Washington DC Chapter of the
System Safety Society

2008 System Safety Training Symposium

College of Southern Maryland

La Plata Maryland

Tutorial

Investigation Lessons Learning System Flaws and Fixes

Ludwig Benner Jr and
William D. Carey

This is a B2 Spirit Bomber - cost estimates range from 1.4 - 2.2 billion depending on who you talk to. Most technologically fantastic ever &c. Flies from Missouri, can hit targets anywhere. We had 21 of them.




May have seen it on the news. Pilots eject in frame 3. Plane go boom.

Now we have 20 bombers.

"This technique was **never formalized** in a technical order change or captured in **'lessons learned' reports**. Hence, only some pilots and some maintenance technicians knew of the suggestion," according to Carpenter's executive summary of the accident.

The report said, "The human factor of **communicating critical information** was a contributing factor to this mishap."

From Maj. Gen. Floyd L. Carpenter, who headed an accident investigation board. (AP News)

What we're going to talk about today:

- What are lessons learned?
- What are mishap lessons learned systems?
- How do you evaluate them?
- How can you make them better?

What are Lessons Learned?

Let's define our terms.

"Lessons learned" are knowledge gained from investigations to enable improved future performance.

The scope of lessons learned activities is everything from the capture to the assimilation of those lessons.

What do we mean by Lessons Learned?
Actually, pretty general agreement on **meaning** of term - but abstraction masks differences in the actual lessons developed in practice, as we'll see.
Focus of tutorial is on lessons learned during mishap investigations - both **before and after they happen,** so think hazard and risk analyses which are "pre-mishap" investigations, as well as post-mishap investigations.
Both provide "new" knowledge from experience.
Scope of tutorial includes everything from the development of a lesson learned to the ultimate verification that a change inspired by the lesson learned has produced the desired effects.

What are Lesson Learned Systems?

They are the systems that produce the lessons learned from mishaps.

Some examples:

- FAA – ASIAS
- NASA – ASRS
- DOD – CELL
- NIOSH (Medical Devices)
- OSHA – Safe Tank Alliance
- USGS (Earthquakes)
- Wildland Fire Lessons Learned Center
- Army – CALL
- DoE – SELLS
- DoT – RITA

Here are just a few examples of Lessons Learned systems, all of which strive to improve safety performance, based on mishap experience through the acquisition and processing of Lessons Learned.
ASIAS = Aviation Safety Information, Analysis and Sharing Center
NASA - ASRS = Aviation Safety Reporting System
DOD - CELL Center for Engineering Lessons Learned
Army - CALL - Center for Army Lessons Learned
DoE - SELLS = Society for Effective Lessons Learned Sharing
DoT - RITA Research and Innovative Technologies Lessons Learned reports for programs

What are the elements of your current **mishap investigation** lessons learned activities and practices? For example

- Identification
- Capture
- Dissemination
- Assimilation

Lets take a look at some practices that constitute the system within your organizations.
What are the components or elements that make up an **investigation** lessons learned System that you can discuss - your own, preferably, but anybody else's you care to mention.

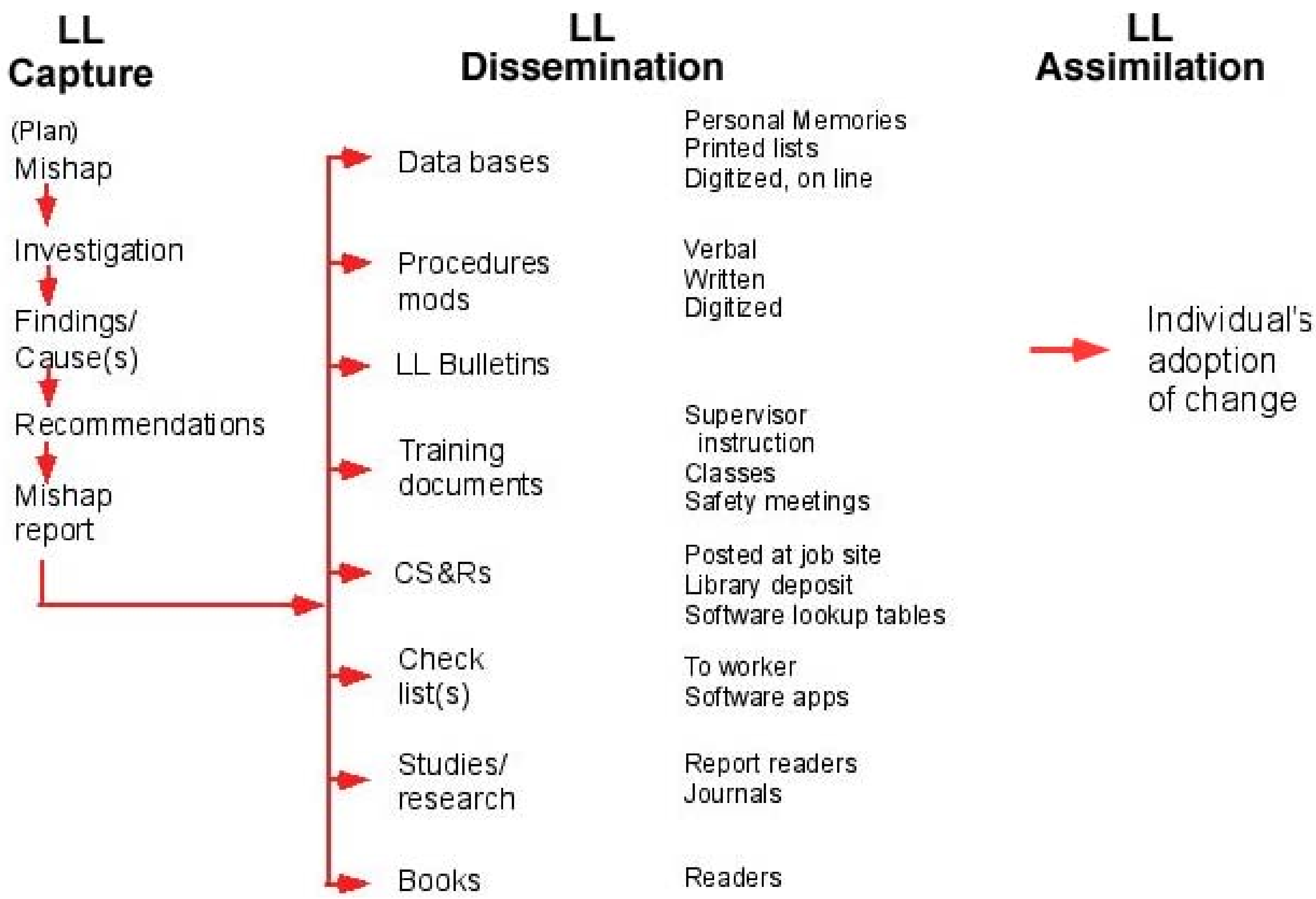
First, lets try to walk through the system, starting with the occurrence of a mishap
After that, we'll do the same for a hazard or risk analysis system - how are lessons learned developed and handled there?

What are the elements of your current
hazard and risk analysis
lessons learned activities and practices?

- Identification
- Capture
- Dissemination
- Assimilation

In our view, hazard analyses are in reality mishaps investigations before the happen, in our view. You just hypothesize what can happen rather than reconstructing what did happened.
Walk through each step of your hazard and risk analysis lessons learned practices like you just did post-mishap investigations.
Introduces you to an approach for defining your predictive Lessons Learned "system"
Could spend hours on this little hummer - but we won't
(Tutorial offers approach and questions you can take home)

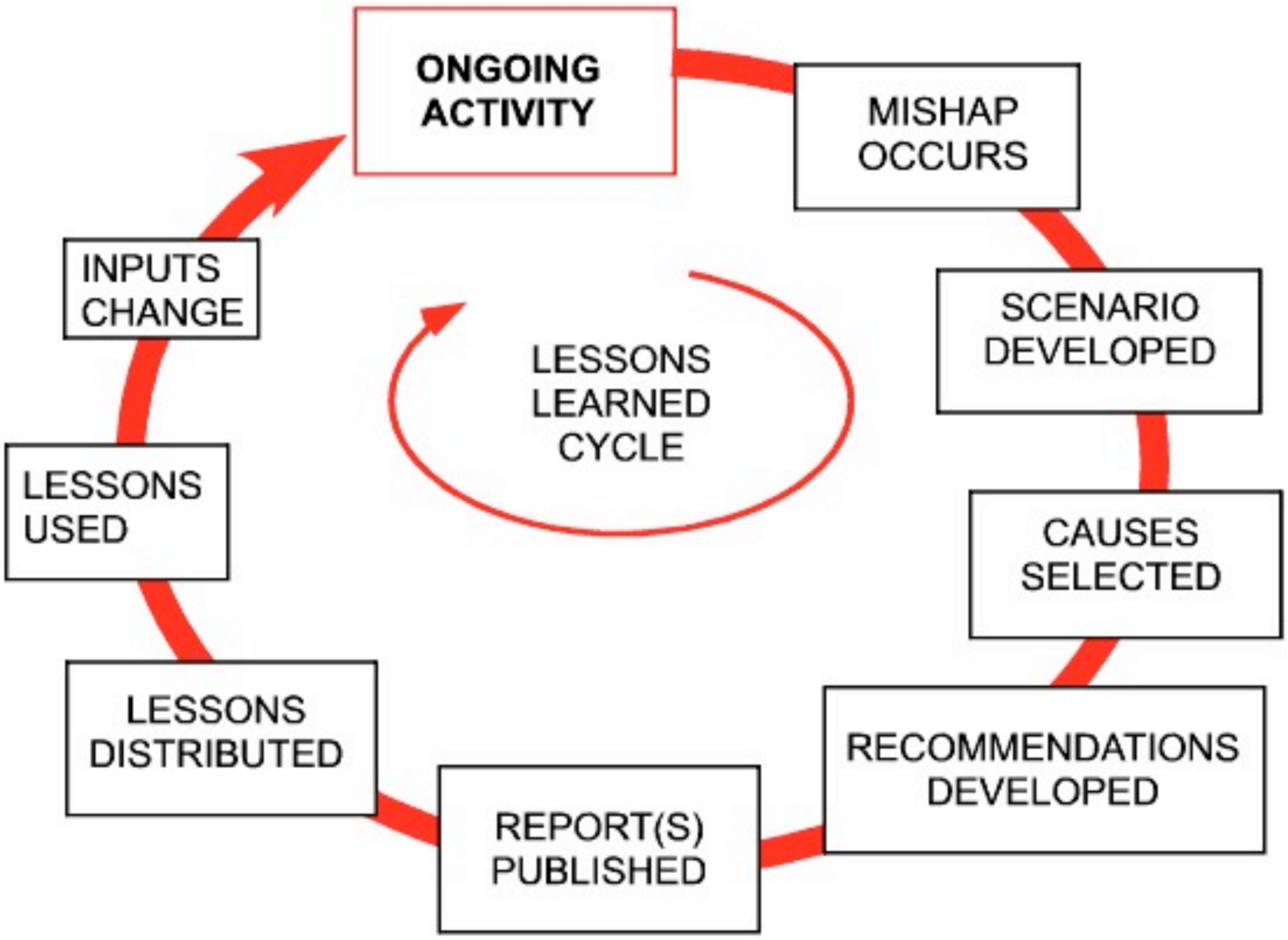
Current Practices



Here's a general representation of current mishap lessons learned system practices we put together before today, and how they fit together.

Note that there's a lot of action between the occurrence of the incident and the final assimilation of lessons learned.

Lessons Learned Cycle



Now, if we look at an ongoing activity, the lessons learned system needs to be continuous if the organization is to be a "learning organization" of the kind envisioned by Senge in the Fifth Discipline: The Art and Practice of the Learning Organization. That's an organization that joins adaptive learning with "generative" learning to enhance its capacity to create its future.

The mishap lessons learning system must be a continuous "loop" where experience changes inputs to the ongoing activity as quickly as it becomes available.

This graphic illustrates the nature of present mishap lessons learned cycles.

View of Lessons Learning System in a Learning Organization

Viewed this way, focus of investigation is to provide the new knowledge that will enable a learning organization to bring about changed behaviors, rather than determination of causes, root causes, probable causes, proximate causes, remote causes, or cause factors, or recommendations.

Lessons learned from each source must be compatible to support a true Learning Organization's needs

Whom should system serve?

Lessons Learning System

- must be designed and optimized to serve users who can bring about changed behaviors.
- design should not be driven by lessons learned producers' perspectives.

How do you evaluate a lesson learned system?

What are the attributes of an optimum Lessons Learned System, from User's or Customer's perspective?

What attributes are worth measuring and monitoring?

What attributes matter?

- Capture Capability
- Accessibility
- Assimilability (This is an actual word.)
- Feedback
- Rapidity

Here are some examples we think are relevant.

(Possibly digression about Karl Popper)

There are probably more.

Any suggestions?

Capture Capability

- What is system trying to capture?
- How many of the available lessons is system capturing?
- How unambiguously does system document them?

How and from what perspective has what the system is trying to capture been identified and defined?

Have the perspectives and goals been documented?

Is the system capturing all lessons generated by a mishap? Does system provide a way to determine that? (If this isn't possible, is there some benchmarking of how many?)

How concretely does system document lessons it has captured?
Are lessons documented persuasively?
Are the lessons actionable?

Accessibility

- How is a system publishing them?
- How fast can an end user locate them?
- How easy is it to sort wheat from chaff?
- Can actual human beings read them quickly?

You might think accessibility is about making what you capture available. It’s not. That’s producers’ perspective.

It’s much more, from an individual user’s perspective.

Users must be able to locate a source of relevant lessons quickly and easily. Are they being published so users can do that?

It can’t take them forever to do that.

When users find a source, they’ve got to be able to filter lessons quickly to find those that will help them.

All this should require minimal read time for user – they have other things to do, too.

Demands high data density lessons learned data

Assimilability

- Can users relate the lessons learned to their activities?
- Can users relate them to the users' behaviors?
- Do they actually prompt the user to change?

When a user finds lessons, how does a user know they're relevant to what he or she is doing now? (Relevance)

Is the context readily discernible?

Is it an actionable item for them – something they can change in how or what they do in their own tasks or activities?

Will the lessons data actually prompt user to change what they are doing

Will they be there in the future if users want to refresh their memories until the desired behavior is a habituated?

Feedback

- Does system identify if the lesson has been learned?
- Did the changed behaviors produce the desired results?
- Does a system provide feedback on the "lesson learned" itself?

Does system provide feedback about what happened to lessons learned?

We're talking about validation, here – of both the assimilation of the lesson, and the efficacy of the lesson

Rapidity

- What's the latency of the system?
- How rapidly are the lessons disseminated
- How rapidly can the lessons learned be implemented?

How much time elapses between the generation of the lessons by the mishap, and the changes they are intended to achieve?

That's the latency.

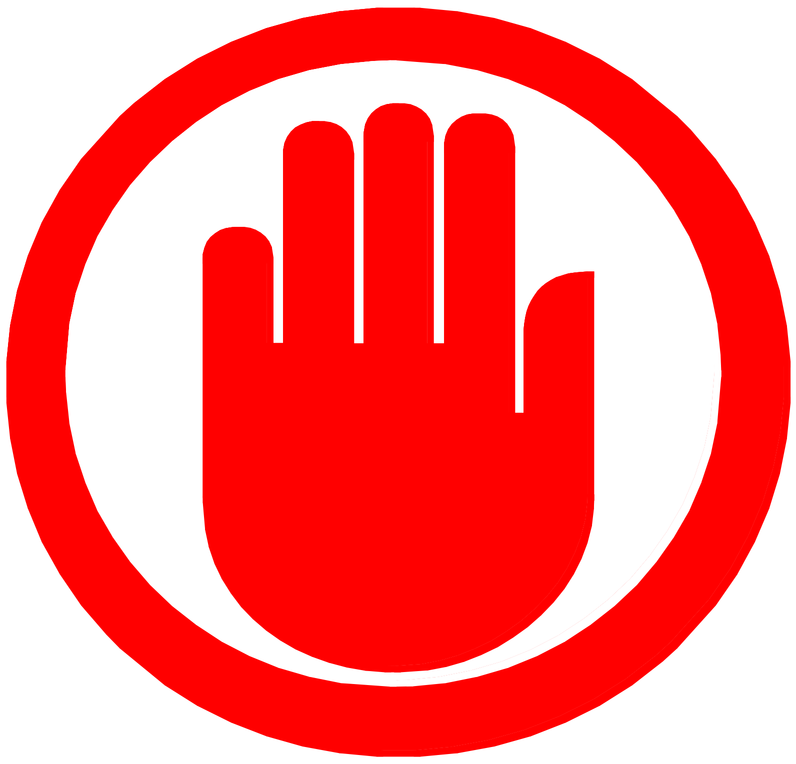
Once identified, how rapidly are the lessons launched on their way to users?

Once accessed, how rapidly can the lessons be implemented by changing what people or objects do?

How quickly can operations be restarted with lessons implemented?

Totally quantifiable.

Break time.
Back in the saddle in 15 minutes.



Review

- We defined a lessons learned system.
- We talked about attributes you can use to evaluate lessons learned systems.
  - Capture, Availability, Assimilability, Feedback, Rapidity

Why are Lessons Learned systems hard to optimize?

- A difficult problem,
- inherited perspectives, and
- murky, incremental benefits
- lead to entrenched legacy systems.

The problem is inherently difficult.

Skepticism greets any changes or alternative systems in part because benefits are unclear.

Any effort to optimize attributes faces constraints that pose challenges
For example, There's **big investment** in present systems, difficult to introduce new stuff.

Present systems conform to **inherited perspectives** of incidents, investigations, and lesson learned.

What are some other constraints might you add?

To modify or supplant legacy systems, something better must be offered.

Optimizing systems is a difficult challenge.

- B-2 Crash on Guam
- Studies like Werner and Perry
- NASA report

We've talked a bit about the B-2.

In their study, Werner and Perry cited a bunch of reasons why lessons learned are underutilized in the aviation community.

"NASA stated that it must do a better job of communicating the various lessons learned sources to employees, improving mechanisms to link these sources, and ensuring appropriate training for employees in order to maximize lessons learning." (United States General Accounting Office (2002) Report to the Subcommittee on Space and Aeronautics, Committee on Science, House of Representatives, NASA: *Better Mechanisms Needed for Sharing Lessons Learned*, GAO-02-195,)

Personal use of public lessons learned data is unknown, quantitatively, but interest in and use to generate new behaviors by individuals seems very limited.

For example, one widely respected and emulated public incident lessons learned database with over 700,000 records (ASRS) had 88 search requests by individuals during a recent six year period.

How many individuals would buy a 334 page, $US 80 book to find lessons learned that might apply to their tasks and then internalize all of them to change their behaviors? How frequently do individuals change their behaviors due to desired interpretations of generalized training, procedures, standards or regulations? We don't know. However, few would argue that present practices maximize investigation lessons learned dissemination and their use by all who might benefit from the data.

Inherited Investigation Perceptions

Based on a Framework with

- Causation orientation
  - aimed at probable cause, root cause, cause factors, etc.
- Legal heritage of practices

  Fact gathering
  > Analysis
  > Findings
  > Conclusions/Causes
  > Recommendations
  > Report
- Value-laden terms
  - failed, failed to, error, caused, etc.
- Natural language

Natural Language Difficulties

- Ambiguity
- Abstractions
- Subjectivity
- The Passive Voice

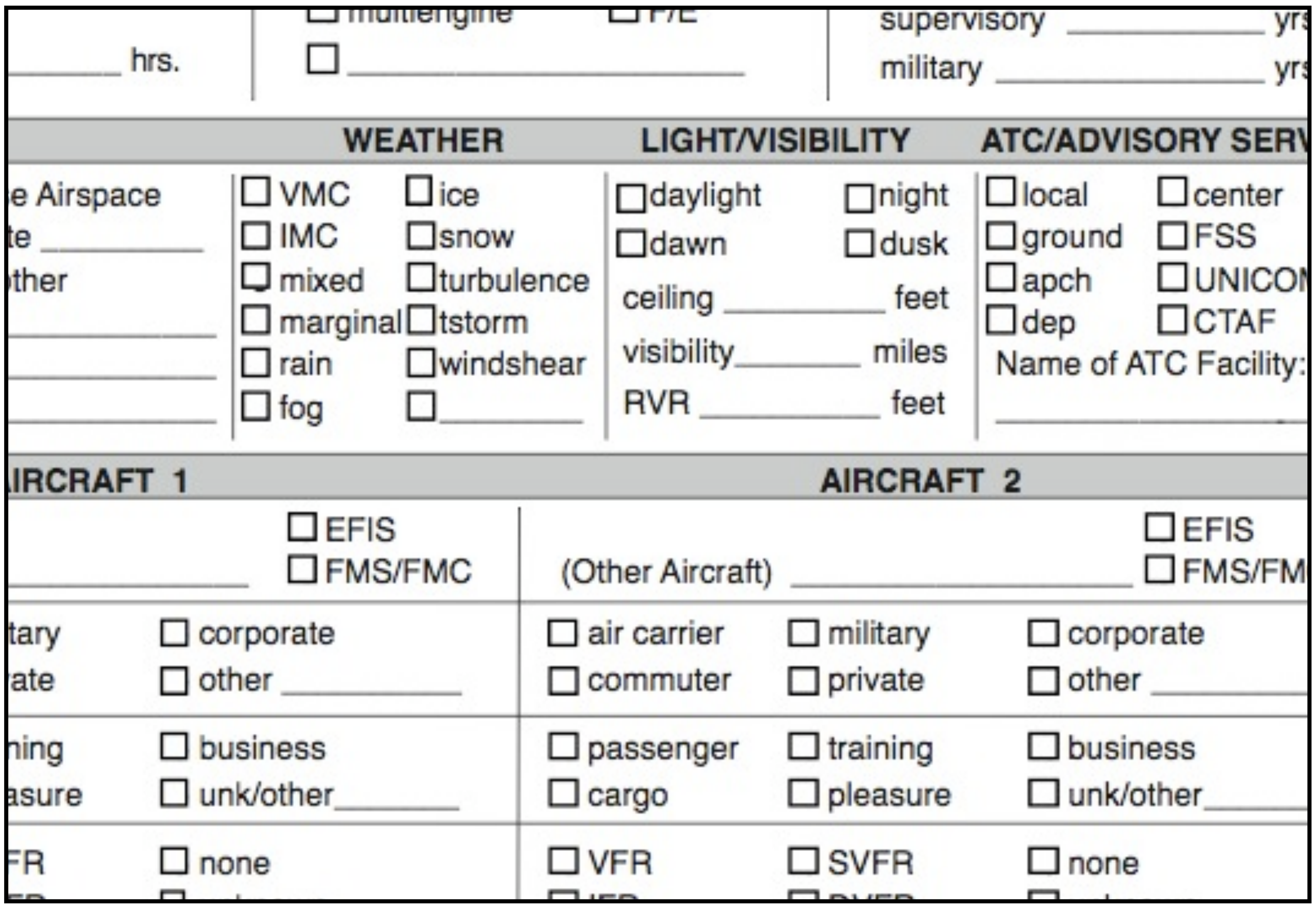
Natural language is a blessing and a curse. It can be very expressive and communicate meaning without being concrete. It can also frustrate the description of people, objects and energies and their actions because of its propensity to use and tolerate ambiguity, the great variety of ways it is possible to express something, value-laden vocabulary, and flexibility of grammar and syntax, for example.

Know about Hayakawa's ladder of abstraction, and the ambiguity introduced as objects rise on the ladder; abstractions can be "cover up" words chosen to cover up lack of specificity of understanding, and thus pose barriers to objective understanding of phenomena, objectively describing and explaining them, identifying lessons learned, and communicating that understanding and the lessons learned.
Descriptions can be enhanced by the definitive vocabulary, grammar, syntax and structure.

The challenge is to work abstraction, ambiguity and subjectivity out of Lessons Learned systems.

And never use the passive voice. Ever.

Overcoming Natural Language Barriers for Static Data:



hrs. ☐ ______ supervisory ______ yrs
military ______ yrs

WEATHER		LIGHT/VISIBILITY		ATC/ADVISORY SERV	
☐ VMC	☐ ice	☐ daylight	☐ night	☐ local	☐ center
☐ IMC	☐ snow	☐ dawn	☐ dusk	☐ ground	☐ FSS
☐ mixed	☐ turbulence	ceiling ______ feet		☐ apch	☐ UNICOM
☐ marginal	☐ tstorm	visibility ______ miles		☐ dep	☐ CTAF
☐ rain	☐ windshear	RVR ______ feet		Name of ATC Facility:	
☐ fog	☐ ______				

e Airspace
te ______
ther

IRCRAFT 1 — ☐ EFIS ☐ FMS/FMC
tary ☐ corporate
ate ☐ other ______
ning ☐ business
asure ☐ unk/other ______
FR ☐ none

AIRCRAFT 2 — (Other Aircraft) ______ ☐ EFIS ☐ FMS/FM
☐ air carrier ☐ military ☐ corporate
☐ commuter ☐ private ☐ other ______
☐ passenger ☐ training ☐ business
☐ cargo ☐ pleasure ☐ unk/other ______
☐ VFR ☐ SVFR ☐ none

We’ve done a lot to overcome NLBs for static data.

- Dictionaries
- ASIAS
- Other Stuff?

Let’s look at the ASRS as an example.

Overcoming Natural Language Barriers for Static Data:

CERTIFICATES/RATINGS

- ☐ student
- ☐ commercial
- ☐ instrument
- ☐ multiengine
- ☐ ____________________
- ☐ private
- ☐ ATP
- ☐ CFI
- ☐ F/E

ATC EXPERIENCE

☐ FPL ☐ Developmental

radar ______________ yrs.

non-radar ____________ yrs.

supervisory __________ yrs.

military ______________ yrs.

WEATHER

- ☐ VMC
- ☐ IMC
- ☐ mixed
- ☐ marginal
- ☐ rain
- ☐ fog
- ☐ ice
- ☐ snow
- ☐ turbulence
- ☐ tstorm
- ☐ windshear
- ☐ _______

LIGHT/VISIBILITY

☐ daylight ☐ night

☐ dawn ☐ dusk

ceiling ________ feet

visibility______ miles

RVR _________ feet

ATC/ADVISORY SERV.

- ☐ local
- ☐ ground
- ☐ apch
- ☐ dep
- ☐ center
- ☐ FSS
- ☐ UNICOM
- ☐ CTAF

Name of ATC Facility:

____

AIRCRAFT 2

☐ EFIS

_ ☐ FMS/FMC

orporate

ther _________

(Other Aircraft) _________________ ☐ EFIS ☐ FMS/FMC

☐ air carrier ☐ military ☐ corporate

☐ commuter ☐ private ☐ other _________

Overcoming Natural Language Barriers for Dynamic Process Data:

"Keeping in mind the topics shown below, discuss those which you feel are relevant and anything else you think is important. Include what you believe really caused the problem, and what can be done to prevent a recurrence, or correct the situation. (USE ADDITIONAL PAPER IF NEEDED)"

- How the problem arose
- How it was discovered
- Perceptions, judgments, decisions
- Actions or inactions
- Contributing factors
- Corrective actions
- Factors affecting the quality of human performance

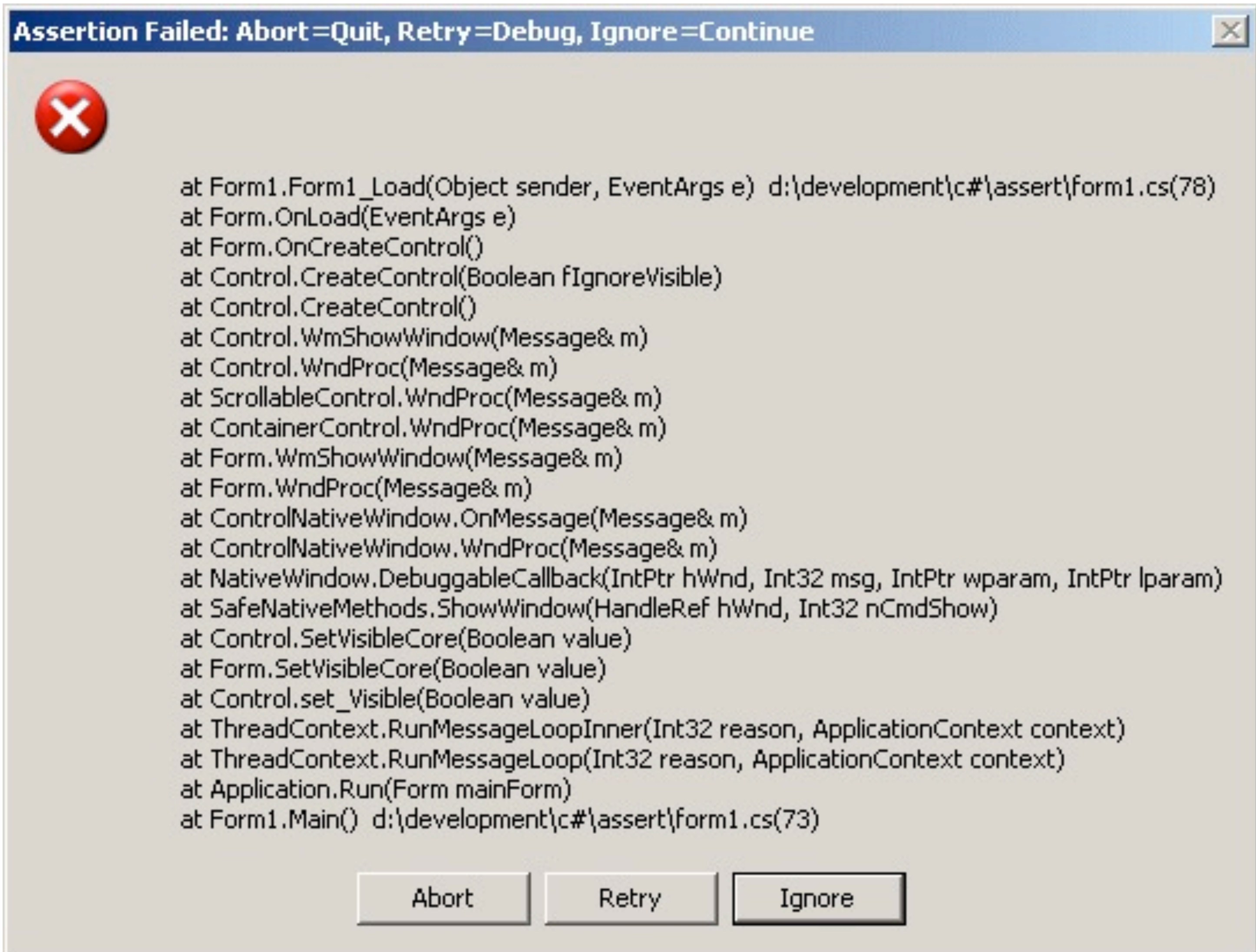
What would a language for dynamic process description look like?

- Describes Interactions
- Consistent Structure
- Consistent Vocabulary
- Concrete, value-free terms

SSDC Washington DC July 2007

It's going to look more like code than like english narrative.

Computer programmers have thought about this problem.



Programmers don't want a narrative explanation of what went wrong. They want a stack trace.

Programmers have developed language (and code) to capture and record "mishaps".

What data? Why? Formats and structure?

Define, Identify, Document, Validate?

Structural language vs dictionary approach – dictionary fine for objects and their description

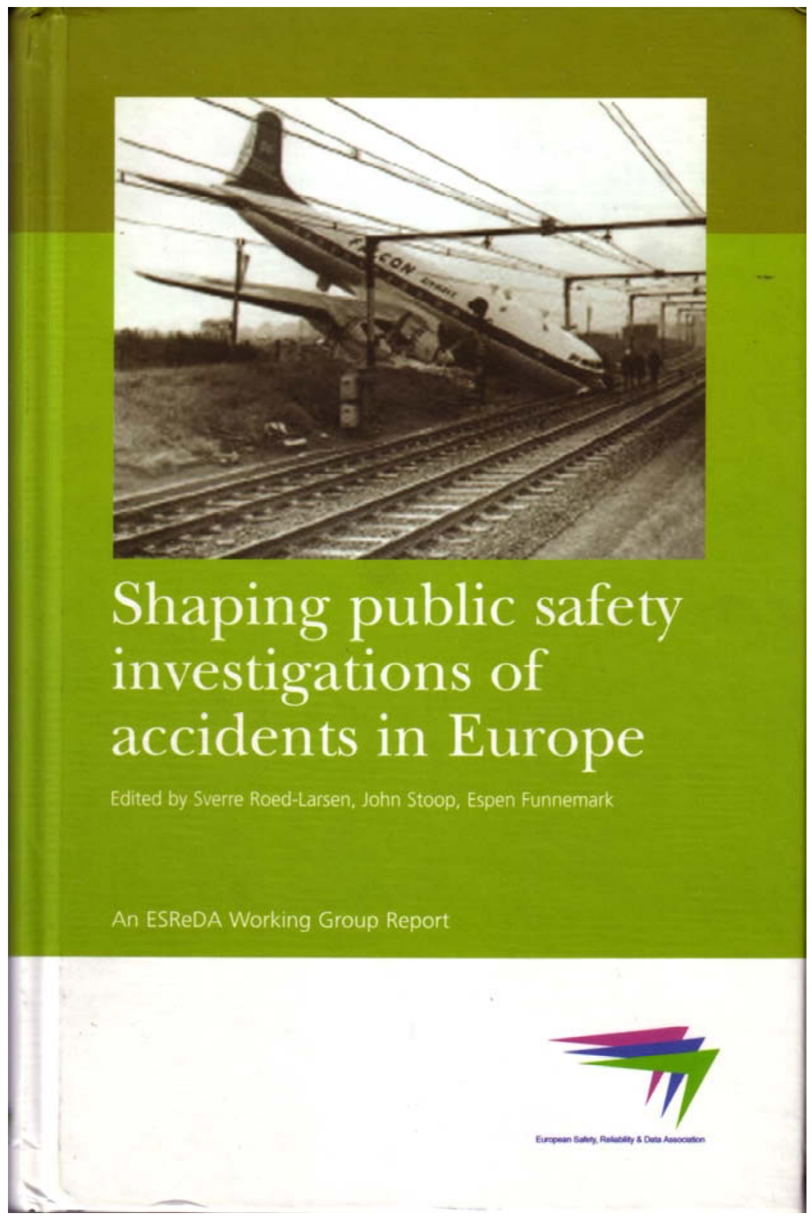
Static data relatively easy to define, and being done.

Process data a little trickier.

Emphasis on Behavior

"…a decomposition of the dynamic behaviour of systems and their actors benefits from a behavioural approach, decomposing the system into events, actions, decisions, errors and scenarios."

From ESReDA Working Group Report (2005) "Shaping public safety investigations of accidents in Europe" p 35



People bring about change

Focus on behaviours - who does what, when, where?

Note the third line -
The focus on behaviors reported in our paper is not particularly original - the ESReDA working group recognized the benefits of the behavioral approach to decomposing the system in 2005.
Our experiences support the benefits envisioned.

Investigation Building Blocks

- Investigators use "building blocks" to reconstruct mishap scenarios
-

Example of Structured Behavioral Building Block

Figure 3. Investigation Building Block Elements in XML Document *

```
<?xml version="1.0"?>
<mesblock unique_id=""> (9)
        <actor></actor>  (1)
        <action></action> (2)
        <object></object> (3)
        <location></location> (4)
        <start_time type=""> (6)**
                <year></year>
                <month></month>
                <day></day>
                <hour></hour>
                <minute></minute>
                <second></second>
                <millisecond></millisecond>
        </start_time>
        <end_time type="">  (7)
                <year></year>
                <month></month>
                <day></day>
                <hour></hour>
                <minute></minute>
                <second></second>
                <millisecond></millisecond>
        </end_time>
        <source></source>  (5)
        <remarks></remarks> (8)
        <nstest> </nstest>***
        <link></link>****
</mesblock>
```


Here's an example of a well defined structure for a behavioral building block using data tags (This is in use)

Times permit ordering of behaviors

Enables investigators to couple behavioral inputs and outputs

Structure helps reduce natural language problems of ambiguity, abstraction and inconsistency

Readily expandable to accommodate addition data elements if needed
enhances machine interoperability

Getting the Lessons to the right people.

- Who are the right people?
- What media do you use to deliver them?
- Do you push or pull?

We've said that accessibility isn't about getting the data out there. Anyone can publish anything. 6 million kitten blogs.

Lots of different media…

Databases

CS&Rs

Procedure Mods

Checklists

Lesson Learned Bulletins

Studies and Research

Training Documents

Books

Current accessibility diffuse, mixture of media and content

Difficulty for users is finding the right ones.

Structured Data
=
More Accessibility

But only for the static data. ASRS does this well.

For dynamic data? Harder? Hamfisted right now?

How do you solve that problem?

Develop Input/Output Relationships

- During investigations, pursue behaviors
- Identify and define necessary and sufficient behavioral <u>inputs</u> to each behavior
- Identify subsequent behaviors induced by each behavior during mishap
- Couple inputs and outputs to each behavior to produce dynamic description

The coupling of the inputs, behaviors and outputs produces behavioral sets which can be manipulated by machine. Those sets can provide context for lessons learned.

This lets you do a more advanced search.

Capture the static attributes related to the incident, as in the ASRS system, for statistical analyses and potential search keys.

Capture the dynamics with coupled structured data elements.

Exploit machine readability

Structured Behavioral Building Blocks could

- improve data entry consistency.
- enhance machine parsing capability.
- enhance data aggregation, archiving and retrieval.

Making Lessons Learned Assimilable

- Behavioral I/O sets enable high data density, data access and interpretation.
- Arrayed behavioral building blocks show actions needed to produce changes.
- Filters on actors or behaviors speed machine relevance checks.
- Statistical analysis benefits from structured data.

Loop back to data needs perception as user.

Context, relevance, behaviour format, inputs and output coupling.

The assimilability needs to be considered for all levels of individuals in an organization, for the operators who run things to supervisors to trainers to designers to managers and executives who establish a corporate culture

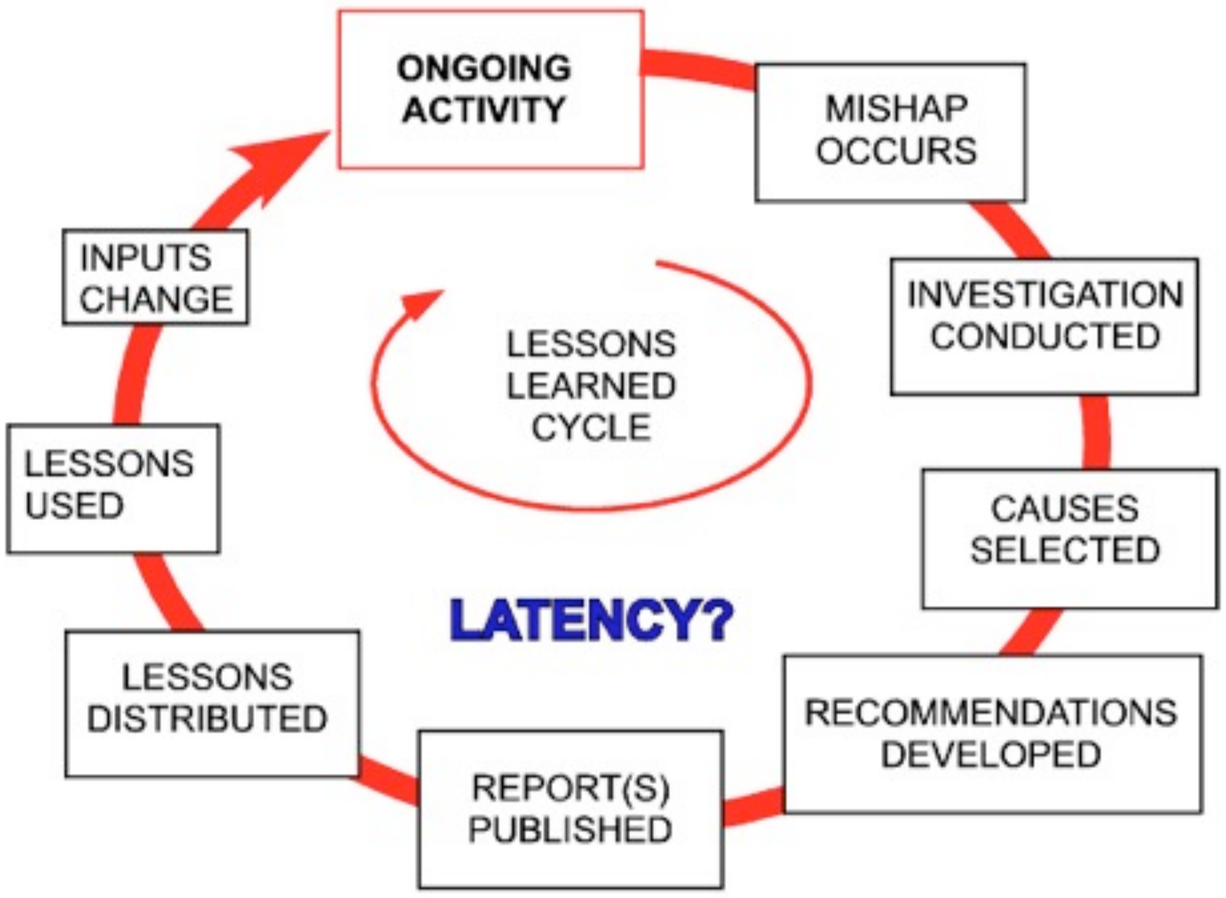
Format that's accessible repeatedly over time.

Getting Useful Feedback

- Part of benefits identification challenge
- Data for measuring "success" of lessons learning system operation
- Intra-system "learning organization" element
- Scope of changes achieved

Systemic Loop
Measure data access
User investment in system
Transition from a just-publish to a more interactive system.

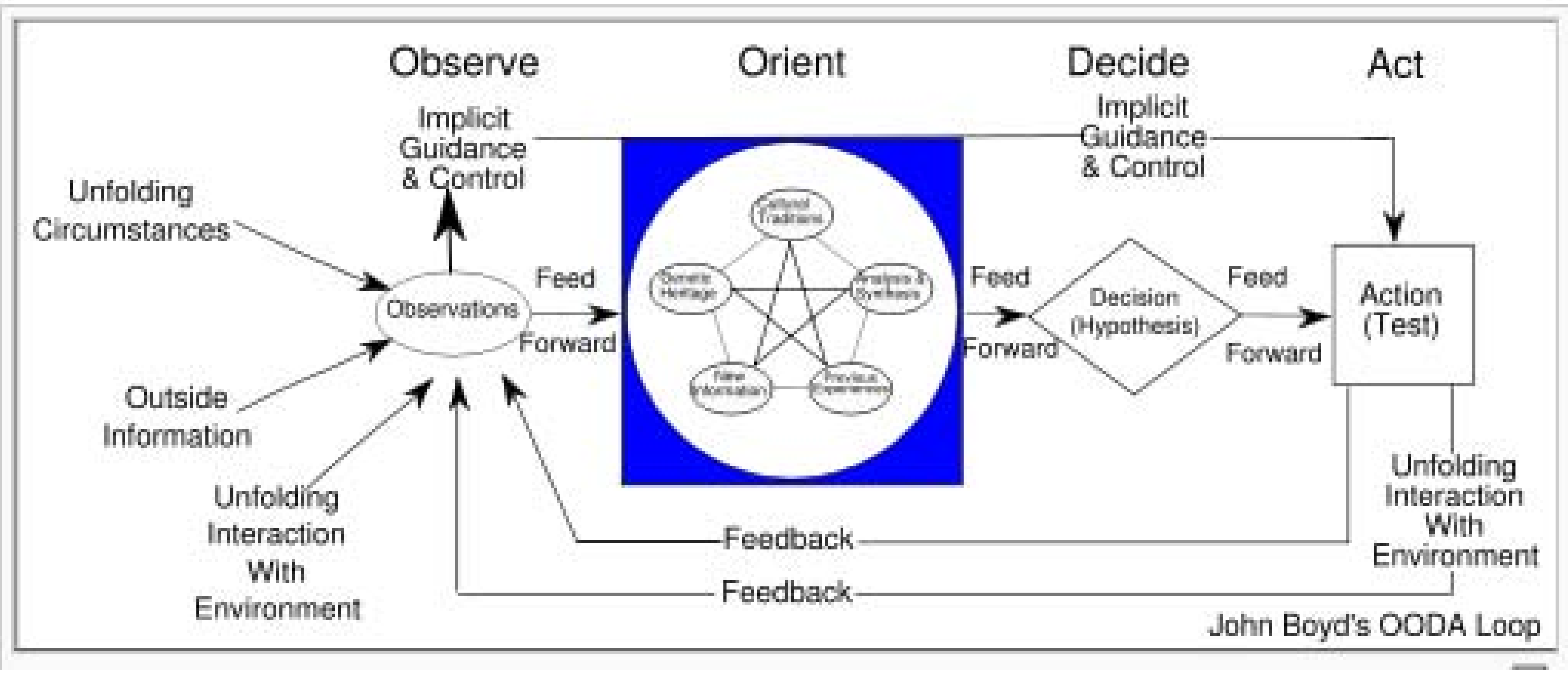
Making it All Go Faster



An important attribute of present Lessons Learned systems is the latency of the Lessons Learned from mishaps.
Latency is the term used to describe the period between the moment something is initiated and the moment its effects begin or become discernible.

In mishaps, Lessons Learned are generated by the mishap. When a mishap occurs, it generates the data needed for Lessons Learned to be derved from the mishap. However, the **elapsed time** between the mishap and the time actual changes, indicated by the Lessons Learned, are accomplished can range from an almost instantaneous reaction, like pulling one's hand back from a hot stovetop, to years! In a sample of the 20 most recent reports published by the NTSB, for example, the latency period until the Lessons Learned were distributed (not implemented) averaged 689 days.

Boyd's OODA Loop



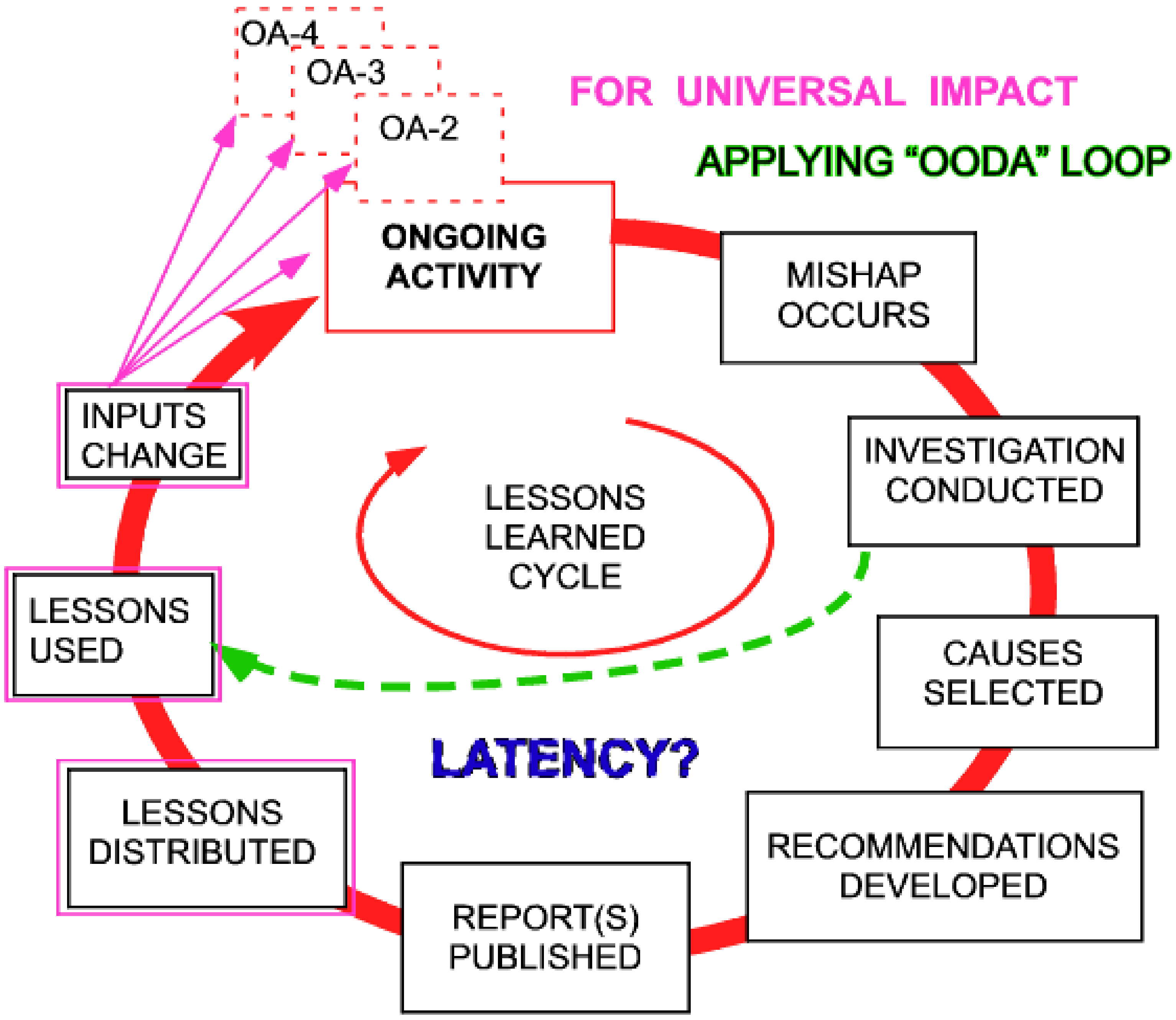
OODA Loop idea as potential strategy for Lessons Learning System improvement strategy. Strategically, the goal to get your OODA Loop inside your enemy's OODA loop, or speed up the decision process.

Idea is to ensure Intervention in Drift to Failure with timely lessons learned before something goes wrong.

Implementation of Senger's Learning Organization concepts.

Strategy to reduce steps and time to intervene with Lessons Learned–
behavior/ input output model.

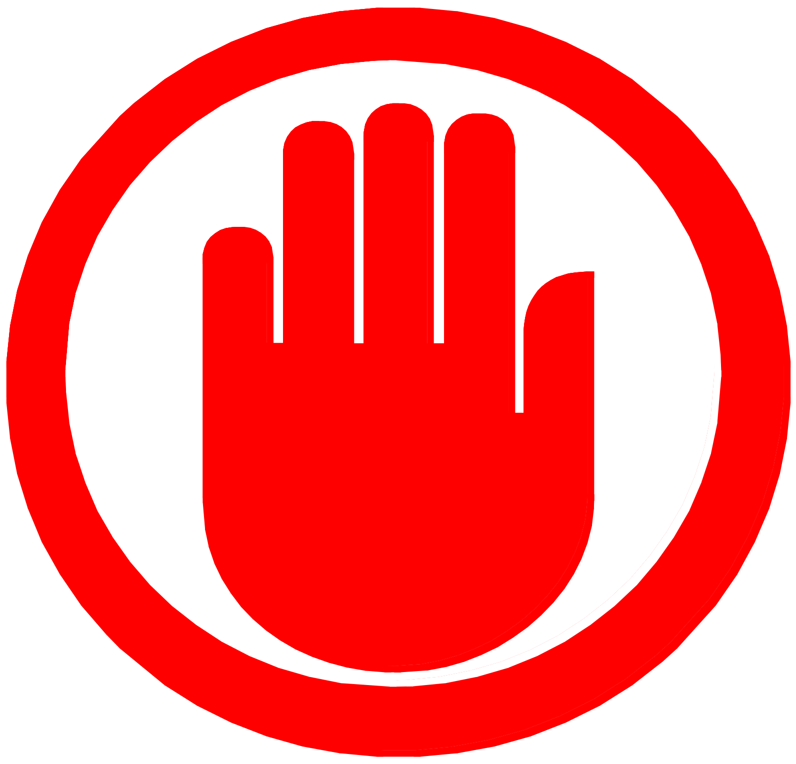
Applying the OODA Loop



Here's how you might illustrate that strategy

(Highlight shortcut from investigation conduct to lessons learned data)

Break time.
Back in the saddle in just 5 minutes.



Review

- Systems must serve consumers
- We offered some reasons creating lesson learning systems is difficult.
- We suggested some examples of ideas that might help optimize lessons learning systems.

What do you look for in investigations?

- Depends on your perspective.
  - Are you a producer of lesson learned?
  - Are you a consumer of lessons learned?
- Should they have the same perspective?

We have observed that what investigators look for during investigations depends heavily on their perspective. try to produce today, e.g. what are LL products

Lessons learned Users are investigator's customers.

By now it's probably appropriate to ask you If you were a "customer" what would you like to get as lessons learned investigation work products?

(try to steer discussion toward BEHAVIOR data they can relate to their own activities and tasks)

The suggestions we've made so far reflect the general systems model approach.

The framework for thinking about investigations

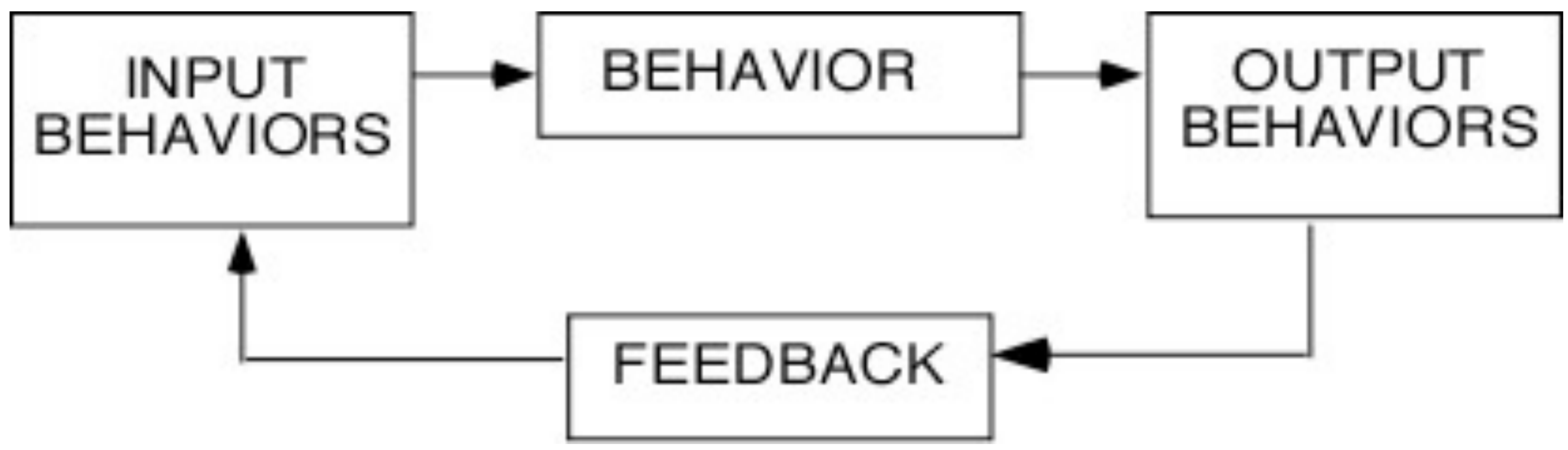
Alternative Framework

General Systems Model

Systems thinking uses the general systems model. This leads to thinking of a lessons learning process and system that produces learning in input/output terms.

That offers an alternative to how we might think about investigations,
when you start thinking about it, it actually offers some attractive benefits, one of which is the substitution of input/output thinking which makes “causal” determination – and all the baggage “cause” carries socially – and uses moot.
Can you imagine a safety community without “cause” and what that might be like?

Adapt Systems Model to behaviors

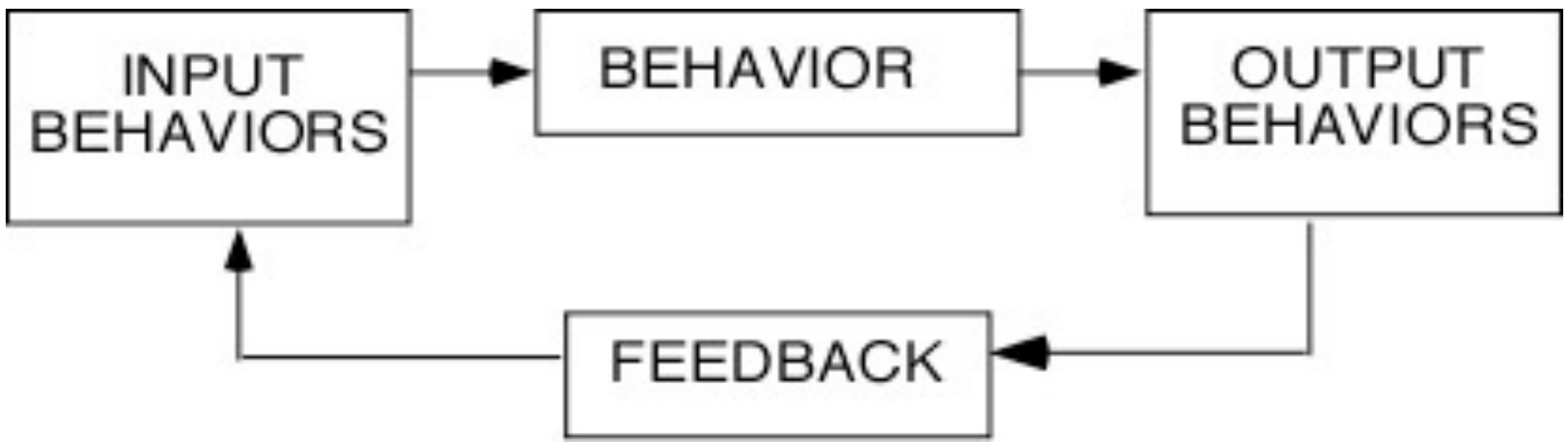


The suggestions actually reflect a modified general systems model, where behaviors are the focus of the model. Acquiring input data as behaviors – or actions – that influenced subsequent behaviors or actions underlies the approach. Determining the input behaviors and coupling them to other behaviors is the key to gaining understanding of what happened.

But what about conditions, you might ask?
Borrowing Newtons ideas, a condition remains static until acted on to change it. Thus you need to track actions – or behaviors – that changed the conditions in the above model.

Suggests alternative framework

Behavioral Adaptation of General Systems Model



Pursue what people, objects or energies DID during incident - e.g., their behaviors

Look for *dynamics* of incident, e.g., BEHAVIORAL INTERACTIONS

Framework can lead to

- Description of behavior flows
- Definition of behavior relationships
- Behavioral I/O data sets
- Capture of every potential Lesson Learned
- Identification of context for Lessons

Behavior flow descriptions:

- Behaviors can be arrayed to show sequence in which they began and ended
- Concurrent behaviors can be displayed in matrixes
- Sequenced arrays can be verified with logic tools

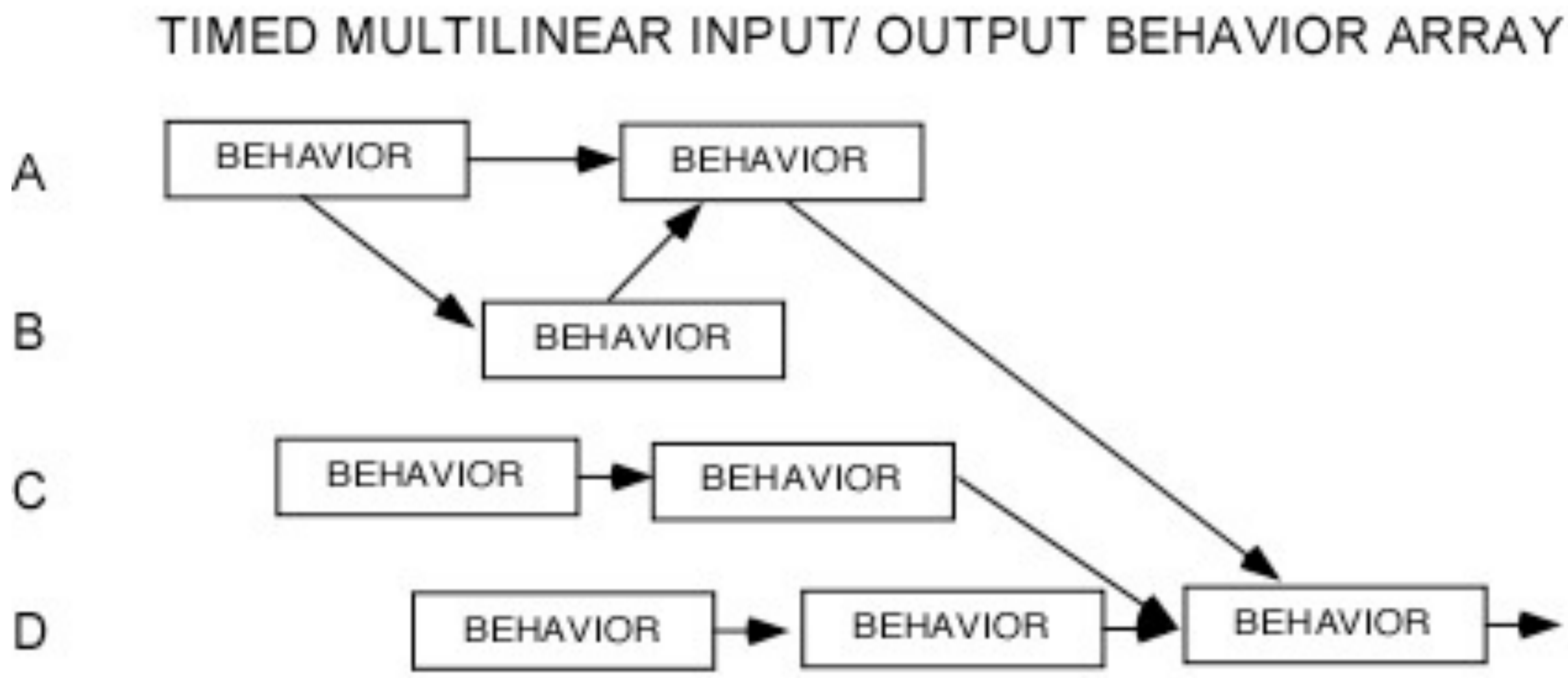
Behavior relationships can be defined

- The input behaviors can be coupled with any specific behavior they induced or influenced
- Any specific behavior can be coupled with output behavior(s) to which it contributed
- The couplings can be verified with logic tools

Coupled Behaviors lead to

- behavior input/behavior sets
- behavior/output sets
- input behavior/behavior/behavior output sets

Process Behavior Description

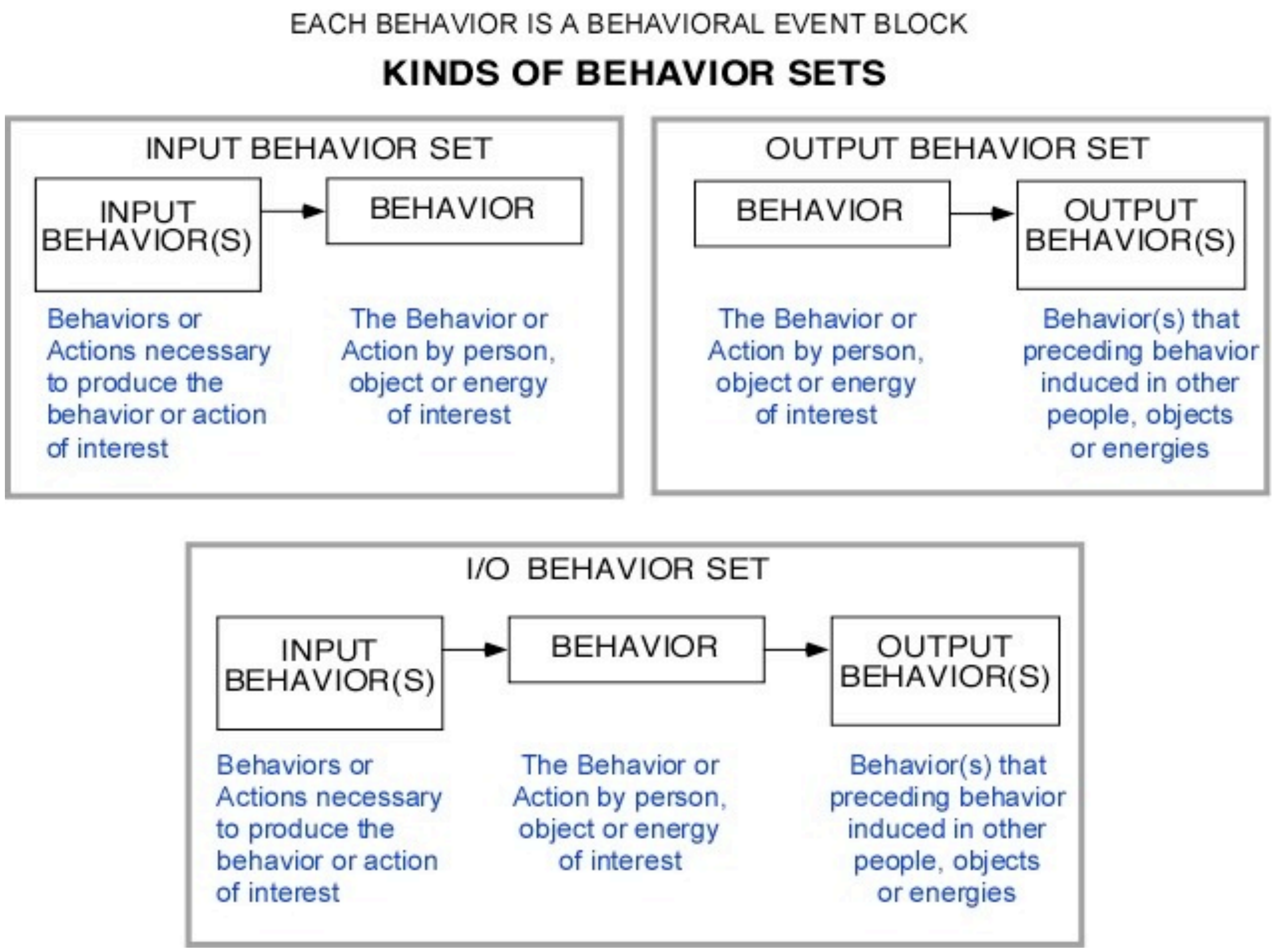


Describes Process Progression

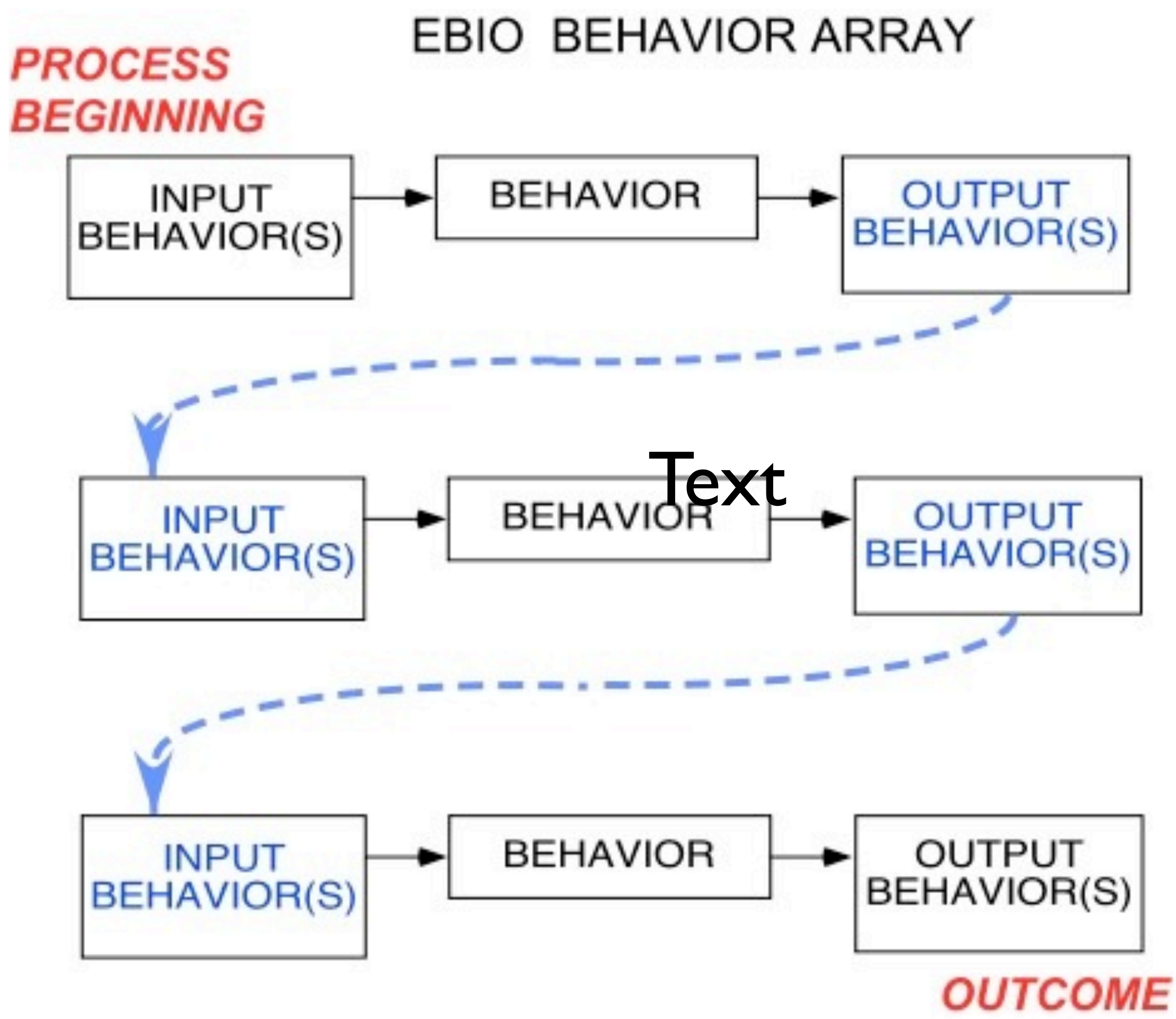
Each I/O data set leads to

- a structured problem definition
- a specific intervention opportunity
- context of problem and opportunity
- maximized lessons learned

Behavior Data Sets



Arrayed I/O Data Sets



Arrayed behavior sets provide individual behavior CONTEXT for all captured Lessons Learned

Patent Pending

Focus on behaviors during investigations

- Pursue what people, objects or energies DID during incident - e.g., their behaviors
- Look for *dynamics* of incident, e.g., BEHAVIORAL INTERACTIONS
- A way to address dynamics of mishaps

With the modified general systems model, we have a way to address to optimization of lessons learning system performance.

Break time.
Back in the saddle in 15 minutes.

We know where current track leads:

Let’s explore another track to optimize

Lessons Learning System

Let’s try to apply some of the ideas we just offered and see where they lead

Case Study: Fictitious NDF Accident Investigation

The case study is based on a model investigation report in Appendices F and G of the American Institute of Chemical Engineers' "Guidelines for Investigating Chemical Process Incidents" (1993 edition.)

Keep in mind your case study focus is on identifying, documenting, disseminating and assimilating lessons learned during investigations

Decided to offer case study exercise because that's where a lot of the Lessons Learning System difficulties arise

Hold up Book, discuss "Best Practices" idea, pros and cons

Case Study Plan

Overview Case Study incident setting, synopsis, investigation process and findings

Provide rules for Case Study

Develop investigation using alternative I/O framework and tools, with rules provided

Discuss experience and Lessons Learned from case study

Fictitious Incident investigation Overview

- Overview of facility
- Incident synopsis
- Description of investigation process
- Selected outputs

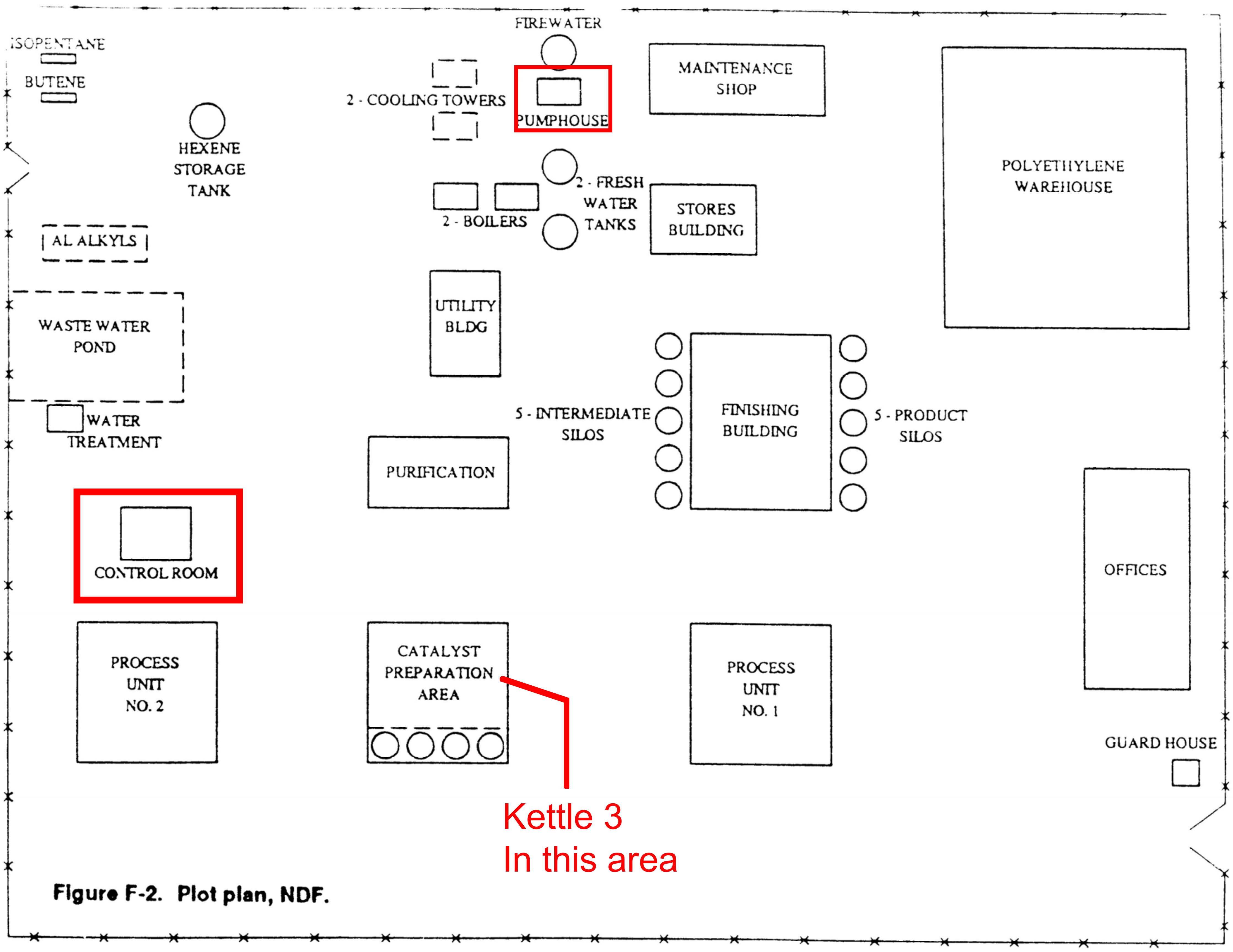


Figure F-2. Plot plan, NDF.

Fictitious Incident Synopsis

On August , 1991 a major fire occurred in a large polyethylene manufacturing plant. The fire originated at 11:10 A.M. in the Catalyst Preparation Area, in Kettle 3, and was discovered by an outside operator. The fire water supply system initially faltered, allowing the fire to intensify.

Several fireballs/explosions during the fire resulted in one fatality and fire personal injuries. One of the diesel fire fighting pumps was finally engaged, and supplied sufficient water for the firefighting efforts.

The fire was extinguished at 12:10 P.M.

Read quickly, mention 2001 revision use to supplement data

Case Study Procedure:

Using case study handout and asking instructor for data about case, class tries to get needed data and identify lesson(s) learned.

Rules:

- Group may organize selves into a teams if so desired, and assign investigation tasks to team members
- Ask questions in rotation (by team or individual).
- Each participant/team manage data the new way.
- Each participant/team find a lesson learned
- Instructor will answer all questions as faithfully as possible - no games

Start case study

Discussion:

What to do with lesson(s) learned?

At end of experiment, after one or two Lessons Learned have been defined,
Pose this question, and solicit answers.
Set up group for web dissemination, and for following slides

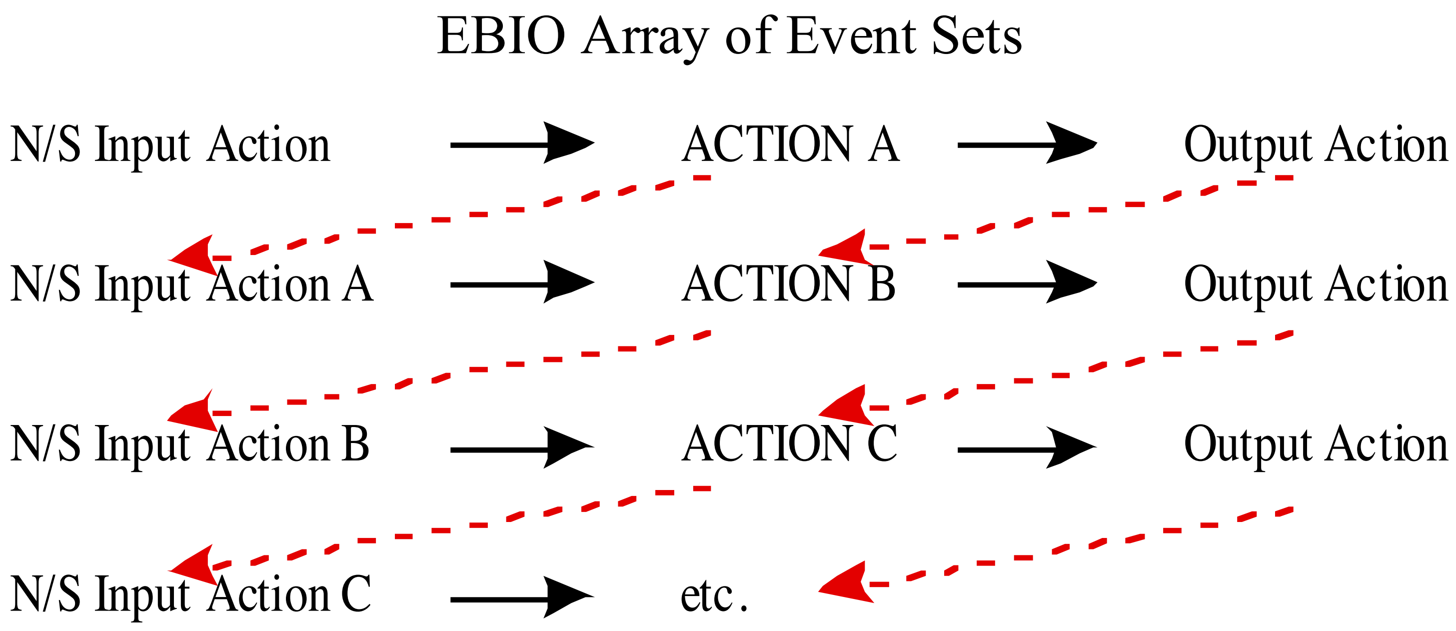
Document for Web dissemination

- Use extensible data structure and format
- Document data in that format
- Store in accessible locations
- Let machines search, filter, collate, retrieve and display

Highlight initial documentaiton of data in format for web to implement OODA strategy, where to store (in or outside organization)
Let machines search , notify via social networking,

Behavior sets recast into Input/Output arrays

Lessons learned display, showing behaviors *in context* (actions, with coupled inputs and outputs)



Patent Pending

Machine-generated from behavioral building blocks created on a personal PC or on the Web

Sample section from an active project:

Case1.ebio

Analyst: Analyst Project ID: Project Date:

Case ID	Set ID	N/S Input Actions	N/S St	Action	Conf/	Affected Actions	User Defined
		02:28:10:000 0d 0h 0m 0s 0ms			√	ash jammed extruder screws during startup attemp 2001-03-12 14:17:20:000 0d 0h 0m 0s 0ms	
Case1	52		X	? told lead operator extruder had been run with purge material 2001-03-12 06:45:00:000 0d 0h 0m 0s 0ms	√ √	Lead operator decided not to purge extruder 2001-03-12 06:48:00:000 0d 0h 0m 0s 0ms Supervisor agreed with lead operator to not repurge extruder 2001-03-12 06:50:00:000 0d 0h 0m 0s 0ms	Need to determine how wrong information was introduced into decision process. Was this data sole input to decisions by Supervisor and Lead operator? (n/s test)
Case1	53	? told lead operator extruder had been run with purge material 2001-03-12 06:45:00:000 0d 0h 0m 0s 0ms	X	Lead operator decided not to purge extruder 2001-03-12 06:48:00:000 0d 0h 0m 0s 0ms	√	Supervisor agreed with lead operator to not repurge extruder 2001-03-12 06:50:00:000 0d 0h 0m 0s 0ms	
Case1	54	? told lead operator extruder had been run with purge material 2001-03-12 06:45:00:000 0d 0h 0m 0s 0ms Lead operator decided not to purge extruder 2001-03-12 06:48:00:000 0d 0h 0m 0s 0ms ? observed small fire 2001-03-12 02:28:05:000 0d 0h 0m 0s 0ms	X	Supervisor agreed with lead operator to not repurge extruder 2001-03-12 06:50:00:000 0d 0h 0m 0s 0ms	√	Lead operator skipped extruder pre-run on startup check list 2001-03-12 10:00:00:000 0d 0h 0m 0s 0ms	Need more information about this decision – habituated action or adaptive action?

Flash NDF EBIO array on screen (file path here)
Filter for one actor (Lead Operator) to show accessibility potential

EBIO Array Attributes help optimize Lessons Learned:

Conducting Investigations.

- Provides continuous feedback about completeness
- Enables behavioral I/O coupling
- Produces assimilable lessons content
- Minimizes subjectivity, abstractions, ambiguity

Disseminating Lessons Learned

- Minimizes latency
- Enables interoperable machine processing
- Offers behavior context
- Enhances assimilation potential

Review slide elements quickly

Discuss experience

Free discussion for about 10 minutes, plus more after session if desired.

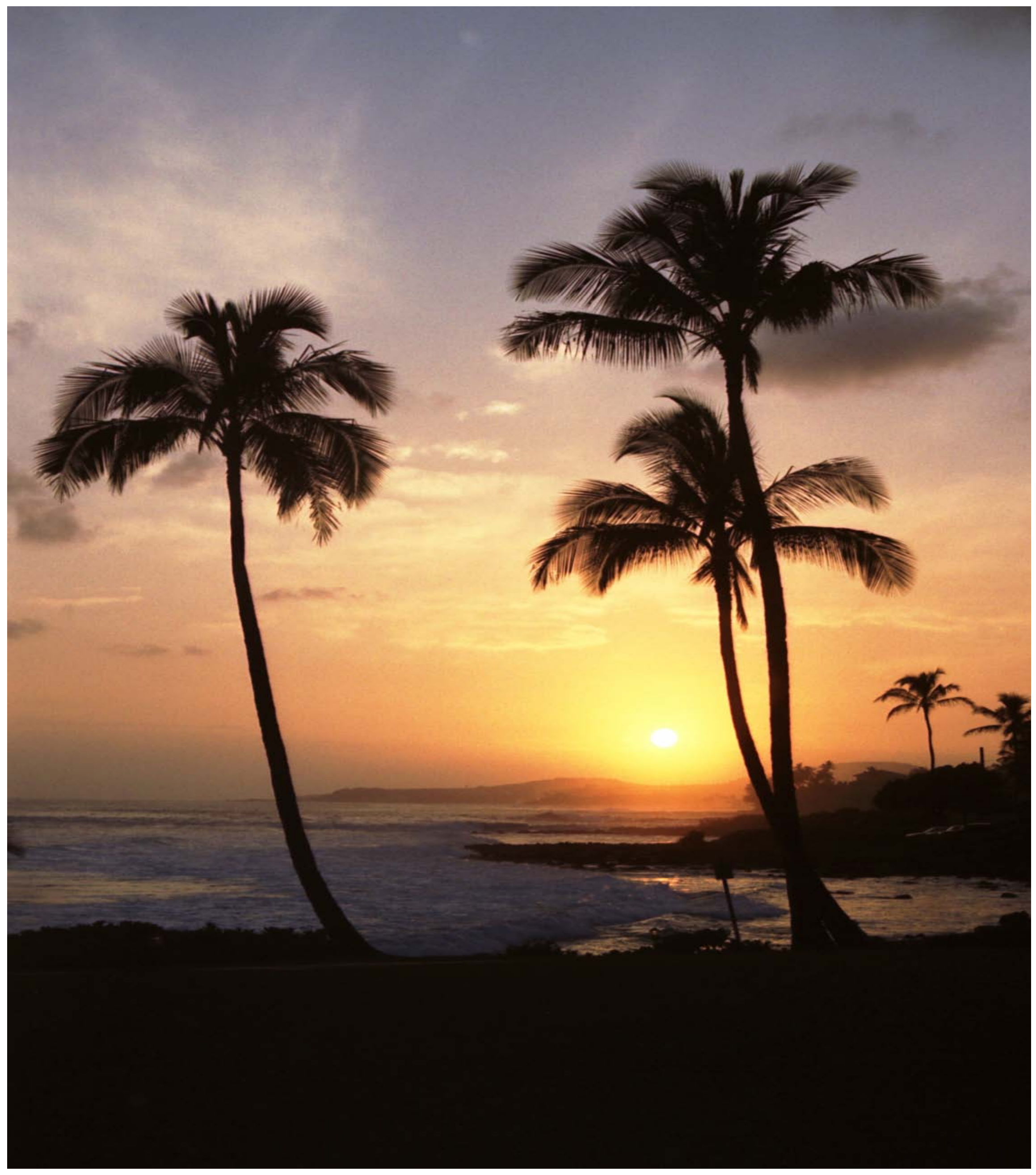
To Summarize

- We indicated Lessons Learning Systems merit examination
- We suggested some System evaluation attributes
- We explained why system attributes are difficult to optimize
- We offered some ideas to accomplish that

We’re done!

Thank you for participating.

Let’s adjourn.



luben@mac.com

billcarey@mac.com