to input items (nouns) that are processed by the system, as opposed to the parameters (adjectives) that describe the operation and characteristics of the system.

Data sources, sinks, and messages should be mapped to the Conceptual Model (As Is State representation).

Data items must be validated for accuracy, authority, and (most importantly) obtainability.

Interfaces should be abstract initially (e.g., with management and through initial discovery and scoping), and then detailed in design and implementation with proper SMEs.

BAs that haven't worked with code and databases directly should work with implementation SMEs (software engineers) to ensure that data and flags, states, formats, and metadata are captured in sufficient detail.

Requirements (To-Be State: Abstract)

Functional

- What the system **DOES**
- Describes components, behaviors, entities, actions, inputs, and outputs.
- Contains the details of the design the user sees and the mechanisms that generate results.

Non-Functional

- What the system **IS**
- Describes qualities (in terms of "-ilities," e.g., reliability, modularity, flexibility, robustness, maintainability, scalability, usability, and so on).
- Describes how the system is maintained and governed.
- Describes how the system is hosted.

The requirements include the criteria by which functional and non-functional elements will be judged to be acceptable.

This represents the **To-Be State** in abstract terms.

All items in the requirements should map to items in the **Conceptual Model** in both directions. This mapping is contained in the **Requirements Traceability Matrix (RTM)**, which can be implemented in many ways.

Design (To-Be State: Concrete)

The design of the system is a description of how the system will be implemented and what resources will be required.

The design of the system also includes plans for maintenance and governance going forward.

All elements of the design must map in both directions to all elements of the Conceptual Model and Requirements. This is accomplished by extending the **Requirements Traceability Matrix (RTM)**.

BAs may or may not participate in the design of the system directly, but must absolutely ensure that all elements are mapped to previous (and subsequent) elements via the **RTM**.

Implementation

This phase is where the implementation is actually carried out, based on the design.

This involves project management methodologies and strategic decision-making for the transition process: prioritizing items for implementation (according to business value and architectural logic), bringing new capabilities online (including training), taking existing capabilities offline, and so on.

All elements of the implementation must map in both directions to all elements of the Conceptual Model and Requirements. This is accomplished by extending the **Requirements Traceability Matrix (RTM)**.

Waterfall, Agile, and other methodologies may be used as appropriate to the scope and scale of the effort.

Test

Operation and Usability (Verification)

- Tests to ensure the system *operates* as intended.
- This process ensures that the system:
  - makes sense to the users
  - enables manipulation of all relevant elements
  - prevents and corrects errors
  - maintains users' situational awareness
  - includes documentation, training, and help
- These types of tests are most able to be automated.

Outputs (Validation)

- Tests to confirm the accuracy of the results produced.
- For simulations ensures behavior matches real-world system for known cases
- Validation of results may be:
  - Objective, e.g., measured comparisons to known values in a simulation or calculation
  - Subjective, e.g., judged as "correct" by SMEs for novel situations and realizations of business value
- The outputs are what provide the business value of the solution.

Test (continued)

All elements of the test plan and results must map in both directions to all previous elements in the **Requirements Traceability Matrix (RTM)** in both directions.

Specialized test SMEs may conduct the majority of system testing, but implementors, managers, customers, maintainers, and end users should all be involved.

Provisions for testing, V&V, and quality should be built into the process from the beginning.

Acceptance (Accreditation)

This phase ensures that the customer's plans and criteria for acceptance are met. All of the stated acceptability criteria must be addressed.

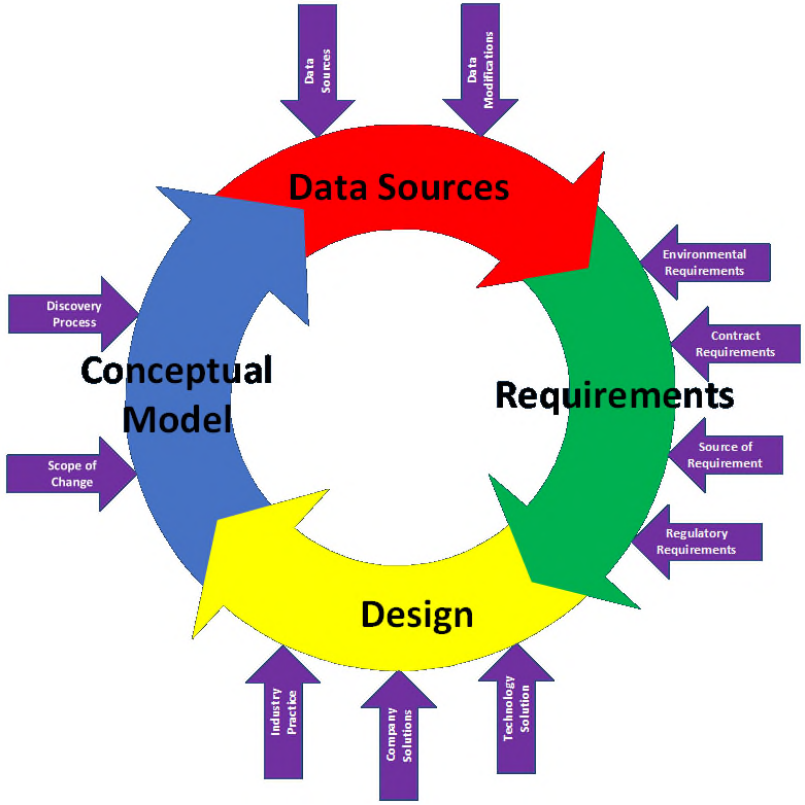
This plan must include the process for handing the system or process over to the customer (internal or external). This process may include documentation, training, hardware, software, backups, licenses, and more.

The customer is the final judge of acceptance and may make three judgments:

- Full Acceptance
- Partial Acceptance with Limitations
- Non-Acceptance

Clarification: Conceptual Model through Design

The Intended Use drives everything.



Discovery vs. Data Collection

Discovery is a **qualitative** process. It identifies *nouns* (things) and *verbs* (actions, transformations, decisions, calculations).

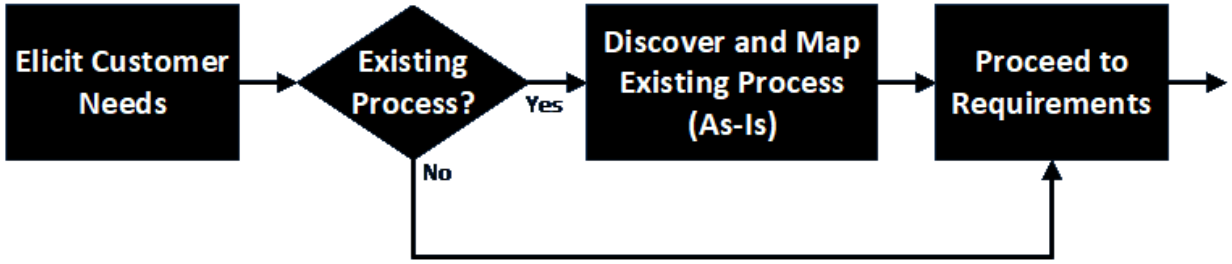
Data Collection is a **quantitative** process. It identifies *adjectives* (colors, dimensions, rates, times, volumes, capacities, materials, properties).

Discovery comes first, so you know what data you need to collect.

Imagine you're going to simulate or automate the process. What values do you need? This is the information the implementation teams will need.

Context of Discovery

Elicitation is discovering the customer's needs. Discovery is about mapping the customer's existing process.



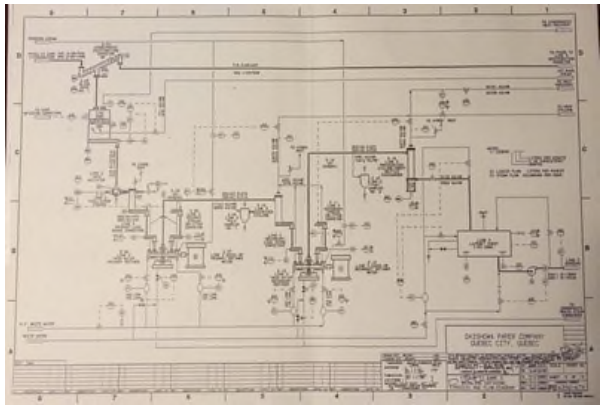
If there is no existing process, i.e., a new (*greenfield*) process is being built from scratch, then a form of discovery will occur during Requirements and Design.

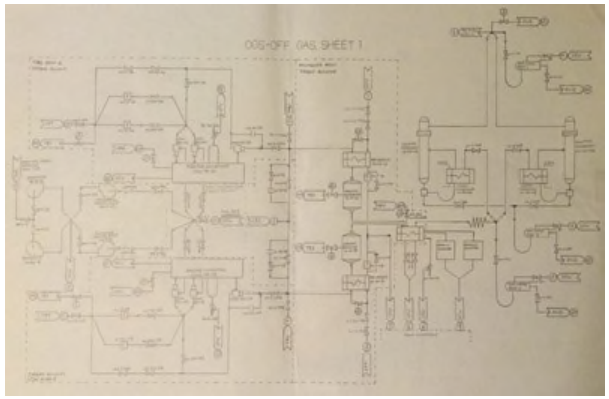
Process Mapping

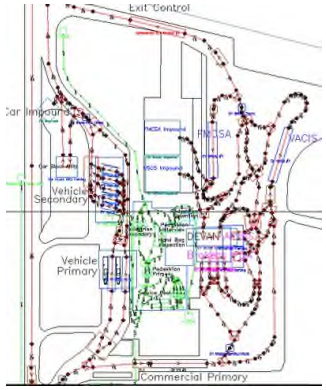
- Describes what comes in, where it goes, how it's transformed, and what comes out.
- Describes the movement and storage of information, material, people, and other entities.
- Maps define the scope of a process. Links to connected processes or "everything else" are called interfaces.
- Are presented at the level of detail that makes sense.
- Process elements within maps can themselves be processes with their own maps.
- State, timing, and other data can be included.
- Entities in process can be split and combined.
- Processes and entities may be continuous or discrete.

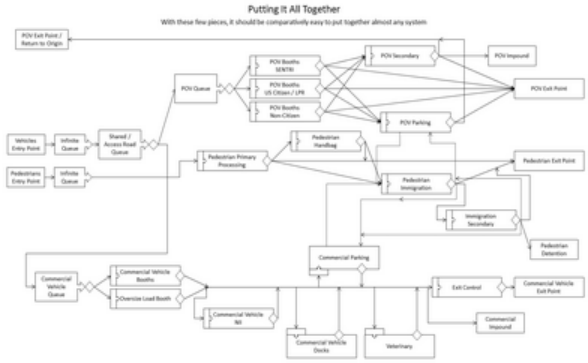
Process Mapping (continued)

Processes may be mapped differently based on needs, industry standards, and the information to be represented.



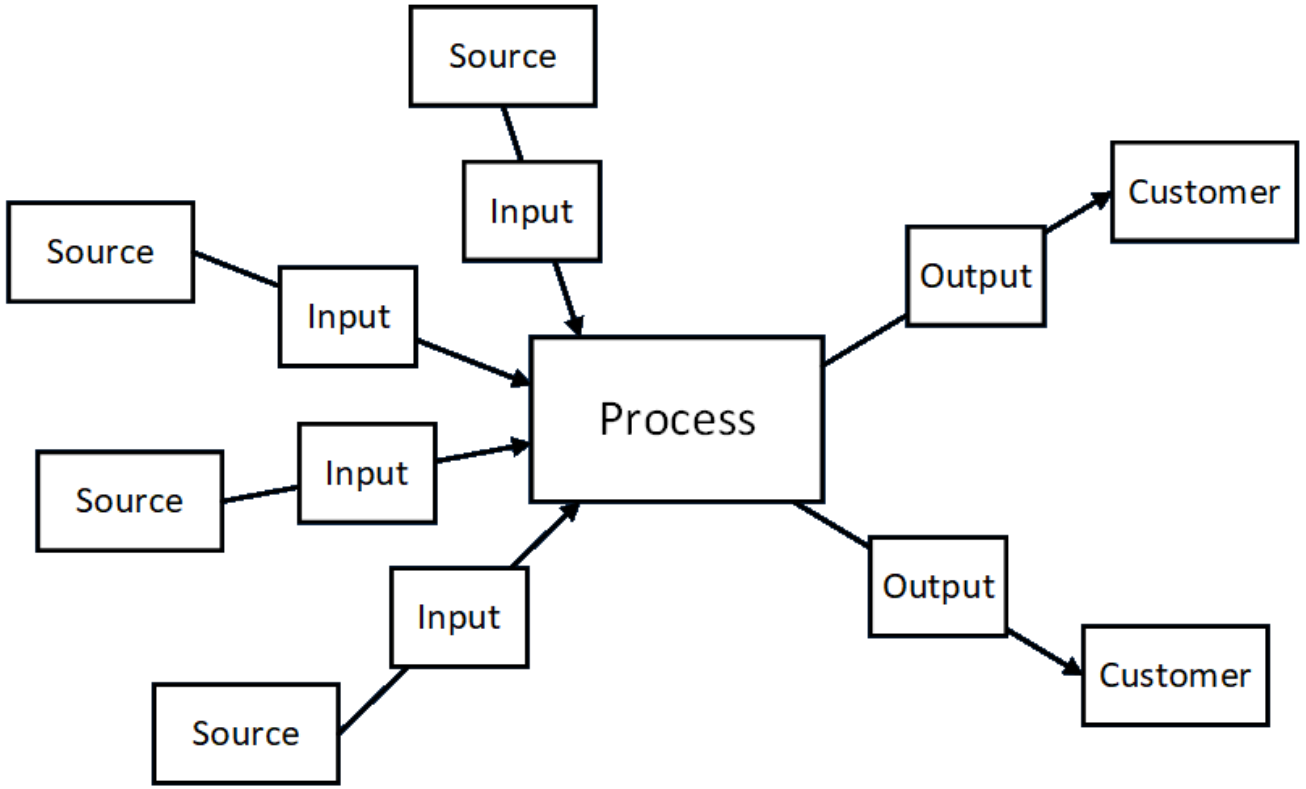






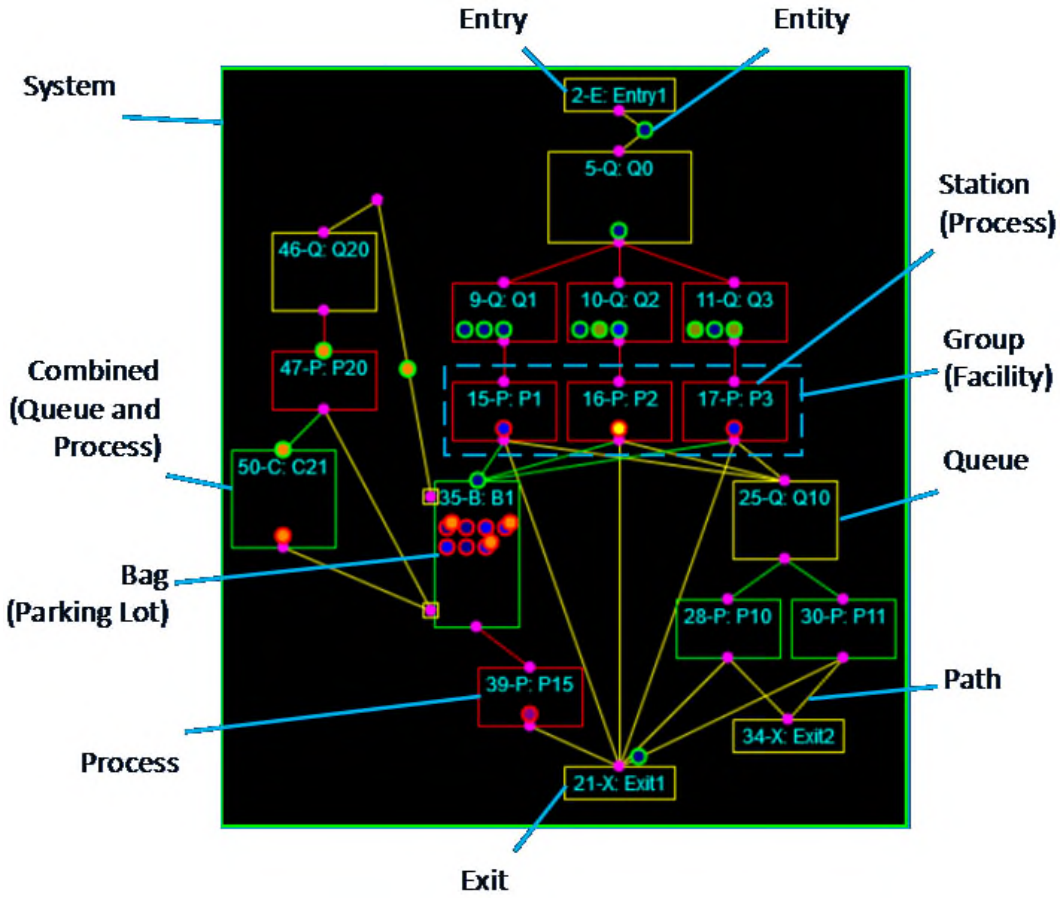
Process Mapping (continued)

- S-I-P-O-C vs. C-O-P-I-S
- Any number of inputs and outputs are possible.



Process Mapping (continued)

I give specific names to modular components.



Data Collection (Process Characterization)

- Captures qualitative descriptions of entity types and characteristics, process types and characteristics, and decisions made.
- Captures quantitative data:
  - physical dimensions, volumes, and storage capacities
  - arrival and departure rates and times
  - diversion percentages (what parts of outputs go where)
  - process durations
  - whatever is needed to describe transitions
  - counts or quantities of what's stored
  - velocities, frequencies, and fluxes
  - number of stations in each sub-process

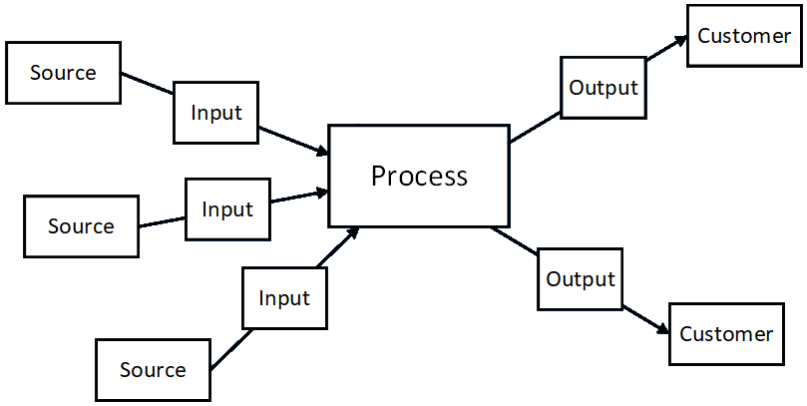
Data Collection (Process Characterization) (continued)

Data collection corresponds to the Observation technique in the BABOK. Methods include:

- Walkthroughs: guided and unguided (Waste Walk)
- Drawings, Documents, Manuals, Specifications
- Electronic Collection (real-time vs. historical, integrated vs. external sensors)
- Visual / In-Person (notes, logsheets, checklists, mobile apps)
- Interviews (with SMEs)
- Surveys
- Video
- Photos
- Calculations
- Documented Procedures and Policies

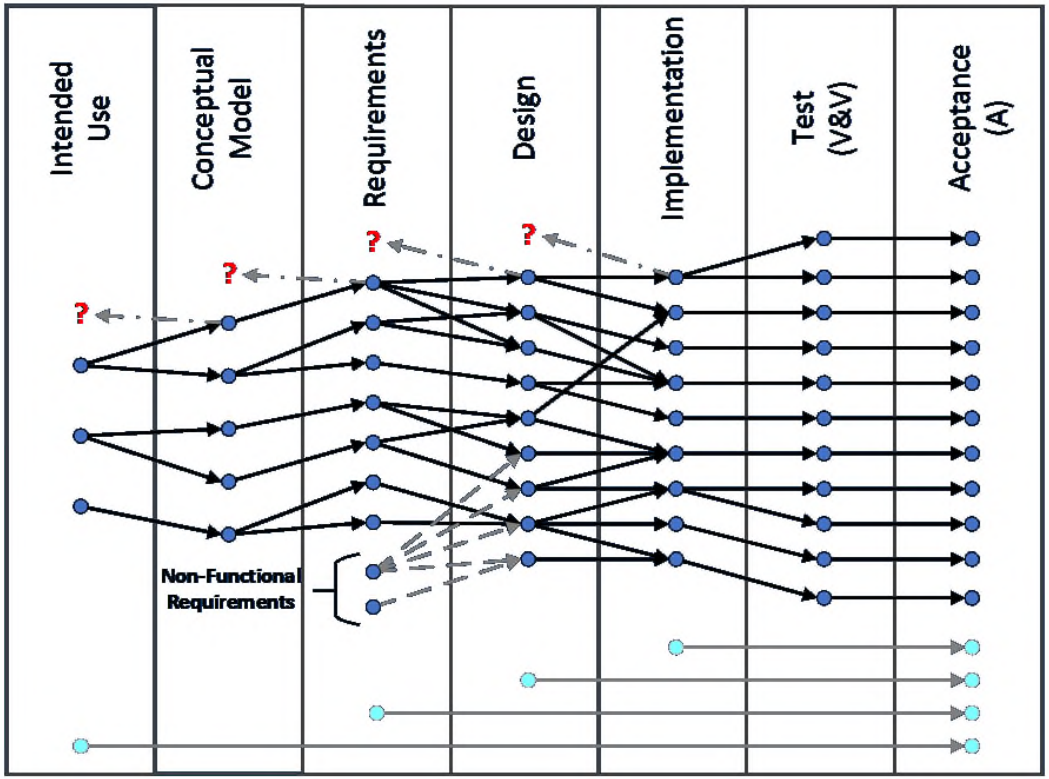
Domain Knowledge Acquisition

- Domain knowledge is acquired from prior experience or training or from the process SMEs as you go.
- What you need to know to:
  - capture process details
  - analyze the operations
  - perform calculations
  - make sure you don't miss anything



Requirements Traceability Matrix

- Best represented by a relational data structure but any method will work.
- Needs discovered later could require the creation of earlier elements.



Customer Feedback Cycle

Review every phase with the customer until they agree it's correct, especially for the Conceptual Model (As-Is State).

Needs discovered later could require the creation of earlier elements.

This framework is usable for both Waterfall and Scrum, and really for any management style.

Example Simulation

Simple pass-through simulation using basic component types.

Independent link

Project link

Scope and Scale of the Project/Simulation Effort

The process described includes the greatest amount of detail. All projects and efforts perform all of these steps implicitly, but some may be streamlined or omitted as the size of the effort scales down.

The entire process takes place implicitly or explicitly even if any one participant or group only sees a small fraction of the activity.

Context of the Project/Simulation Effort

Appropriate to processes that have a defined beginning and end. Not for ongoing, reactive, support and maintenance operations.

Works in Waterfall, Agile, and other project management environments.

The framework does not inhibit the ability to explore and test multiple options.

IVV&A Inspiration for the Framework

Tell 'em what you're gonna do. Do it. Tell 'em what you did.

Accreditation Plan	**V&V Plan**	**V&V Report**	**Accreditation Report**
Executive Summary	Executive Summary	Executive Summary	Executive Summary
1. Problem Statement	1. Problem Statement	1. Problem Statement	1. Problem Statement
2. M&S Requirements and Acceptability Criteria	2. M&S Requirements and Acceptability Criteria	2. M&S Requirements and Acceptability Criteria	2. M&S Requirements and Acceptability Criteria
3. M&S Assumptions, Capabilities, Limitations, and Risks & Impacts	3. M&S Assumptions, Capabilities, Limitations, and Risks & Impacts	3. M&S Assumptions, Capabilities, Limitations, and Risks & Impacts	3. M&S Assumptions, Capabilities, Limitations, and Risks & Impacts
4. Accreditation Methodology	4. V&V Methodology	4. V&V Task Analysis	4. Accreditation Assessment
5. Accreditation Issues	5. V&V Issues	5. V&V Recommendations	5. Accreditation Recommendations
6. Key Participants	6. Key Participants	6. Key Participants	6. Key Participants
7. Planned Accreditation Resources	7. Planned V&V Resources	7. Actual V&V Resources Expended	7. Actual Accreditation Resources Expended
		8. V&V Lessons Learned	8. Accreditation Lessons Learned
Suggested Appendices	Suggested Appendices	Suggested Appendices	Suggested Appendices
A. M&S Description	A. M&S Description	A. M&S Description	A. M&S Description
B. M&S Requirements Traceability Matrix	B. M&S Requirements Traceability Matrix	B. M&S Requirements Traceability Matrix	B. M&S Requirements Traceability Matrix
C. Basis of Comparison	C. Basis of Comparison	C. Basis of Comparison	C. Basis of Comparison
D. References	D. References	D. References	D. References
E. Acronyms	E. Acronyms	E. Acronyms	E. Acronyms
F. Glossary	F. Glossary	F. Glossary	F. Glossary
G. Accreditation Programmatics	G. V&V Programmatics	G. V&V Programmatics	G. Accreditation Programmatics
H. Distribution List	H. Distribution List	H. Distribution List	H. Distribution List
	I. Accreditation Plan	I. V&V Plan	I. Accreditation Plan
		J. Test Information	J. V&V Report

Process Improvement

- Incremental improvement vs. Quantum Leap (The Big Kill!)
- Center to target and reduce variation (Six Sigma)
- Rearrangement and Compression (Lean)
- Substitution / Elimination / Automation
- Modify a sub-process and see how it affects the whole system
- Theory of Constraints: The Five Focusing Steps:
  - Identify the constraint
  - Exploit the constraint
  - Subordinate to the constraint
  - Elevate the constraint
  - If constraint is "broken" go back to step 1

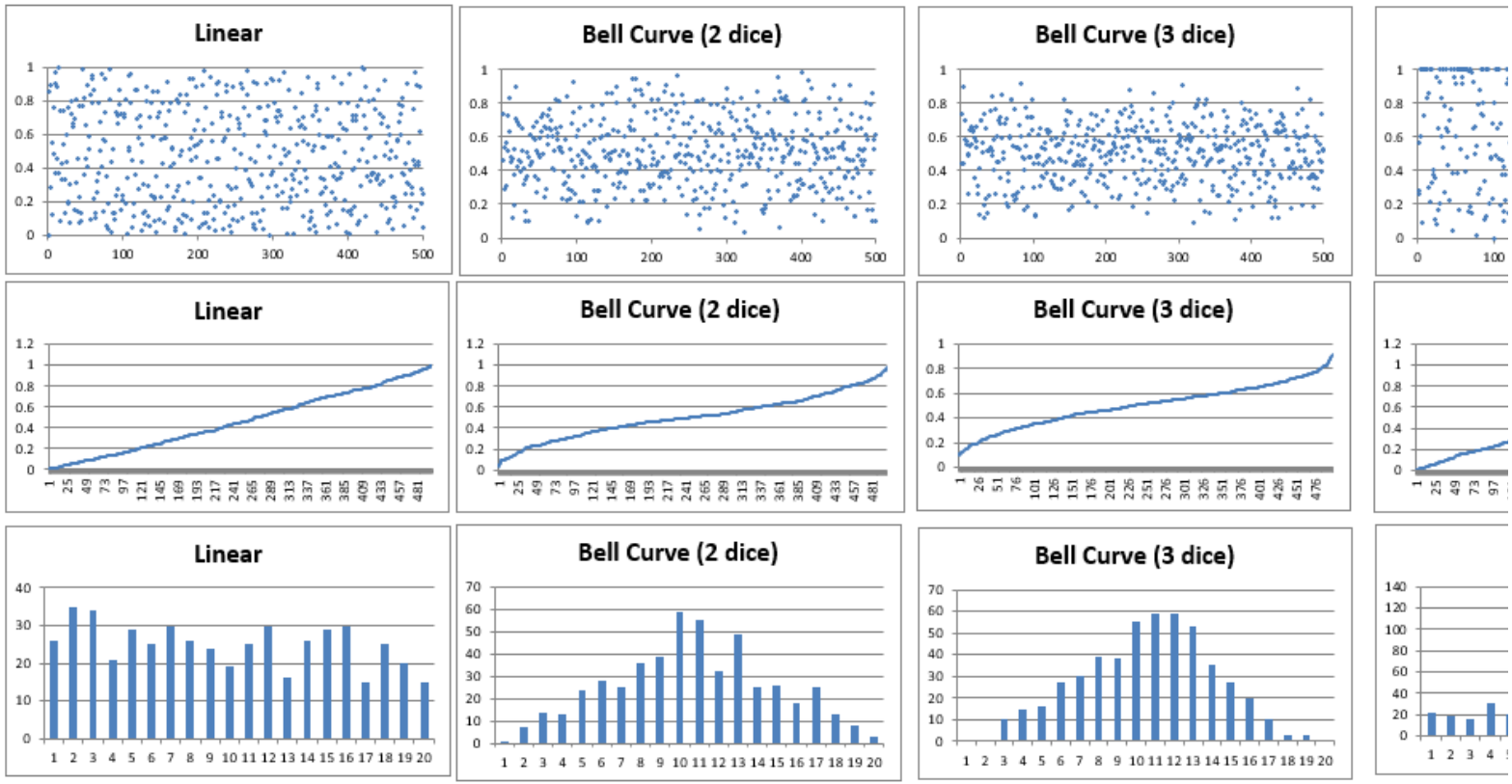
Monte Carlo Analysis **"It was smooth sailing!" vs. "I hit every stinkin' red light today!"**

Involves running multiple trials of complex models including combinations of numerous randomly generated outcomes that yield a range of complex results.

Models may incorporate scheduled and unscheduled elements.

- Randomly generated outcomes may include:
  - event durations
  - process outcomes
  - routing choice
  - event occurrence (e.g., failure, arrival; Poisson function)
  - arrival characteristic (anything that affects outcomes)
  - resource availability
  - environmental conditions
- Random values may be obtained by applying methods singly and in combination, which can result in symmetrical or asymmetrical results:
  - single- and multi-dice combinations
  - range-defined buckets
  - piecewise linear curve fits
  - statistical and empirical functions
  - rule-based conditioning of results

Examples of Data Driving Random Outcomes



Source Spreadsheet

This presentation and other information can be found at my website:

rpchurchill.com

E-mail: bob@rpchurchill.com

LinkedIn: linkedin.com/in/robertpchurchill