cimetrix®

by PDF/SOLUTIONS™

The Role of Streaming Data in Smart Manufacturing: Methods, Applications, and Benefits

Alan Weber

SEMICON Europa

November 2022

cimetrix by PDF/SOLUTIONS

This presentation contains forward-looking statements regarding PDF Solutions' future products and business prospects that involve risk and uncertainty. Actual results could differ materially from those discussed. You should review PDF Solutions' SEC filings, including its annual report on Form 10-K and quarterly reports on Form 10-Q, for more information on these risks and uncertainties. PDF Solutions does not undertake an obligation to update any such statements.

Agenda

- Presentation objectives and context
- Manufacturing application examples
- Enabling industry standards for data collection and analysis
- Cloud-based implementation platform architecture

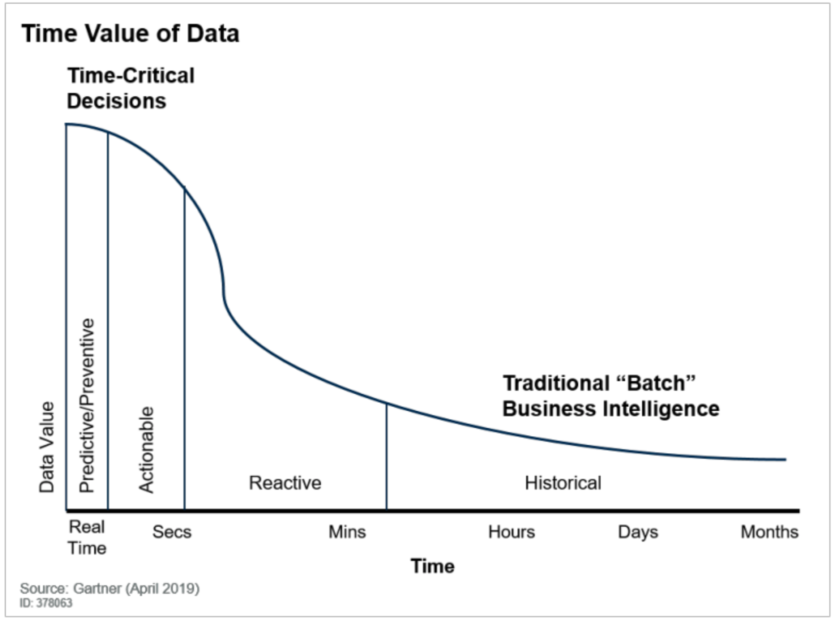
Objectives

Provide participants a basic understanding of

- Definition and distinctions of streaming data
- Manufacturing applications that leverage streaming equipment data
- Industry communications standards that support its access and analysis
- Performance expectations from the leading manufacturers
- Cloud-based platform architecture for reliable, scalable implementation

Context

- Streaming equipment data collection approaches complement the traditional message-/file-based methods in common use today
- Both approaches are well supported by the current generation of industry standards for connectivity and control
- Several manufacturing use cases can leverage streaming data collection to improve Key Performance Indices (KPIs)
- Key consideration: if the benefit of knowing the right answer decreases sharply over time, there is an opportunity for using streaming data



Definitions and terminology

- Streaming data
  - Data that flows continuously from its source(s) to its destination(s) to be consumed and analyzed in [near] real time
- Batch data (by contrast)
  - Data that is collected during a process and transferred to its consuming application(s) when that process is complete
- Manufacturing equipment [and process] data
  - Variables, settings, sensors, actuators, components, context, derivations, …
  - Events, alarms/exceptions, states, sequences, …
  - Analog, discrete, Boolean, text, timing, …
  - Fast/slow, simple/complex, sparse/dense, arrays, structures, …

For an excellent high-level discussion of the topic visit https://quix.ai/data-streaming-faq/

Familiar example

Roku vs. DVR

Manufacturing applications that leverage streaming data collection

Streaming data application examples

- Production throughput monitoring and excursion detection
- Fingerprinting / component performance monitoring
- Continuous calculation of productivity metrics (e.g., OEE)
- Real-time anomaly detection (LSTM-based learning)
- Substrate routing policy evaluation / improvement (Q-learning)

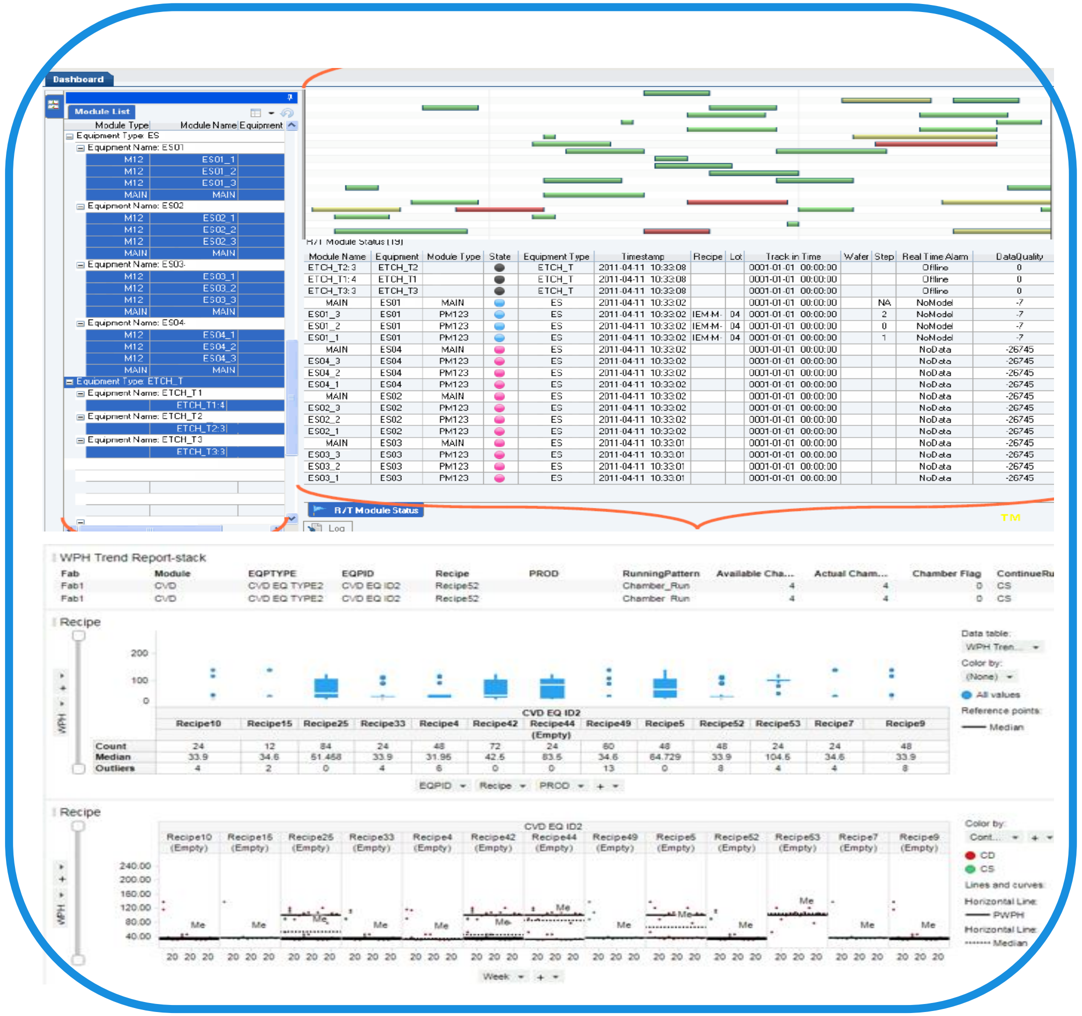
Note the wide range of stakeholder coverage

cimetrix by PDF/SOLUTIONS

Production throughput monitoring and excursion detection

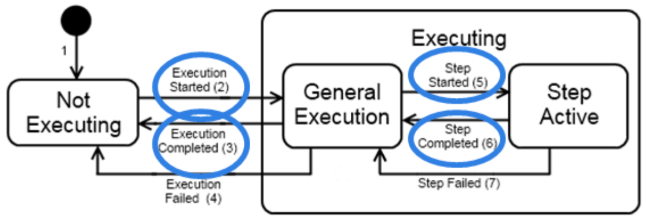
Application profile

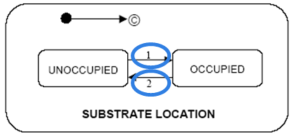
- Problem statement
  - Monitor bottleneck (e.g., litho) tool throughput performance to know when it drifts away from "normal" for whatever reason
  - This is important because any loss of throughput ripples throughout the line
- Solution components
  - Monitor events and calculate process time in real time
  - Evaluate context to compare "equivalent" runs; flag outliers
- Streaming data leverage
  - Standard material movement and recipe execution events
  - Context available at event occurrence
- Key ROI factors
  - Cycle time, productivity excursion MTTD (50% reduction), equipment throughput improvement (3-5%)

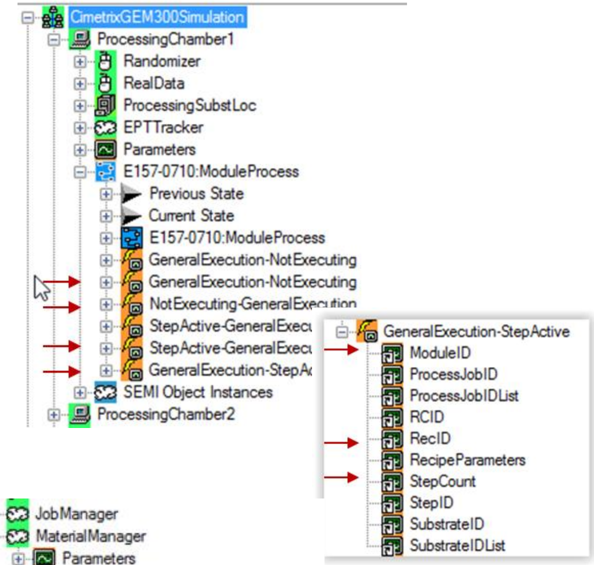


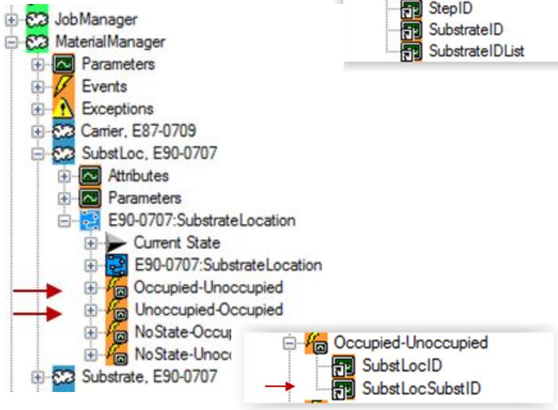
Production throughput monitoring and excursion detection

SEMI E157, E90 state machines, model content, and results

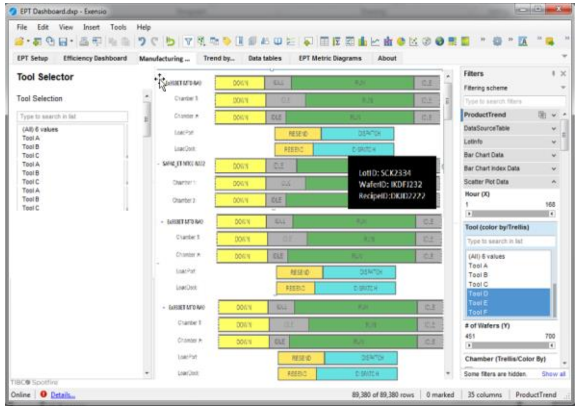




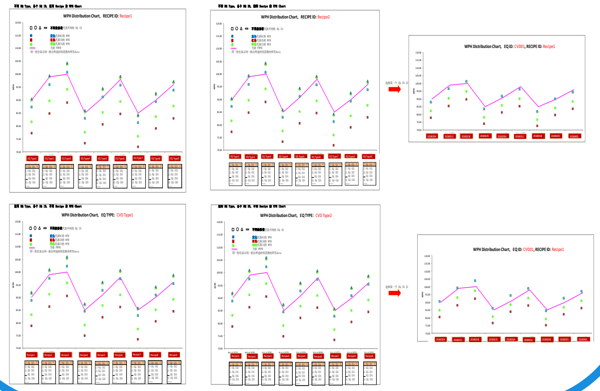




Equipment state report Gantt chart

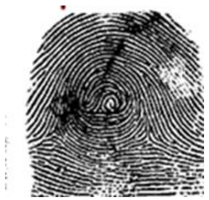


min/max recipe step times by equipment/product/recipe

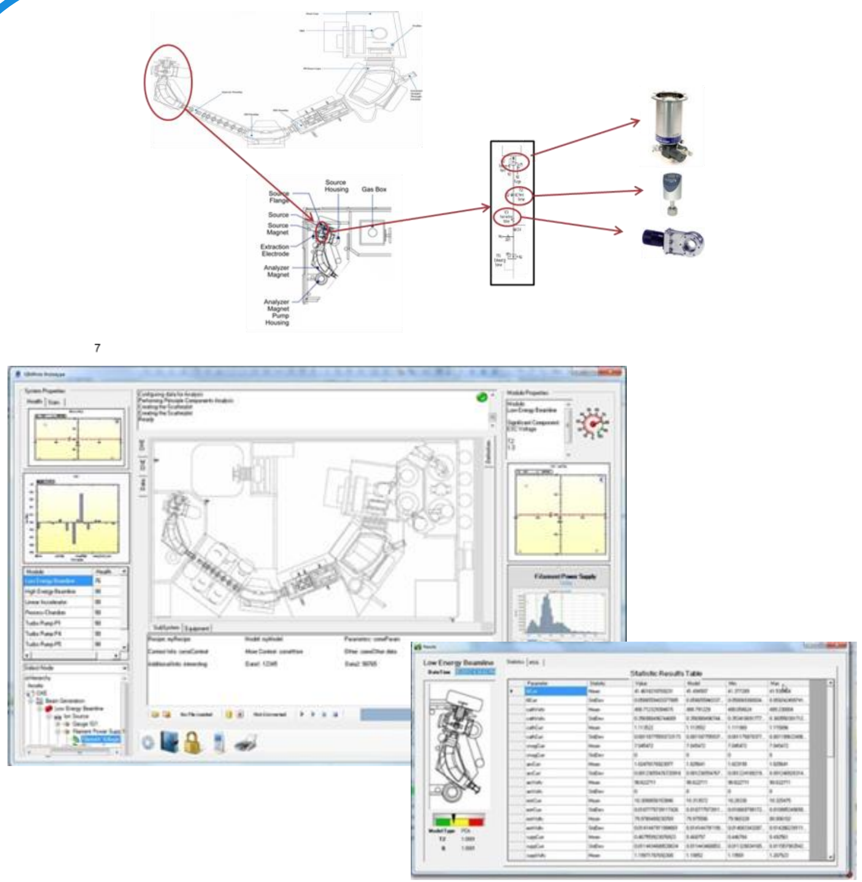


Fingerprinting / component performance monitoring

Application profile



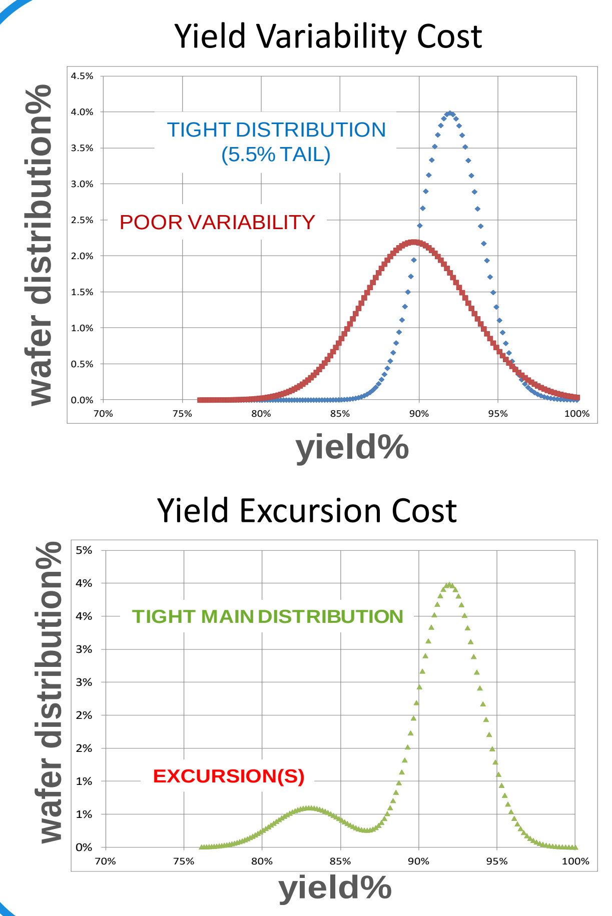
- Problem statement
  - Robust Fault Detection and Classification (FDC) systems may compensate for component drift and mask incipient equipment failures
- Solution components
  - Characterize behavior of important equipment components and subsystems and normalize to make recipe independent – this is a “fingerprint”
  - Monitor equipment performance parameters, calculate fingerprint in real time, compare with stored expected behavior and flag discrepancies
  - Think of this as “pre-fault detection”
- Streaming data leverage
  - Continuous and accurate status of equipment/component health
  - Identify and address equipment faults before they occur
- Key ROI factors
  - OEE improvement, especially in reduction of unscheduled downtime
  - Shorten MTTD (mean time to detect) potential equipment problems
  - Reduce MTTR (mean time to repair) because fingerprint points to root cause



ROI factors and process excursion costs

Hypothetical megafab

- Factor values
  - Wafer value - $10,000 (average cost of WIP)
  - Hour of engineering/tech time - $150
- Cost of process excursions
  - Wafers per excursion – 500
  - Delta yield per excursion – 3%
  - Engineering time required to resolve – 160 hours
  - Cost per excursion = 500*10,000*.03 + 160*150 = $174,000
  - Excursion rate – 24 per year
  - Total excursion cost = $174,000*24 = $4.12M
- Benefit of fingerprinting with streaming data
  - Reduction in # and severity (yield loss) of process excursions – 25%
  - Improvement in excursion MTTD (50% for remaining excursions)
  - Annual savings = $2.9M (1.7 + 2.2)

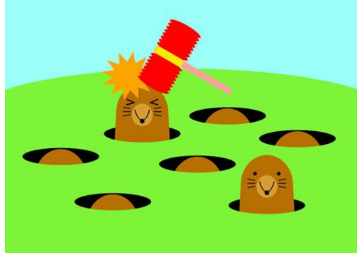


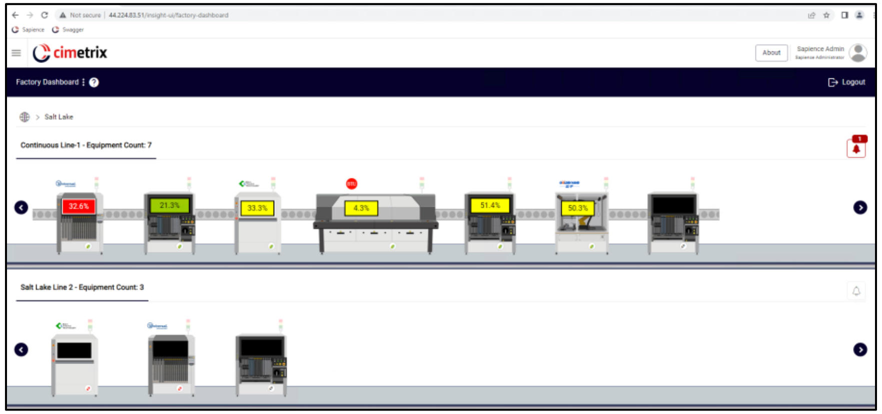
cimetrix by PDF/SOLUTIONS

Continuous calculation of productivity metrics

Application profile

- Problem statement
  - By the time you can analyze OEE and other productivity data the factory capacity has already degraded
  - Productivity metrics must be considered as a set because individual metrics can work against one another
- Solution components
  - Monitor events and associated parameters as they occur and calculate important KPIs and related metrics “on the fly”
  - SEMI Standards exist for wide range of productivity metrics
- Streaming data leverage
  - Use real-time GEM300 events to calculate instantaneous values
  - Start root cause investigation while data and pain is still fresh rather than after the fact
- Key ROI factors
  - OEE, cycle time, equipment throughput, factory capacity
  - MTTD improvement for all the above

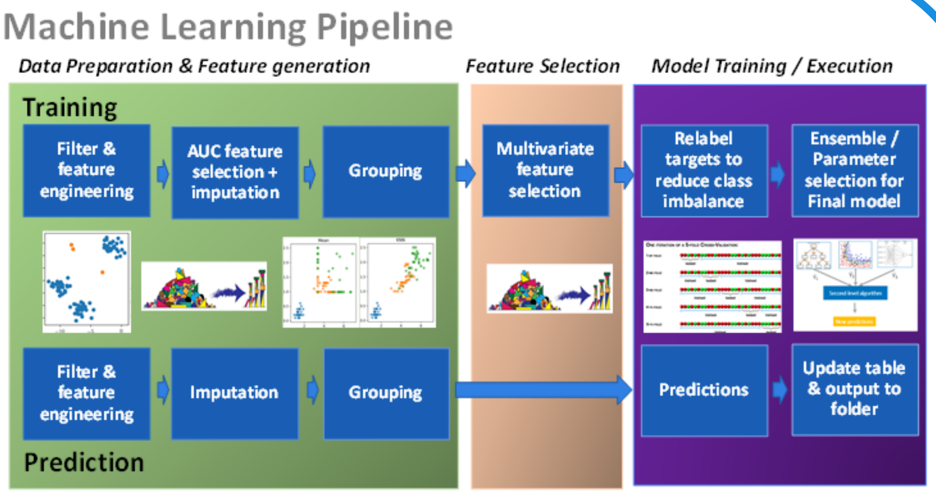


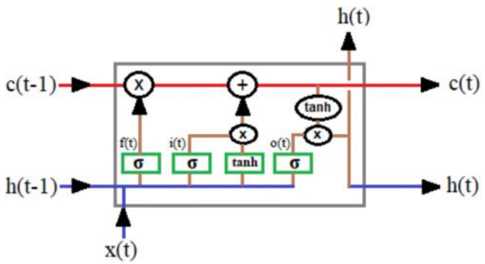


Real-time anomaly detection

Application profile – unsupervised deep learning from system logs

- Problem statement
  - It is hard to identify "abnormal" behavior in large, complex stochastic systems
  - Nevertheless, some sort of general-purpose early warning system could avoid costly downtime
  - Deep Learning technologies offer potential solutions, but they depend on massive amounts of good data… and much of it in near real time
- Solution components
  - Public machine/deep learning libraries, AI frameworks
  - Feature extraction/transformation modules
  - Experimentation frameworks to support algorithm selection, model training/cross validation, hyper-parameter comparison, etc.
- Streaming data leverage
  - Continuous access to system logs and other real-time indicators of equipment and manufacturing system performance
  - Takes another step on the path from reactive to predictive behavior
- Key ROI factors
  - MTTD for previously unseen system issues; overall system availability





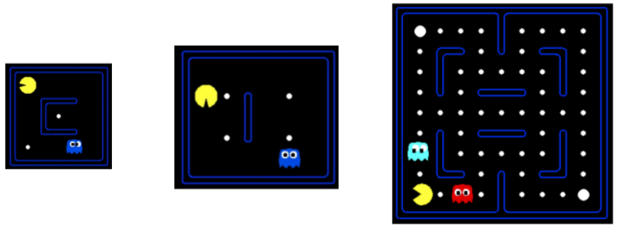
Long Short-Term Memory
Neural Net Architecture

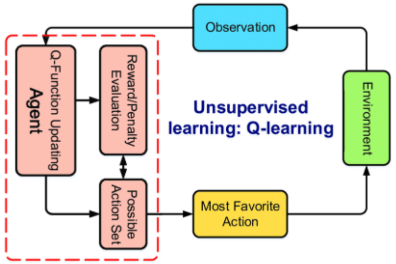
Widely available
Software Libraries

Substrate routing optimization

Application profile – [substrate] routing policy evaluation/improvement

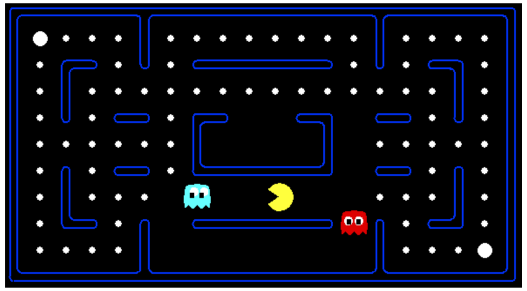
- Problem statement
  - Most manufacturing state spaces are far too large to represent completely
  - It is likewise impossible to “visit” (observe) enough of them during training to derive optimal “policies” (state/action pairs)
  - How can we bound/improve some aspect of system behavior as experience is accumulated?
- Solution components
  - Use feature vectors to describe system states (“Approximate Q-Learning”)
  - Learn about relatively small number of states through experience
  - Generalize (and continuously update) that experience for new states
- Streaming data leverage
  - Can update decision process continuously as additional experience (samples) is accumulated
  - Does not require explicit model of the system
- Key ROI factors
  - Depends on application focus… (e.g., equipment throughput optimization)





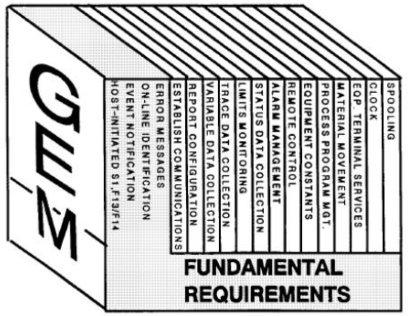
Reinforcement Learning

Policy Iteration

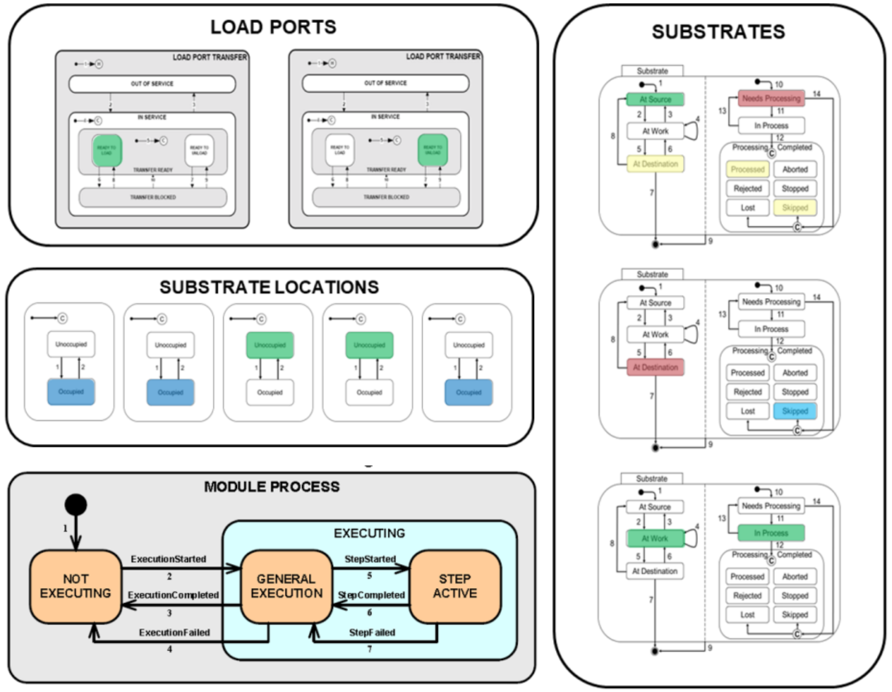


Enabling industry standards for streaming data collection and analysis

SEMI GEM/GEM300
Equipment automation

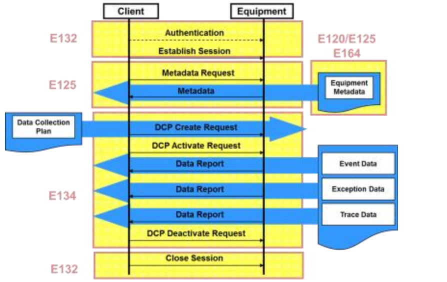


- Key concepts
  - Definition of unambiguous state machines (states, transition events, associated parameters, naming conventions) for all equipment mechanisms
  - Definition of additional mechanisms to support full 300mm automation (Process/Control Job, Carrier Management, Substrate Tracking, Recipe Execution)
  - Definition of interlocking operations sequences (“scenarios”) to indicate how these are standards are used together in automated 300mm manufacturing
- Streaming support
  - Trace (S6, F27) and event (S6, F11) reporting
  - Binary format can be very efficient

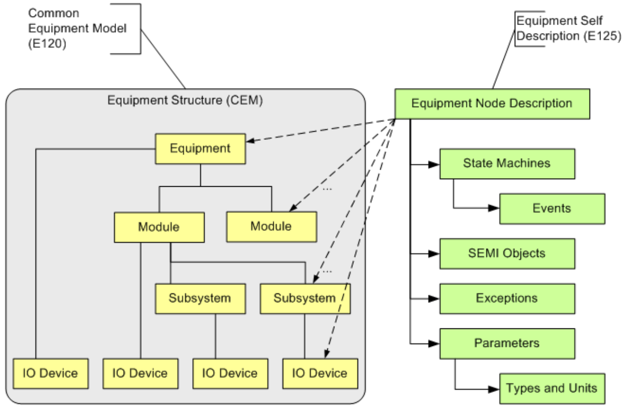


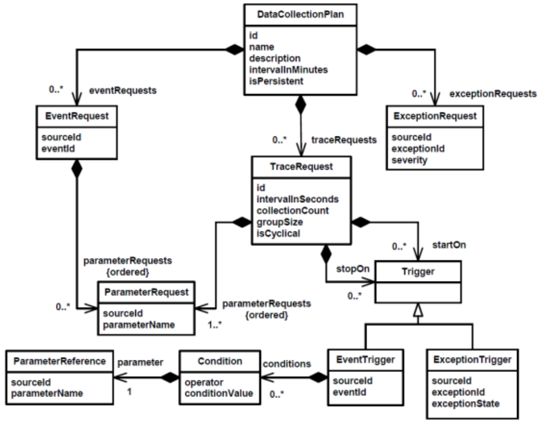
SEMI Equipment Data Acquisition (EDA) / Interface A

High-performance on-demand data collection



- Key concepts
  - Equipment-resident hierarchical metadata model represents complete interface capability
  - Powerful data collection plan (DCP) structure used to express consumers' data collection requirements to equipment; DCPs are defined and activated by user but validated by and stored on equipment
  - Multi-client architecture supports independent consumers (applications) of equipment data
  - Internet-based networking technologies (SOAP/XML (now), gRPC and Protocol Buffers (coming in Freeze 3))
- Streaming support
  - Trace request and event request elements of DCP design
  - Conditional triggers provide just-in-time data collection





Important SEMI Standards for Productivity Improvement
Provide evolutionary, complementary insights and capability

- Overall Equipment Effectiveness/Efficiency (OEE)
  - E10, E58 – Equipment Reliability, Availability, Maintainability (RAM, ARAMS)
  - E79, E116 – Measurement of Equipment Productivity, Equipment Performance Tracking (EPT)
- Product Time Measurement (PTM) (née Wait Time Waste, or WTW)
  - E168, .1, .2, .3 – PTM for 300mm production equipment, Material Control System (MCS), transport equipment (AMHS)

Metrics
(definitions)

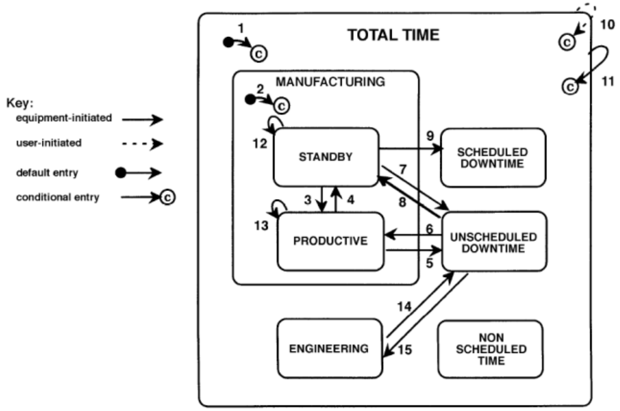
- GEM/GEM300
  - E30, E40, E94 – Machine States, Process Job Management, Control Job Management
  - E87, E90 – Carrier Management, Substrate Tracking
  - E157 – Module Process Tracking, Recipe Execution Tracking
- Equipment Data Acquisition (EDA / Interface A)
  - E120, E125, E164 – Equipment Metadata Model
  - E132 – Authentication and Authorization
  - E134 – Data Collection Management

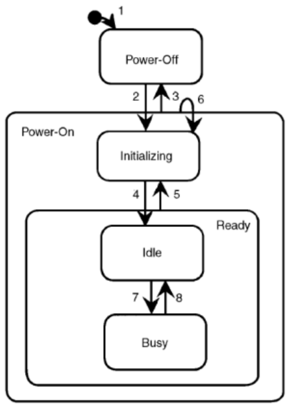
Mechanics
(data collection)

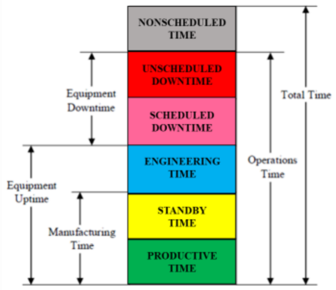
SEMI E58

Automated Reliability, Availability, and Maintainability Standard (ARAMS): Concepts, Behavior, and Services

- Key concepts
  - Conversion of E10 stack chart into state diagram
  - Definition of explicit equipment events that trigger transition between states
  - Definition of sub-states to provide more detail about equipment behavior
    - Standby for what?
    - Unscheduled downtime for what reason?
    - Scheduled downtime for what purpose?
    - Engineering time for what stakeholder group?
  - Object/attribute and table structure definitions to support services (messaging) interface to factory systems
  - Configuration options involving user-initiated transitions
  - Detailed compliance requirements (fundamental, additional)





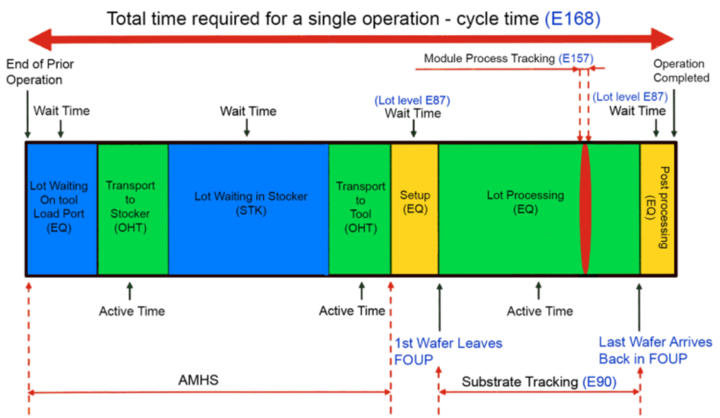


Legend	
Undefined	
Wait	
Exception	
Machine Failure	
Maintenance	
Engineering Time	
Waiting for Setup	
Change Lot / Setup	
Standby for Operator	
Standby for Material	
Engineering for Sale	
Productive Time	

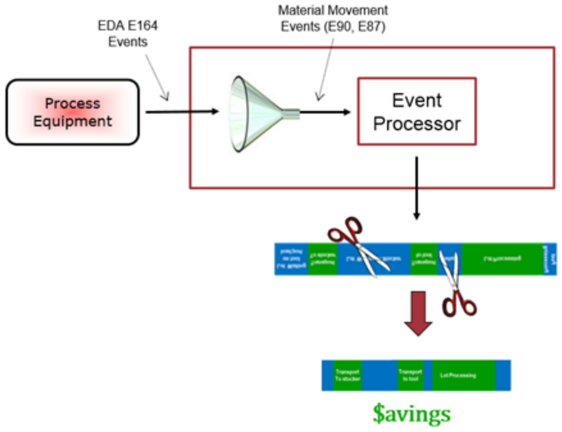
SEMI E168

Specification for Product Time Measurement (PTM)

- Key concepts
  - Definition of non-overlapping product material states (“time elements”) intended to cover the entire manufacturing calendar
  - Time elements fall into two categories (“wait” and “active”) at multiple levels; most detailed level (4) can be subdivided on lot or substrate basis
  - Definition of specific GEM300 events that bracket beginning and end of each time element (E30, E40, E87, E90, E94, and E157)
  - Parent standard (E168) defines overall concept and methodology; subordinate standards define specific time elements for
    - 300mm process equipment (E168.1)
    - Material Control System (MCS) interaction with factory system (E168.2)
    - Automated Material Handling System (AGV, OHT, …) hardware (E168.3)



SEMI E168 (and supporting GEM300/EDA standards)



The importance of architecture and standards

Implementation platform architecture and related cloud-based system technologies

Gigafab streaming data expectations
Aggregated across multiple companies and factories

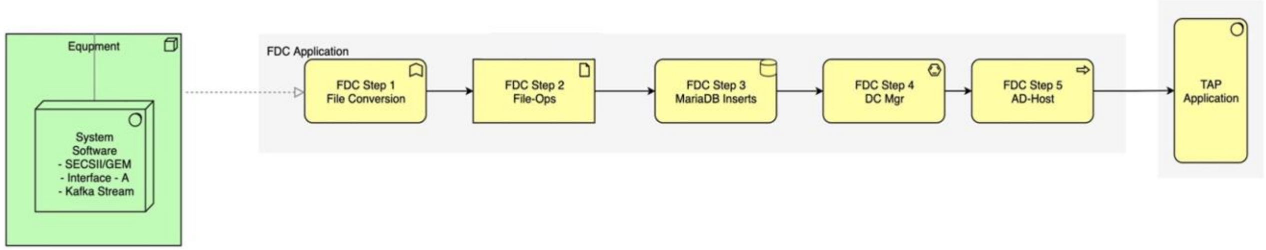
- Per process equipment (average)
  - 500-1000 parameters
  - 1-10 Hz sampling rate
  - 10-20 simultaneously active data collection plans (DCPs) – say, 1-2 per chamber
  - Worst case: 10K samples per second
- Per factory
  - 500-2000 units of equipment
  - Overall volume – no greater than 2X storage expected for binary values
  - Latency – less than 1 second for 90% of message traffic to support on-line apps
  - Storage in “data lake” for subsequent analysis
  - Worst case: 20M samples per second
- Scalability and availability
  - Seamless support for increased data collection demand and fault tolerance

cimetrix by PDF/SOLUTIONS

Streaming Value Delivery

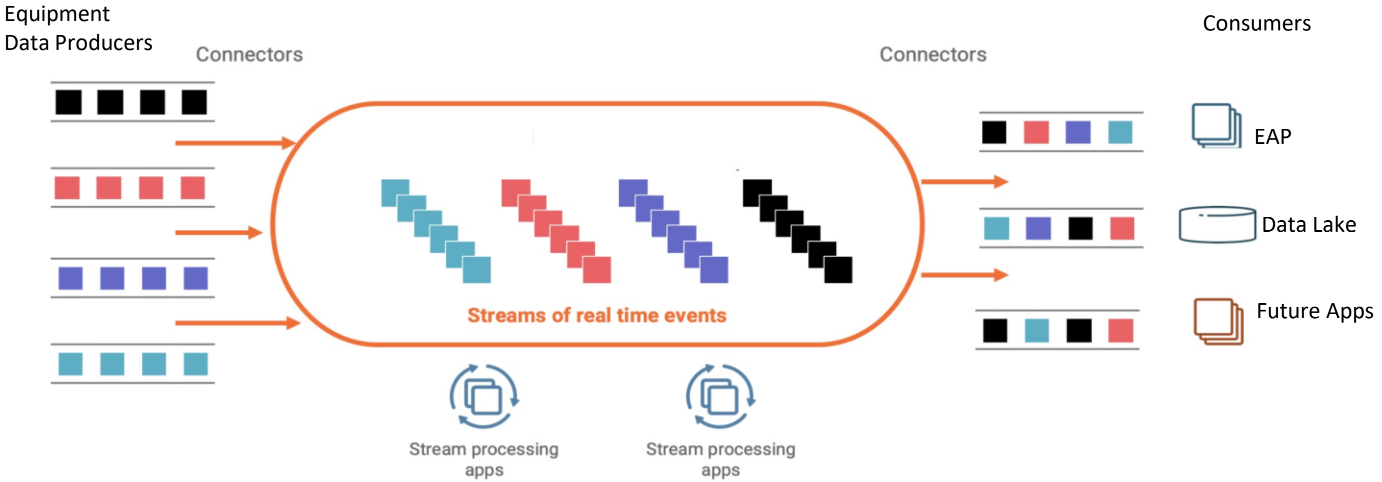
Status Quo

FDC App is limited by file operations, SQL loading, DB Inserts, old technology



Future Architecture

Enhanced FDC with streaming architecture is scalable, highly available, and provides competitive data bandwidth



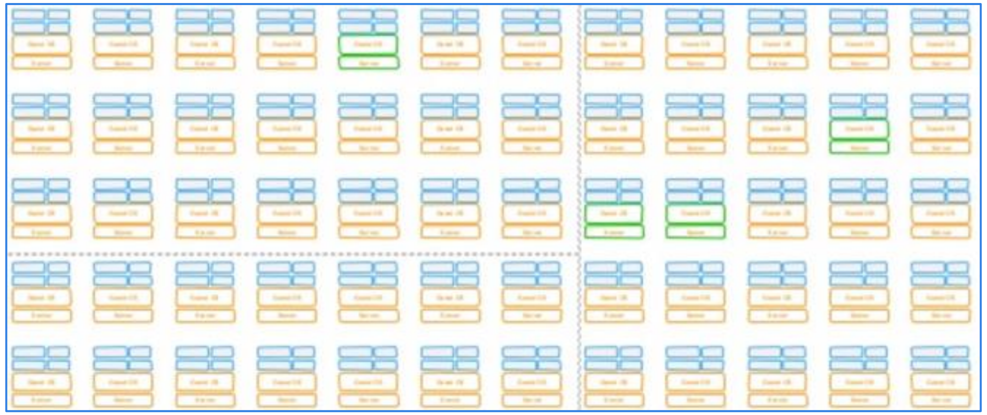
Streaming Philosophy

- Event Streaming with Kafka
  - Messaging
  - Persistence
  - Data integration
  - Fault Tolerance
  - ➢ **New Use Cases Are Inevitable**
- Kafka is designed to write data in a fast, scalable, and durable manner. It’s named after Franz Kafka

Cloud-Native Architecture

Manage 1 Host: Not Hard

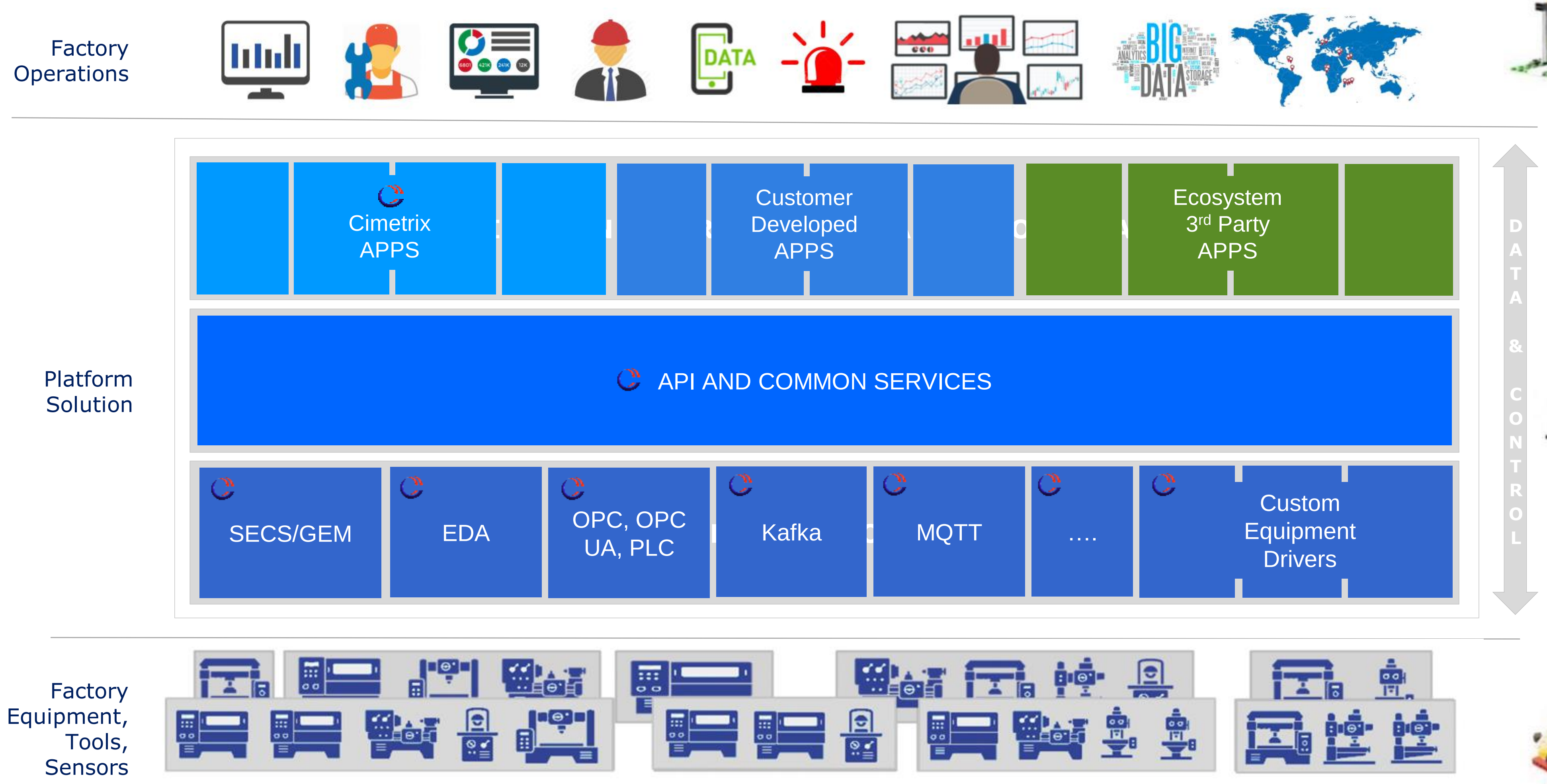
Manage a Fleet: Hard

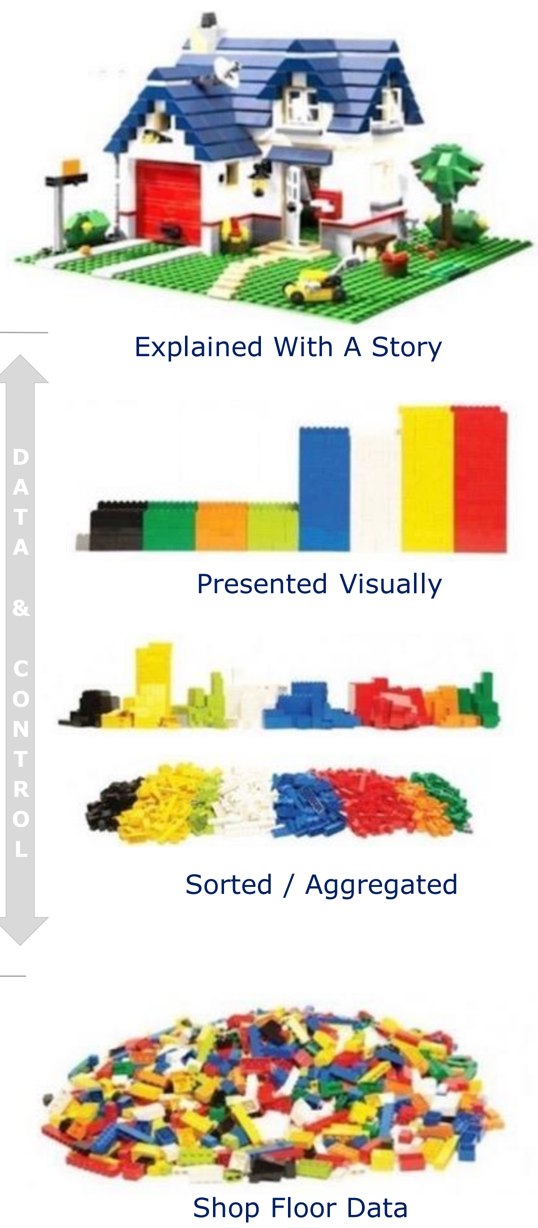


Designed for the Cloud

- Cluster Orchestration is Handled by Kubernetes
- Kubernetes Manages
  - Horizontal scalability
  - Load Balancing
  - Health Checks
  - Autoscaling
  - Container Lifecycles
  - Automate Deploys

Smart Factory Platform





Thank You