What do Virtual V&V and Digital Twins Have in Common?

Scott C. James[1] and Brian Hale[2]
Applied Dynamics International, Ann Arbor, MI, 48108, USA

Modern aerospace systems have continued to grow increasingly complex, demanding an advanced approach in test and evaluation. This paper explores broad industry trends toward growing capabilities in virtual Verification and Validation and details how investments in this area can be used to build and deploy full-fledged Digital Twin systems. These modern, automated methods are contrasted against traditional test capabilities, and the returns when applying a harmonized approach in PIL, SIL, and HIL testing. A framework-based cyberphysical development and test platform is offered as a solution to effectively develop an automated V&V capability while leveraging the latest Machine Learning and Artificial Intelligence technologies. The three emerging types of Digital Twins are identified, including Type 1 - Digital Twin for MBSE Systems Development; Type 2 - Digital Twin for What-If Analysis, and Type 3 - Digital Twin for Live Monitoring, and presented alongside a discussion of the commonalities between a Digital Twin and a virtual V&V system. This project began when MxD, the Nation's Digital Manufacturing Institute, worked with their key member manufacturers and identified a critical technology gap preventing the widespread adoption of Industrial IoT and Digital Twin capability. Industrial IoT and Digital Twin technology enable you to access your manufacturing data in real-time to reduce downtime and energy consumption, improve safety and efficiency, and more.

I. Nomenclature

ADEPT	=	ADI's industrial process automation software platform
AI	=	Artificial Intelligence
AIAA	=	American Institute of Aeronautics and Astronautics
API	=	Application Programming Interface
CPU	=	Central Processing Unit such as Intel x86 processors
DoD	=	US Department of Defense
DT	=	Digital Twin
GPU	=	Graphics Processing Unit such as NVIDIA processors
HIL	=	Hardware-in-the-Loop simulation-based test
HSI	=	Hardware-Software Integration
IoT	=	Internet of Things
IIoT	=	Industrial Internet of Things
ISA	=	International Society for Automation
IVV	=	Integration, Verification, and Validation
LRU	=	Line Replaceable Unit
MBD	=	Model-Based Development
MBSE	=	Model-Based Systems Engineering
ML	=	Machine Learning

OA	=	Open Automation
ONNX	=	Open Neural Network eXchange
O-PAS	=	Open Process Automation Standard
PIL	=	Processor-in-the-Loop simulation-based test
PNT	=	Position, Navigation, and Timing System
RDT&E	=	Research, Development, Test and Evaluation
ROI	=	Return-on-investment
RTOS	=	Real-Time Operating System
SIL	=	Software-in-the-Loop simulation-based test
T&E	=	Test and Evaluation
UUT	=	Unit-Under-Test
V&V	=	Verification and Validation
VIVV	=	Virtual Integration, Verification, and Validation or Virtual Test

II. Introduction

This article follows-up on papers published at the AIAA's 2014 Aviation and 2020 SciTech conferences, by the same author [1] and co-author [2], looking at the current state of cyber-physical and virtual systems integration, verification, and validation and its role in aircraft certification. In six years, the use of these methodologies has grown exponentially within the aerospace and defense industry and across many heavy industrial product markets. This article discusses best practices for cyber-physical and virtual product development seen across various industries using the case study example of civil aircraft development. However, these concepts and methods apply equally to a broad range of complex systems, including:

- Autonomous ground/air/sea transportation systems
- Cloud-based software systems
- Power generation and distribution system
- Space systems
- Weapon systems
- Naval systems
- Smart Manufacturing capability
- Smart Warehousing capability

This article also discusses how virtual and cyber-physical Verification and Validation (V&V) methodologies and Model-Based Systems Engineering (MBSE) engineering processes create a necessary foundation to realize Digital Twin (DT) capability for complex systems.

One rapidly emerging trend seen from many advanced heavy industrial firms is the adoption of the MBSE product realization capability based on a combination of commercial software solutions, and optionally, proprietary MBSE tools, and connected using an open architecture implementation philosophy [3]. This paper will discuss an example design framework leveraging powerful technologies including

SysML, Simulink, Python automation, analysis, and report generation, auto-coding, formal methods verification, and traceability.

A virtual or cyber-physical test framework is made up of one or more models, running as executable Linux/Windows applications with time/process synchronization and sharing data at a predefined rate. Cyber-physical test frameworks enable aircraft Line Replaceable Units (LRU) to be connected, in closed-loop, with simulation of dynamics like 6-degrees-of-freedom motion, aerodynamic forces, engine dynamics, and ground reactions.

A common example of an aircraft cyber-physical test facility used in commercial and military aircraft development programs is an "Iron Bird" test lab. The Iron Bird lab is used to test the aircraft's integrated fly-by-wire flight control system, by integrating the aircraft hydraulic system, flight control actuators and electronics, avionics networks, pilot controls, and the Flight Control Computer, and taking the integrated set of systems through highly-representative simulated flight through test in both normal and failure mode operations. A cyber-physical framework is used to interface the aircraft devices-under-test with real-time simulation of flight dynamics, engine dynamics, landing system dynamics, and potentially other systems of interest within the scope of test, and while enabling interactive and automated integration, Verification, and Validation. The testing performed on an Iron Bird lab generates test data and test evidence used to demonstrate that the flight control system's design performs as expected and to validate engineering assumptions made as part of the normal engineering process. While traditional Iron Bird test labs that interface directly with aircraft test hardware are extremely cost-effective in comparison to flight test, physical constraints and operational costs have pushed many to develop completely virtual V&V assets – a virtual Iron Bird. A virtual Iron Bird framework includes no real hardware being tested but only simulation models of all the pieces within the scope of the Iron Bird test approach, including software-in-the-loop models of each of the software-enabled aircraft systems are utilized. Because a virtual V&V test framework includes no real hardware, testing can be performed at an accelerated pace, with many test cases being executed in parallel, in the cloud, to accelerate the test campaign. The hardware-free nature of virtual V&V frameworks also makes it possible to greatly increase the scope of testing, achieving much higher test coverage and thorough evaluation of the design, running many more parameter sweeps, and analyzing the design in a more comprehensive manner.

The concept of a Digital Twin (DT) shares much in common with virtual V&V applications. Both virtual V&V and DT represent physical systems as realistically as appropriate, a true Digital Twin compares inputs from a real physical system being "twinned" and compares against the virtual twin to look for differences in the data and from those differences glean knowledge about the system. The Digital Twin will continuously or periodically update its knowledge base by feeding maintenance data, and other gained knowledge about the system, back into the Machine Learning (ML) algorithms parameters. With the explosive growth of Internet of Things (IoT) technologies, the Digital Twin can be interconnected to the physical system or process in ways not previously possible. This real-time connectivity is a significant differentiator between a Digital Twin and a virtual V&V application. However, to be highly effective, both a virtual V&V capability and a Digital Twin capability will leverage detailed functional, logical, and physics-based dynamics and software-in-the-loop dynamics to include some of the most complex and time consuming verification efforts. This article will discuss a best-in-class approach to Digital Twin capability that starts by building and leveraging a best-in-class virtual V&V platform.

This article discusses how advanced technologies, including Digital Twin (DT), Machine Learning (ML), and Artificial Intelligence (AI), are enabling more multi-physics dynamics to be incorporated in the design process, and delivering operational efficiency gains by implementing capability such as operational test process anomaly detection. In late 2019, the Open Neural Network eXchange (ONNX) ML model format became "the standard" when the ONNX project was adopted by the Linux Foundation. This article will expand on a framework approach that allows for drag & drop import and real-time deployment of ONNX models. Surrogate modeling methodologies, leveraging Machine Learning algorithms, allow multiphysics dynamics to be incorporated into virtual verification testing and design validation efforts and Digital Twin development. This article will explore how these maturing technologies can be integrated into a virtual test framework that can be utilized for virtual V&V testing as well as Digital Twin development.

III. Overview of Test and Evaluation for Military and Commercial Complex Systems

In 2020, the US DoD indicated strong demand for digital engineering through a range of innovation research solicitation requests raised throughout the year [4][5]. US Air Force Secretary Barbara Barrett stressed the Air Force commitment to innovate through expanded use of digital engineering, saying "Digital engineering isn't an option … it's essential. It's faster, it's cheaper, it's better," The Air Force plans to employ digital engineering to include the creation of virtual and digital models and environments, harnessing their full power to learn and experiment so the actual, physical systems are fully integrated and tested before production. The goal is to provide the Air Force with the ability to move much more rapidly from design to fielding of new weapon systems, on a near continuous basis as cutting-edge technology becomes available to integrate in follow-ons. [6]

The global industrial economy spends a tremendous amount every year on the testing of complex systems. Complex systems testing can be divided into two main categories:

- Development Testing – Systems V&V as part of a new system development program
- Operational Testing –Testing to support operation and sustainment of a system

These two areas of test tend to operate, both in commercial and military industry, as very siloed organizations with limited communication and low-likelihood of leveraging common methods, procedures, automation functions, design artifacts, etc. across the two testing organizations. Leading commercial industrial organizations are strategically removing these silos, as part of digital engineering and digital transformation initiatives, as will be discussed later in this paper, to realize significant ROI.

Within the DoD, both development testing and operational testing typically have a common characteristic of highly-manual, repetitive test activities with human error representing significant waste and cost within the test cycles. As a result, the digital transformation of T&E through automation and other modernizations represents significant opportunity for efficiency gains and cost reductions for the US military. Commercial industry, in general, has more actively incorporated modern test methods and automation technology into their operations, to date. Poor alignment between major prime military contractor goals and the goals of the US military commands is often cited as the cause for defense programs trailing behind commercial industry with testing efficiency.

A market estimate for military or commercial test automation, globally or within the US, is not readily available. However, the review of available market data is highly informative. Within the US DoD, spending on development test, known as T&E, is included within the Pentagon's RDT&E budgetary line item, requested to be $106.6 billion for FY 2021[7]. Through discussion with DoD program officers and other stakeholders, ADI estimates that T&E represents between 5% and 20% of the total RDT&E budget spend on a given new systems acquisition program, or between $5.3 billion and $21.32 billion spent annual on development testing alone.

T&E automation, or Open Test Process Automation, can be seen as a subset of a wider Open Process Automation technology and market space. Open Process Automation is discussed in greater detail later in this paper. The global industrial automation market is a very large technology solutions market with rise in recent years fueled by Industrial IoT growth and benefiting from IoT connectivity and edge computing technologies. In 2018, the global industrial automation software market was $35.4B, growing at 8.4% annually. Cloud-based software represents 38% of the spending whereas on-premise software holds 62% of the segment [8]. The IoT software platform market, a component of the broader industrial automation market, is forecast to be $1.64B in 2021, growing at an astounding 33%, and with the US as the largest geographic market at $469M in 2021 [9]. The Industrial IoT platform market (includes software, hardware and total solution services) has significant overlap with the global test automation market with a size of $10.6B in 2019. The largest segment within the test automation market is Functional Test automation, holding 49% of the overall test automation market or $5.15B in 2019 [10].

Traditional test assets can be divided into several distinct categories. While not necessarily focused on automation and often relying on labor intensive methods, these assets do provide value that a modern T&E approach can build upon.

- Material Testing – Material assets are involved in the evaluation and analysis of materials to be used in a system. These can include test setups for material degradation, corrosion, fatigue, and failure analysis.
- Software Testing – Assets to include system software, models, and internal developed or commercial testing applications.
- Environmental - Test assets used to replicate environmental conditions, which can often include extreme temperatures, pressures, and vibrations.
- HIL Testing – Traditional hardware-in-the-loop test methodologies evaluate controller design with real I/O interfaced with simulated sensors, actuators, and loads in real-time.

- Field Testing – Field test assets are used to fully test the system deliverable in real-world conditions. A field test asset will only be slightly modified from the final product to enable the collection of more data e.g., a test plane in a flight test program.

Consider the example of jet engine development. The table includes examples of test assets that can be found in many current jet engine development programs:

Jet Engine Test Assets	
Type	Purpose
Materials testing	Evaluate material performance of metals, alloys and composites to be used in construction of fans, turbines, combustion chambers, and nozzles.
Rotating test rigs	Test engine components under representative centrifugal forces. Rotating rigs allow evaluation at various operational speeds and corresponding temperature gradients.
Engine test beds	Ground-based and flying test beds are used for engine development and certification testing. Engines are heavily instrumented for data collection and monitoring.
Advanced engine system test rigs	Evaluate performance and assist in design of advanced control systems, to include the Full Authority Digital Engine Controller (FADEC), Electronic Engine Controller (EEC), engine health monitoring system, fuel metering unit, and others. Emulated sensors, loads, and communications are utilized.
High-altitude test beds	Evaluate engine performance across a wide range of operational atmospheric conditions in various system configurations. Large vacuum chambers enable testing to reproduce the low pressures found at the upper limits of the flight envelope.

Earlier in this paper we delineated the difference between development testing and operational testing. Development testing includes all the mechanical, electrical, software, integration testing, etc. associated with developing a new system or a new version of the system. Operational testing includes the test capabilities required to maintain and sustain a system. Leading complex systems companies and organizations have historically operated their development testing organizations very separately and decoupled from their operational testing organizations. Development testing is a function that must exist as part of the systems development process and exist amongst the new product development team. Operational testing is the capability delivered and used to maintain the system, identify failures, and recommend actions. Often operational testing capability must be delivered to the field. Development testing capability tends to have greater integration with agile development methods, whereas the larger focus for operational testing is on reliability and simplicity.

Historically, development testing capability was built up alongside the designs and implementations of the complex system itself, released, and then archived. Soon after the new design was finalized, the asset investments made for development test also came to a halt. Operational testing was different. Operational testing was a capability that must live on so long as the customer fleet of systems was operational. This difference in lifecycle for testing capability has likely contributed to the decoupling of development and operational testing capabilities in leading complex systems companies and government organizations. However, in today's modern complex systems, the development test assets will typically live on, throughout the life of the customer fleet. Business models for complex systems have dominantly shifted to include some portion of the business model that includes "revenue-generating upgrades" for the system. For example, commercial aircraft systems suppliers now commonly offer software and hardware/software upgrades delivering improved efficiency, range, health assessment, and other desired features, at a cost. This upgrade business creates a demand to operate the development test assets through the life of the aircraft platform. Similarly, the DoD intends to use modern systems platforms like the F-22 and F-35 fighter jets as capability intended to be upgraded and modernized in an intelligent fashion, for many decades, which will see development test capabilities operated more like operational testing capabilities.

Development and operational testing typically require vastly different types of test capability. In other words, the kinds of tools one would use for development test would not often be used on operational test or vice versa. Exceptions exist across most complex systems markets. A good case-in-point exception for our aircraft example is the jet engine test bed. The gas turbine engine testbed is a very expensive test asset used for both development and operational testing. In the later stages of a new engine development program, development engines are built and tested using test beds, as a critical component of the product development plan. Engine test beds are also used to support maintenance of the engine fleet.

IV. Adding Advanced Test Methods to the Mix

The adoption of advanced test methods has continued to increase the efficiency and capability of virtual V&V testing, making full Digital Twin development increasingly feasible. Standardized test methodologies and languages, such as SysML, open the door for high levels of automation in virtual test. Test methods are utilized at various stages of the design process, often in increasing complexity, to evaluate system design and performance:

- Virtual – Virtual test takes place completely in the cyber realm. The plant model and controller model communicate with simulated hardware systems. An effective virtual test is highly portable and lends itself well to collaborative efforts.
- SIL – Software-in-the-loop testing exercises production code, interfacing with a physics-based model of the plant. SIL testing can be completed entirely on low-cost client PCs, without any need to develop, maintain or occupy costly hardware assets.
- PIL – Processor-in-the-loop testing moves the test execution onto the targeted project architecture. Production code that has been compiled for the processor can reveal issues not found in SIL testing.
- HIL – Hardware in-the-loop testing brings real system hardware into the test setup. The controller and system are evaluated with interfaces to physical sensors and actuators. As the HIL system matures, components of subsystems can eventually be switched between real and simulated, depending on test objectives.
- Final – Final testing evaluates a system in a state nearly identical to the finished product, save for some modifications for data monitoring and collection. Operational costs of final testing are high, as are any changes to the design required when issues are found at this stage. Ideally, final testing is conducted largely for purposes of certification for safety critical systems, e.g. a test flight program

required for type certification. When the final test stage is reached, system dynamics and behavior are highly predictable and understood based on the previous levels of testing.

Figure 1 below shows a V-Model of an automated T&E systems development lifecycle. Programs often utilize both open and commercial tools to develop (and make use of existing) test assets. With model assets developed in standards-based environments, reusability and portability are inherent benefits. As physical mechanical and electrical design and development activities are taking place, so do their virtual and cyber-physical counterparts. Early in systems development, requirements are captured and may be modelled using standards-based modeling technology such as SysML. SysML allows for a system or subsystems to be characterized by the elements of which they are composed, the relationships and interactions between those elements, and the behaviors the system exhibits. This allows functional requirements to be captured, verified using automated simulation, and formally verified for consistency[11]. Another very powerful capability offered by formal modeling tools, including SysML, is the ability to automatically derive and execute functional test cases, i.e. test vectors, from the SysML model[12].

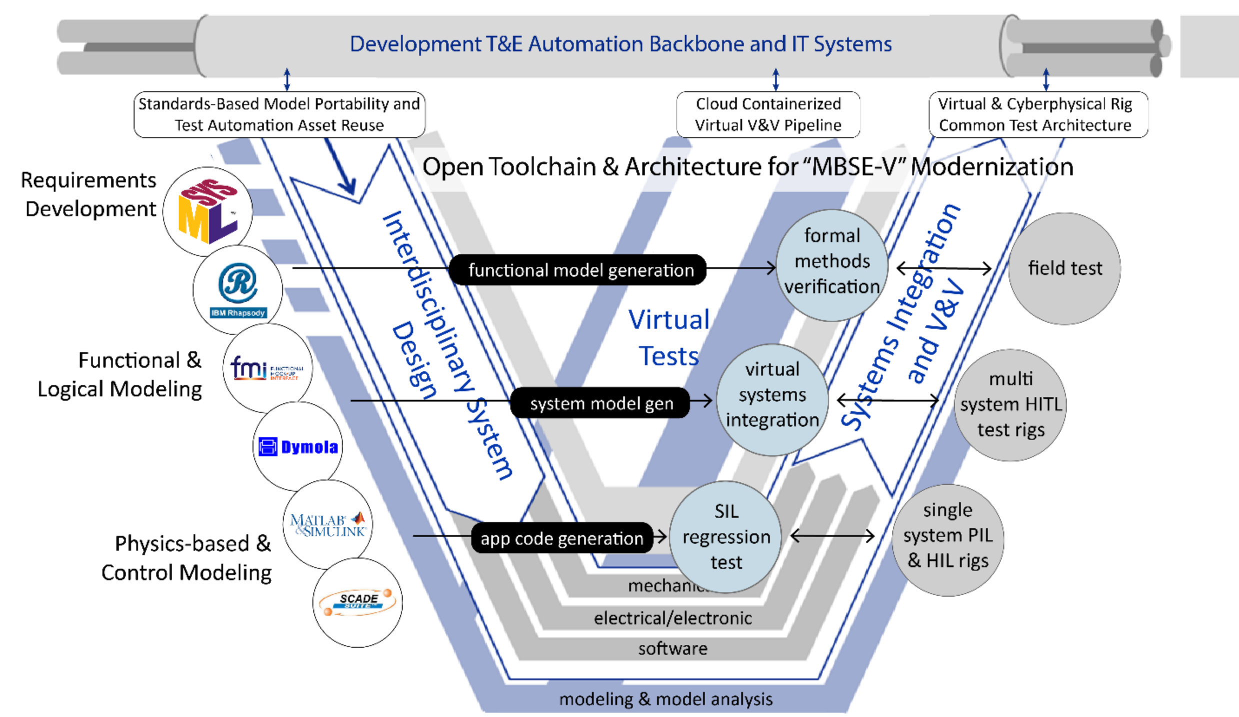

Figure 1 - Open Toolchain and Architecture for MBSE Design-V Modernization

In all test setups, test vector generation serves a crucial function in creating test data that is used to automate testing of virtual systems. Given the amount of test coverage required in safety critical applications, manually generating or recording test data is not feasible. Test vectors can be built up to create a library of scenarios in which the system will be validated.

Various commercial tools have been developed to address this need in MBSE testing.

T&E Capability Test	Test Rig Cost (US dollars)	UUT Cost (US dollars)	Total Test Equipment Cost
HIL Testing	$500,000	$80,000	$580,000
PIL Testing	$15,000	$1,000	$16,000

The table above presents a cost comparison between HIL testing and PIL testing capabilities for a widebody, Airbus 350-XWB type commercial aircraft engine for the development of the engine control systems for certification under DO-178C. As seen in the table, a HIL testing capability costs ~36x more than a PIL testing capability. For many complex systems development projects, if PIL testing is not employed then all test activities that would have been tested using the PIL rig must be performed using the HIL rig. While there is some non-recurring cost for each complex systems project associated with designing a PIL processor board that is used as the UUT in a PIL testing rig, especially for larger complex systems projects, the recurring savings from PIL testing provides fast payback on the upfront investment. Furthermore, many large complex systems organizations have standardized on a single processor architecture across multiple systems development programs. This significantly reduces the upfront design costs for PIL testing capability.

V. T&E as a Filter for Design & Implementation Issues

The combination of traditional and advanced T&E methodologies can be thought of as a set of filters rejecting different types of design and implementation issues and flaws. Virtual test is used to ensure fundamental design flaws are rejected early in the development project. SIL test is brought to bear once production code becomes available and provides a fine filter ensuring application system and operating system bugs are rejected ahead of integration with hardware. The PIL test capability provides a test platform for finding and rejecting Hardware-Software Integration (HSI) issues as early as possible. The HIL test provides a platform for exercising integrated subsystems as well as the complete integrated system in a “fail-safe testing” environment. Figure 2 illustrates the complete set of development T&E capability as a set of filters to catch and reject design and implementation flaws and issues.

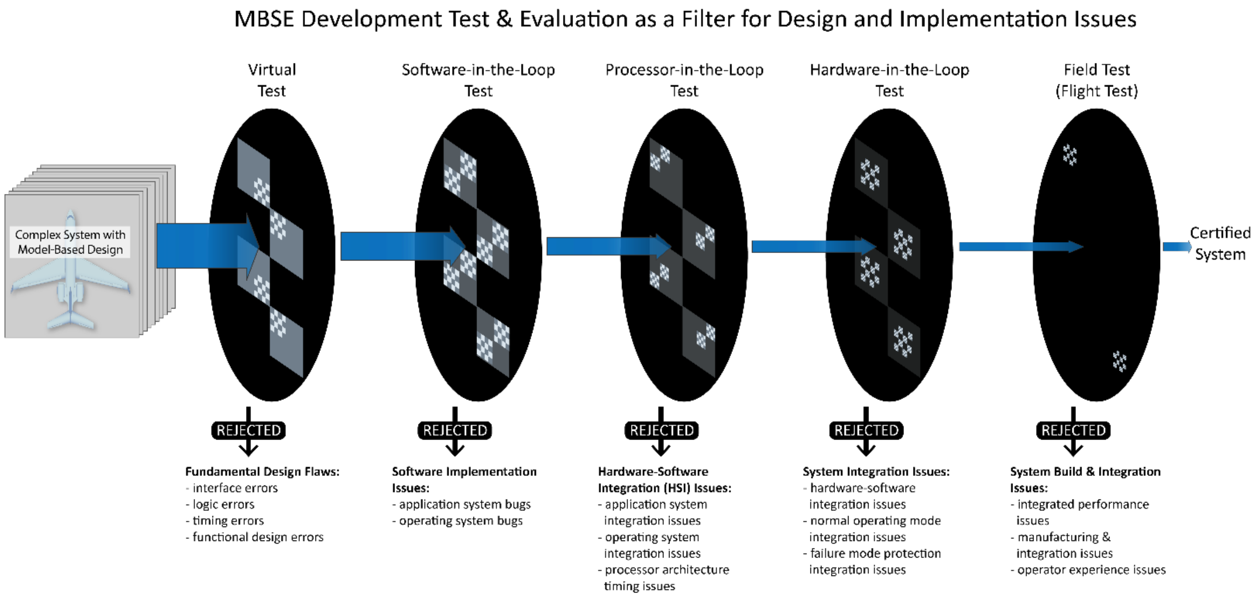

Figure 2 - MBSE Development Test & Evaluation as a Filter for Design and Implementation Issues

VI. High ROI on Test Automation Investments

T&E automation and modernization is receiving a lot of attention these days. This is partially due to the removal of some obstacles and some additional benefits and capabilities made available with emerging technologies in the areas of Industrial IoT and AI/ML. The main driving force for interest in T&E automation is the high level of ROI often achieved. Complex systems test campaigns vary widely in their shape and size, driven largely by the certification standards associated with a given system type. But the level of manual versus automated testing performed also varies widely across industries and organizations. In many areas, testing continues to be dominated by slow, manual testing with high risk of human error.

Aircraft cockpit development testing provides a good illustrative example for the range of testing approaches, extending from all-manual to highly-automated. In modern military and commercial aircraft programs, cockpit development testing is largely focused on verifying and validating new software releases as the development program progresses and then later as the aircraft platform is upgraded over its life. The table below compares all-manual versus highly-automated cockpit testing.

Testing Approach	Description
All-Manual	A ground-based cockpit test rig allows test pilot(s) and test crew to perform testing using HIL-simulated flight test methods ahead of flight test. The test pilot performs some number of simulated flight tests. Testing begins on the runway with takeoff, climb, cruise, landing, and specialized maneuvers performed as dictated by written test procedure. The test pilot "flies" the aircraft by operating flight controls and observing information on the flight deck indicators and glass cockpit. A test crew monitors the test progress, viewing live data plots, recording events of interest to be reviewed in detail post-run, and helps operate the cockpit test rig.
Highly-Automated	A ground-based cockpit test rig allows test pilot(s) and test crew to perform testing using HIL-simulated flight test and supports a high-level of automation. The test pilot performs only a portion of the total simulated flight tests with the majority of testing performed as a computer-driven automated activity. Testing is initialized to any stable flight condition, e.g. mid-air, without need to start from a stop on the virtual runway and maneuvers are performed through computer-execution of scripted test automation, analysis, and reporting. The test pilot "flies" the aircraft for a subset of total testing to be certain to incorporate some human-in-the-loop assessment as part of the test campaign. Test crew analysis and monitoring is replaced with live, automated test analysis with automatically generated test evidence reports.

It is somewhat counterintuitive to observe that commercial T&E automation tends of be further advanced than military T&E automation, given that such military asset investments do not face the same labor relations issues that commercial industry faces when rolling out automation investments. The reasoning behind the slow pace of US military T&E automation is generally cited to be caused by unfavorable supply chain power dynamics between the US military and the prime defense contractors. After decades of slow progress on US military T&E automation, new, agile contracting initiatives and disruptive technology investments performed by the DoD [13]-[15] appear to be creating an opportunity for the US DoD to achieve dramatic T&E costs savings through investments in open test process automation.

VII. Dramatic Cost-Reduction through Harmonized Test Automation

This paper has discussed a range of traditional and advanced T&E capabilities. The traditional T&E capability represents testing that has always been performed and needs to continue being performed. Advanced T&E capability is added to reduce the cost of and dependency on our traditional T&E capability. Each of the types of T&E capability, e.g. Virtual, SIL, PIL, HIL, are intended to reduce the duration and cost of their downstream capability. For example:

- HIL is used to reduce the cost of flight test
- PIL is used to reduce the cost of HIL
- SIL is used to reduce the cost of PIL

Virtual, SIL, and PIL capability also offer a cost-effective way to dramatically increase design and implementation scrutiny with far higher resolution and higher detail testing with much faster test cycle times. Leveraging these various, complex systems test capabilities can appear cost-prohibitive but are often viewed as the most cost-effective way to achieve certification for many categories of complex systems, particularly where safety-criticality, mission-criticality, and/or cyber-criticality are included within the requirements of the complex system.

Best-in-industry organizations have been able to manage the cost of their traditional and advanced T&E capabilities by designing and implementing a harmonized capability solution for test automation, test analysis, and test data & results reporting across the T&E capability types.

Designing a harmonized T&E automation capability involves a 360-degree assessment of the definition of the system and system testing approach. This includes a definition of the following T&E interfaces and functions:

- T&E Interfaces
    - System Interfaces
        - Operator interfaces to the system
        - Data interfaces from the system to other systems
    - Test Equipment Interfaces
        - Test instrument interfaces
        - Specialized test equipment interfaces
    - T&E Enterprise System Interfaces
        - Revision control systems
        - Data archive systems
        - Reporting & dashboarding database systems
- T&E Functions
    - Integrated system test procedures
    - Subsystem test procedures
    - Software test procedures
    - Hardware test procedures
    - Analysis methods
    - Test framework file management functions
    - Data archive functions
    - Test dashboard reporting functions
    - Test evidence report-generation functions

It is common to begin developing a harmonized T&E automation capability by adding open test automation frameworks, as described later in this paper, to overlay and digitally transform existing T&E rigs and facilities. As traditional T&E facilities go through a digital transformation, a harmonized T&E automation design may be built up, generalized, and extended. As automation is implemented, existing T&E facilities receive the automation benefits. As advanced capabilities are implemented for new systems designs, these capabilities can also leverage these test automation functions within their test operations.

Test Automation, Analysis and Reporting Language

Whereas open automation frameworks, discussed later in this paper, provide open connectivity across equipment from different suppliers and at different ISA-95 levels, interfacing with DUT, and providing COTS-based computing to execute simulation models and analysis, it is the Test Automation, Analysis and Reporting Language, or Test Automation Language, that embodies the bulk of system-specific test automation capability. We refer to this system-specific, or ideally system-family-specific, capability as a Test Automation Language. However, this software-based test automation capability may be implemented using a wide range of commercial and open source technologies. We refer to this capability as a language mainly because a significant majority of deployments we have observed have implemented most of this Test Automation Language capability using a scripting language. Increasingly the open source Python scripting language is used to implement the bulk of advanced test automation developments. Figure 3 illustrates the high-level batch testing structure included in a Test Automation Language.

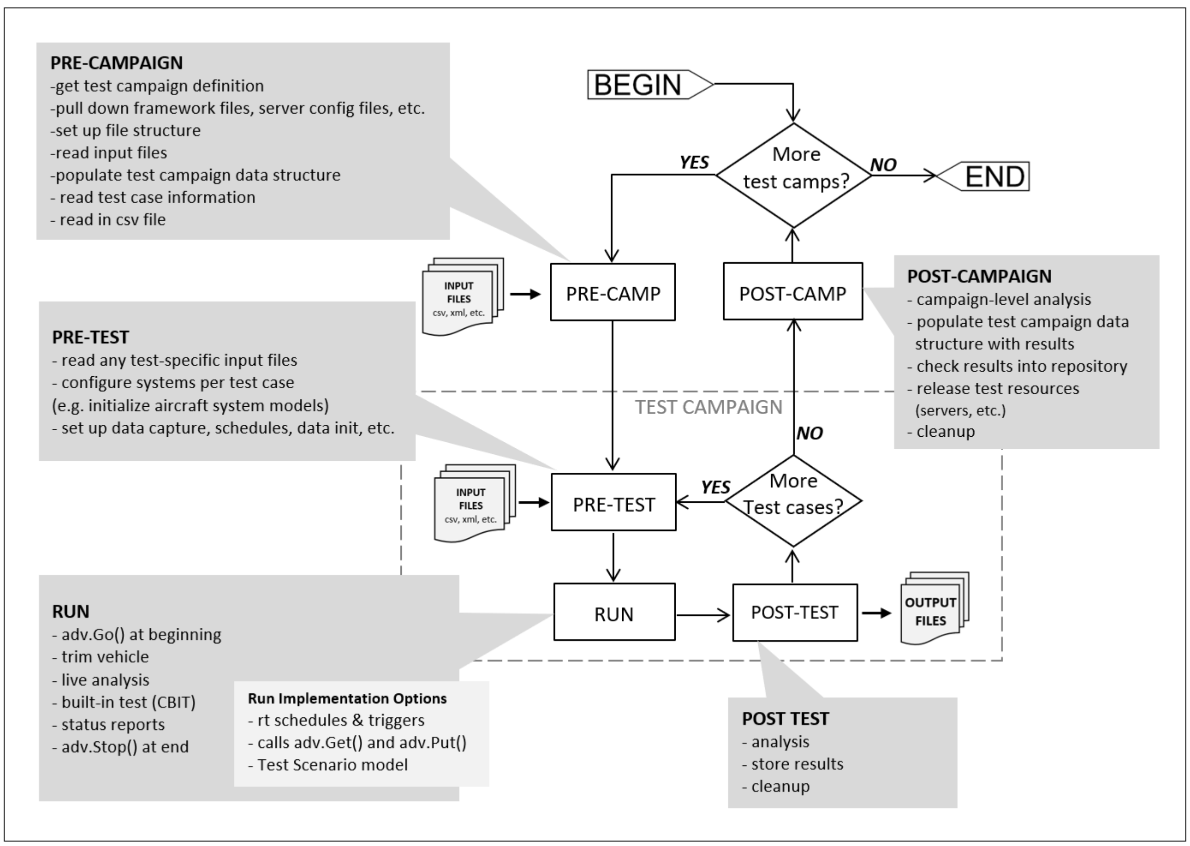

Figure 3 - Test Automation Language Batch Testing Structure

The batch testing structure, as part of the Test Automation Language, is responsible for automating the execution of a package of automated batch test cases, including test progression, test results analysis, and test report and other test evidence generation. A single instance of the batch testing structure is responsible for executing a single test campaign. A test campaign is made up of one or more test cases executed as a batch. The batch testing structure begins by performing pre-campaign activities including obtaining the test campaign definition, pulling down all files necessary to perform the test campaign, setting up the test campaign file structure and performing other preprocessing and readiness checks. After pre-campaign activities, the batch testing structure enters the batch cycle where multiple test cases are typically run in a sequence. Upon executing all test cases as an automated batch, the batch testing structure performs a set of post-campaign activities to perform post-run analysis, report-generation, and clean-up.

Multiple instances of the batch testing structure may be executed in parallel, if computing resources allow. Virtual testing and SIL testing may be operated with many instances of the batch testing structure executed with many open automation frameworks, all running in the cloud.

Beneath the batch testing structure layer of the Test Automation Language is where the bulk of the test automation, analysis, and reporting capability is implemented, ideally using a well-structured set of object-oriented design patterns.

Through harmonizing a complex systems T&E automation capability, maximizing the reuse of test automation assets across the various T&E capability assets, dramatic cost reduction is achieved. These T&E cost reductions come from the following:

- Reduced test automation asset development through reuse
- Reduced system T&E schedule and manual effort through finding problems earlier
- Reduced system T&E schedule and manual effort through faster cycle times with less waste (non-productive testing)
- Reduced system T&E schedule and manual effort by leveraging highly-parallel cloud-based virtual test pipelining

VIII. Computing Requirements for Automated T&E

To maximize efficiency of an automated T&E effort, a common platform for cloud-based virtual and real-time test is necessary. The MBSE "test framework" is a systems verification concept that helps describe virtual verification testing. A test framework is a flexible computational structure that allows simulation models, hardware, software, and/or a person to be interconnected to achieve a specific scope of testing capability, or cyberreality, as illustrated in Figure 4. Items inserted into a test framework, e.g. models, hardware, software, human, are referred to as "framework assemblies".

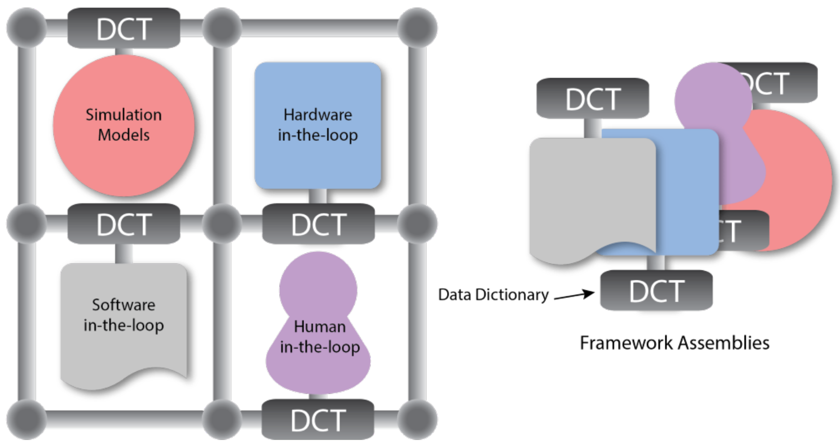

Figure 4 - Model Based Systems Engineering Test Framework

The test framework constructs a cyberreality in which tests may be performed. The test framework provides and progresses time and event interactions, progresses the computation of each model or software assembly, and handles communication between assemblies by moving data to and from memory locations called data dictionaries (DCT).

When the test framework includes hardware and/or human in-the-loop assemblies interfaced with simulation then the test framework will typically operate in real-time, i.e. cyberreality time progresses in synchronization with real time, and the framework becomes a cyber-physical system. When the test framework does not include a person or hardware then the clock may be progressed faster than real-time, if computational resources are sufficient, or slower than real-time, if large and complex models are executed. The test framework allows the framework clock to optionally be operated in real-time, to be operated at some multiple of real-time, or to run as fast as possible.

One such application that utilizes a test framework methodology is the ADEPT Framework - an industrial data and control software platform that links real-time industrial Linux servers as a distributed resource and provides desktop client control of the time-deterministic computing and data handling capability. The ADEPT Framework is used in the largest, most demanding industrial data and control applications across the global aerospace and defense industry, but also scales down to work with low-cost computing and open source real-time Linux. The open architecture framework allows users to leverage best-in-class COTS and open-source technologies in a common, project-based environment.

The ADEPT Framework allows for a heavily automated T&E workflow, from project development and test definition all the way through live and post-run results analysis. Figure 5 below shows a typical workflow when using the ADEPT Framework in a coordinated fashion for advanced computing and connectivity capabilities in T&E. A reusable project framework is developed in the ADEPT-DE development environment, based upon the system requirements, design, and test plans previously defined by the program. The project framework brings in model assets, in various popular formats, and assigns the framework to a real-time server for run-time. Project I/O is configured and connected to models using data dictionaries. Communications network definitions are created and maintained in ADEPT-DB, effectively managing thousands of signals required of complex systems, in a variety of protocols including ARINC-429, CAN bus, UDP, RS-422, RS-436, etc.

Test cases are executed, controlled, and automated using the developed frameworks in ADEPT-VI. While allowing for more traditional manual operation of test cases, advanced programs utilize the automation capabilities of this tool to operate at high efficiency at run-time while high value hardware assets are in use. Manual operation is replaced with scripted Python automation. Automation APIs allow for simple interfacing to external applications. Data monitoring and logging is configured for live or post-run analysis.

Live data visualization and analysis of test data is provided by ADEPT-GD. This graphical interface displays data collected from the real-time server on the client PC. Test results can be automatically collected using Python automation and deposited into test reports. Post-run, the high-level analysis can be performed utilizing the wide breadth of open Python libraries.

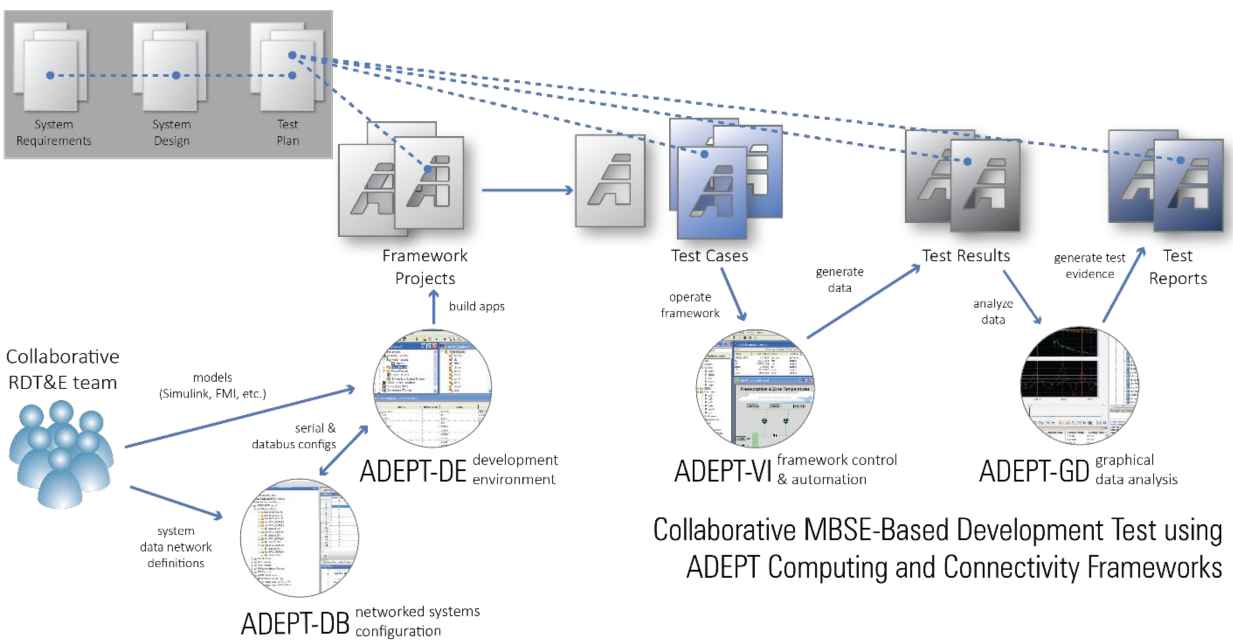

Figure 5 - Collaborative Development Test using ADEPT Computing and Connectivity Frameworks

Distributed multi-processor-core and multi-server computing frameworks are increasingly required to support the heavy real-time computational load of executing numerous simulation models, performing live analysis alongside the simulation, and supporting I/O handling computation, e.g. digital, analog, serial, network interfaces. These distributed test setups require the framework and multiple servers to operate as a single, time-synchronized industrial computing capability. Historically, implementation of distributed computing and connectivity frameworks put a heavy manual burden on the test engineering team to implement the distributed connectivity, synchronization and on control mechanisms. Newer industrial computing platforms, like ADEPT, are designed from the ground up with distributed capability built into the microservices platform to automate these manual activities.

Figure 6 illustrates an example distributed computing and connectivity framework. This example shows three real-time servers participating in a real-time HIL test framework. Whereas specialized real-time simulation appliances were once the only way to perform this type of high performance HIL testing, today COTS edge servers from well-known suppliers including Dell, HPE, and IBM can be employed with reduced cost and improved resiliency features. And where specialized commercial Real-Time Operating Systems (RTOS) were once the only way to perform this high-performance real-time testing, today open source real-time Linux can compete with commercial RTOS at a fraction of the cost. [16]. Executing on each server is one or more real-time Linux apps, or assemblies, controlled and synchronized by one real-time server daemon app. The figure shows aircraft subsystem models, e.g. left engine, right engine, flight control system, each executing within an encapsulated Linux binary executable. I/O apps are used to perform reads and writes to any I/O channels required to interface with the equipment under test, or DUT. Read and write operations on COTS analog and digital interface boards can require significant computational load and therefore the selection of real-time appropriate peripheral hardware is very important. Serial and network interfacing can consume a heavy computational load with message receiving, unpacking, optional error injection, packing, and sending. Standard network, databus, and serial communication such as ARINC-429, ARINC-664, MIL-STD-1553, CAN bus, and industrial Ethernet protocols such as Ethernet/IP, EtherCAT, Modbus-TCP may require a dedicated I/O app running on a dedicated processor core in order to handle the associated computational load. In addition to interfacing with the DUT equipment, other instruments and test equipment are also required to be included within the scope of testing. Specialized data acquisition and industrial control equipment, network and bus analyzers, and other equipment may be incorporated into the T&E automation framework to provide a single, unified test capability[17].

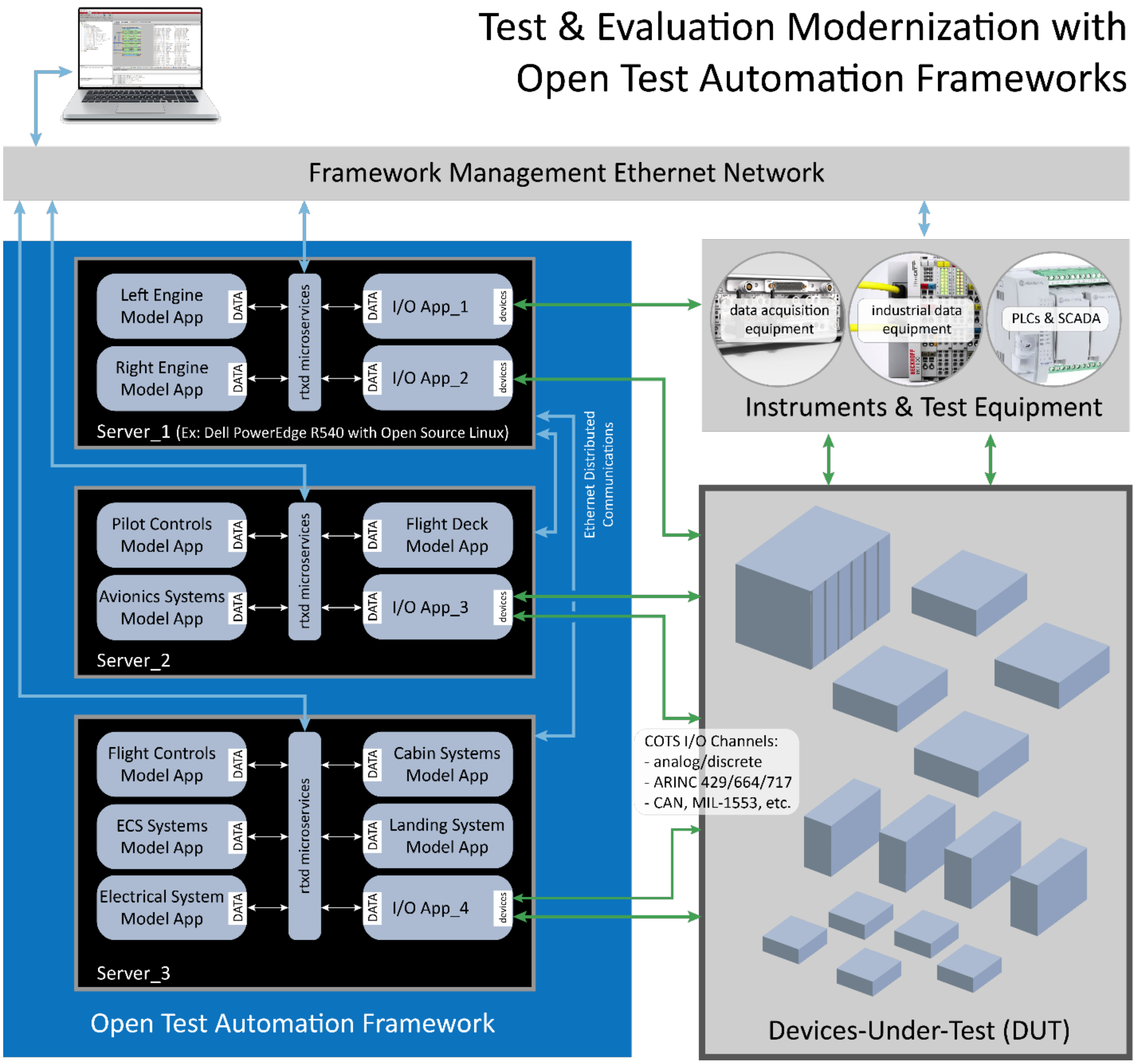

Figure 6 - T&E Modernization with Open Test Automation Frameworks

The ADEPT framework operates on a client-server architecture. Client tools are provided for project framework development, operation, and automation analysis. A real-time executive server daemon, the rtxd, provides a set of optimized real-time Linux services with a multi-processor, multi-server distributed real-time computing and data handling architecture. By keeping client and server functions separated, model test assets are easily reused throughout the program lifecycle, as they are not tied to complex hardware configurations. The computing demands on the servers are not burdened with functions that can be completed on the client, maximizing computational resources and enabling real-time operation. Figure 7 shows an Open

Test Process Automation Architecture, illustrating the separate client-server functions that utilize program test assets with traditional and advanced test capabilities.

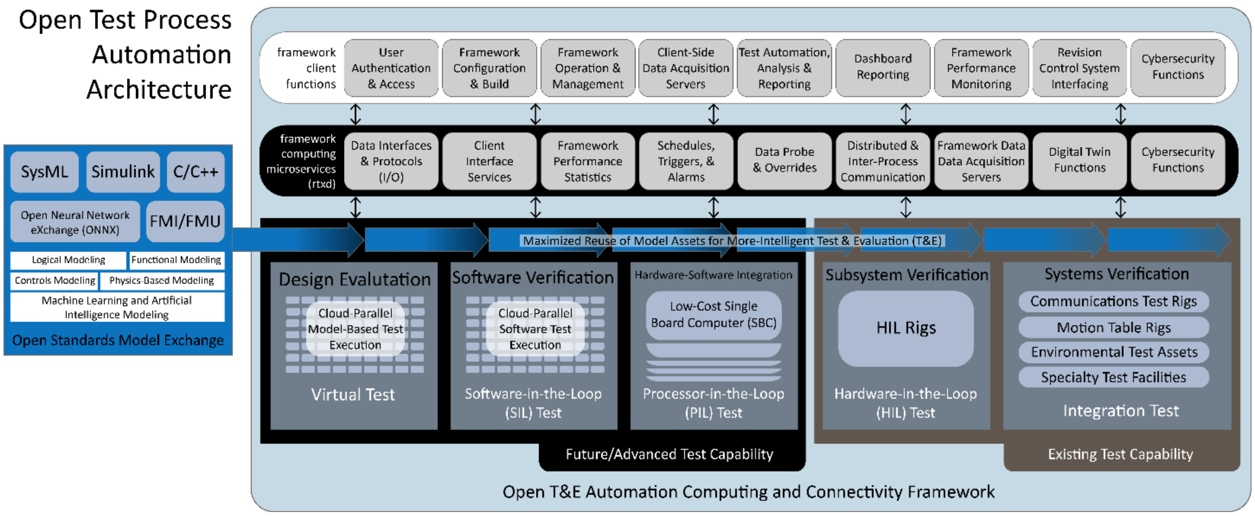

Figure 7 - Open Test Process Automation Architecture

A. Open Process Automation Standards

T&E automation and modernization share a great deal in common with broader industrial automation markets and technologies. Standards developed and maintained by organizations such as the International Society for Automation (ISA), and targeting large industrial markets including discrete and process manufacturing, can provide great benefit to T&E automation initiatives. Applicable standards include:

- ISA-106: Procedure Automation for Continuous Process Operations
- ISA-95: Enterprise-Control System Integration
- ISA-18.2: Management of Alarm Systems for Process Industries

Traditional industrial automation technologies have been developed and controlled by a single (or small group) of process control solution companies that require their version of hardware and software to be used. While this enables those companies to control quality, it creates single-source lock-in, which limits users' options for integrating new technologies, thus reducing competition and motivation for new entrants to address cutting edge challenges. Open Automation (OA) technologies allow products from multiple vendors to interoperate as a single, cohesive system, thus enabling increased options and competition for delivering future innovation.

In addition to the ISA standards listed above, in 2019 the Open Process Automation Forum, O-PAF, a standards organization managed by The Open Group, released version 1.0 of the Open Process Automation Standard (O-PAS). This standard incorporates the real-time and virtual computing capabilities required for process automation and discuss in detail in this paper.

IX. Digital Twins in T&E Automation and Modernization

DT capability is called out as one of the 5 biggest technology trends disrupting engineering and design in 2020 and listed on Gartner's Top 10 Strategic Technology Trends for 2017 and 2018[18]. The definition of a DT varies widely. This paper attempts to present a generalized and harmonized view of what is a DT. A DT is a purpose-driven dynamic digital replica of a physical system [19]. There are three distinct types of DT found in the literature:

Type 1: Digital Twin for MBSE Systems Development[20]-[22]
Type 2: Digital Twin for What-If Analysis[23][24]
Type 3: Digital Twin for Live Monitoring [25]-[29]

A DT is a virtual representation of a physical system that can be monitored and compared against the real thing. A DT can be used in real time to detect anomalies and predict failures before they occur, and it can also be used at an accelerated pace (and with many copies in parallel) in a completely virtual environment. When a DT is used in a virtual environment, it enables testing "what if" scenarios much more efficiently and safely, without the risk of harm to people or damage to the facility, and can provide insight that greatly improves quality, efficiency and safety. And since a DT is virtual, it can be tuned and improved over time to provide better feedback and more effective predictions.

The first type of DT listed above, used within MBSE product development, is the same T&E automation technology discussed throughout this paper. The second and third types of DT listed above are extensively interrelated to the first and offer potential benefit to T&E automation and modernization.

Product Digital Twins (DT)		
DT Type	Description	How Is It Used?
Type 1	DT for MBSE Aircraft Development	Used to perform automated T&E of design artifacts through an agile development process
Type 3	DT for Live Aircraft Monitoring	A live monitoring DT would exist alongside each aircraft sold. During the life of the aircraft, its DT would be executed in the cloud and compared against the real aircraft flying through the sky, taxiing on the runway, and stationary. The results of this comparison between the real aircraft and its digital replica provide information that may be used to reduce operating costs, increase safety, and improve the product experience.

Process Digital Twins (DT)		
DT Type	Description	How Is It Used?
Type 1	DT for MBSE Development of the Aircraft Manufacturing Line	Used to reduce the cost and schedule of developing a smart manufacturing line for the system.
Type 2	DT for What-If Analysis of the Aircraft Manufacturing Line	Used to help dynamically respond to unexpected system events using an assessment of possible actions to take
Type 3	DT for Live Monitoring of the Aircraft Manufacturing Line	Used for manufacturing line: predictive maintenance, anomaly detection, predictive quality, and operational optimization

Boeing is an example of a complex systems company having declared a plan to develop a range of DT capabilities for both product and process [3][30]. Significant economies-of-scale cost and schedule benefits can be achieved by developing a comprehensive strategy to make use of multiple types of DT capability, potentially across both product and process, that leverages and shares modeling and data assets across each capability and development initiative.

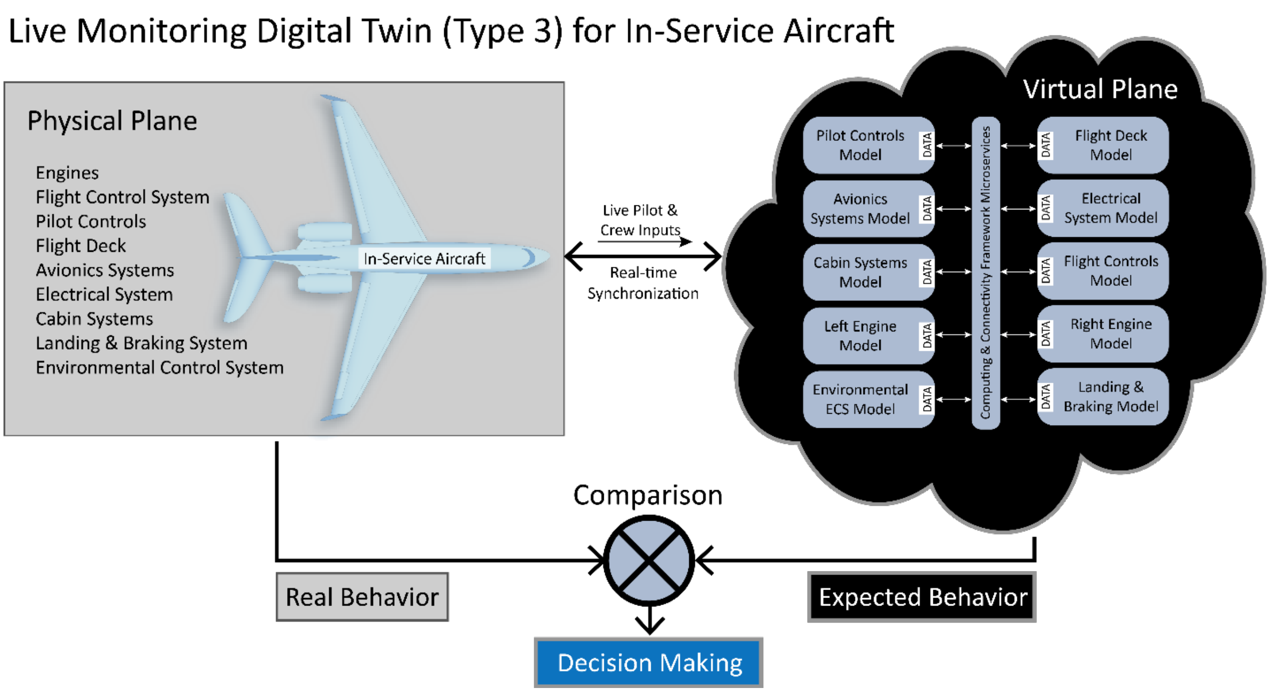

Figure 8 - Live Monitoring DT for In-Service Aircraft

Figure 8 illustrates a Type 3 live monitoring DT. Data is taken from the physical plane, an in-service aircraft, and from the virtual plane, a set of aircraft subsystem and environment models. As the pilot flies the aircraft, pilot control inputs are fed in real-time to the virtual plane models. The physical plane is compared to the virtual plane in real-time and the differences are used to better understand the system, e.g. predictive maintenance, predictive performance, and anomaly detection.

X. Machine Learning in T&E Automation

Machine Learning has been one of the leading emerging technologies utilized in the effort to increase the power and capabilities of automated testing and evaluation. Over the past several years, a tremendous amount of investment has been made in Machine Learning across industries, and the aerospace industry is no exception. Machine Learning offers the chance for not only T&E processes to be automated, but for the rigor and quality of the processes to be improved, as algorithms improve automatically over time. Interactions that previously required intensive and costly human labor continue to be replaced by Machine Learning enabled processes. Test tasks that were simple or trivial for humans, such as extracting information from an image, or physically pressing a button on a control system, were often costly and difficult to automate, prior to the growth of machine learning and artificial intelligence.

Computer vision, a subset of machine learning, aims to provide the capability to interpret digital images and video, often in real-time. Military applications of computer vision have long been applied for purposes of combatant detection and missile guidance. In a T&E setting, computer vision allows for increased automation of test functions, anomaly detection, and results processing.

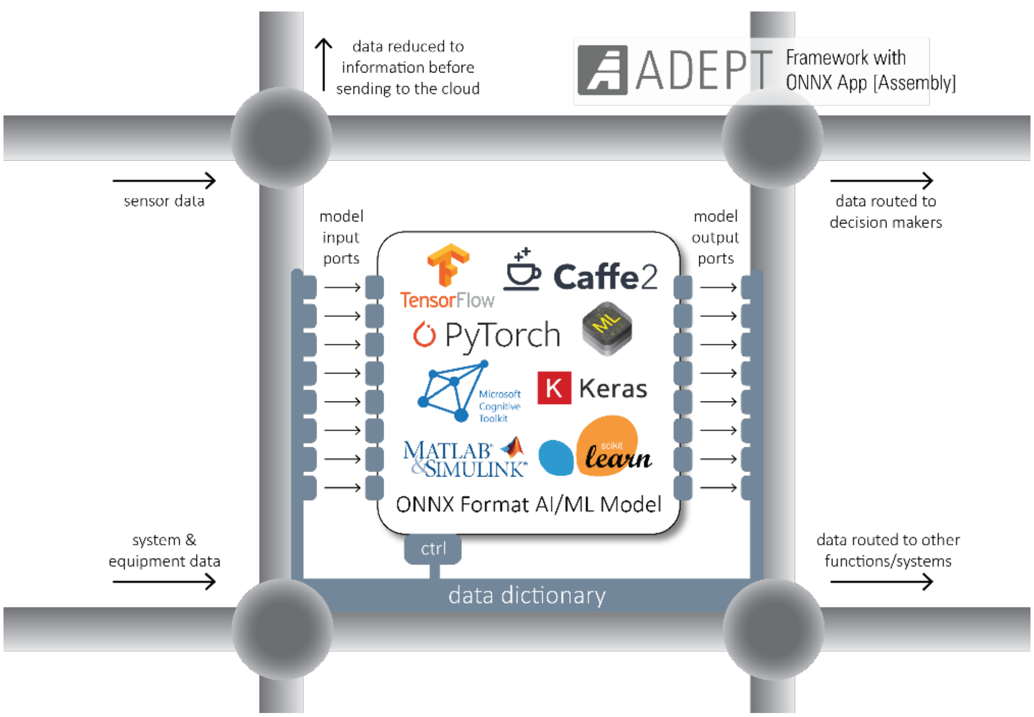

Figure 9 - ONNX Machine Learning Model Executing Live within an Open Automation Framework

Collaborative robots, known as cobots offer adaptive automation options in test and automation. Test operators can guide cobots to perform physical, repeatable tasks. Machine learning algorithms enable the cobots to learn the physical procedures necessary for automation, allowing for faster and safer test operations. Cobots have been utilized in cockpit testing to automate pilot procedures that are required for many certification tests.

Artificial Intelligence and Machine Learning both show tremendous potential to add benefit across a range of systems and industrial settings. However, to-date most innovations in AI/ML have been incubated in simulated and/or lab environments without a smooth, standards-based path to integration into systems and industrial deployments. A technology gap has existed in order to accelerate AI/ML deployments to the real world. Figure 9 illustrates a new methodology for AI/ML capability to the real world and shows an AI/ML algorithms or model operating within a real-time industrial computing framework. The AI/ML model is imported into the framework as an ONNX format file and is built into a real-time Linux assembly, or application (app) that may be executed on a COTS Linux server (or Windows machine). The AI/ML model includes input and output ports allowing the user to connect data sources to the model input ports and connect model output ports to the desired data consumer(s). The AI/ML model can connect to any data interface that offers Linux driver support, allowing AI/ML models to be flexibly connected to data acquisition cards, Ethernet channels, publish-subscribe protocols, software APIs, and nearly anything that offers a suitable interface.

There exists a number of powerful open source and commercial technologies that may be used to deploy AI/ML applications to the cloud. Platforms like Google's TensorFlow were designed from-the-ground-up to operate in the cloud. However, many of the most interesting and economically valuable AI/ML capabilities industry is looking to adopt, e.g. Digital Twin, require some portion, if not all, of the AI/ML algorithm to be executed with some level of reliable, deterministic real-time performance and be in close proximity to, or embedded within, physical equipment in the real-world. For many of these applications, the goal is to get the AI/ML application as close as possible to data sources, e.g. cameras, vibration sensors, microphones, to enable the AI/ML capability to perform heavy data reduction at the point of data collection. The purpose of this approach is to greatly reduce the infrastructure and operating costs associated with sending all that data to the cloud. For example, an AI/ML algorithm receiving video input and identifying and classifying objects typically uses artificial neural networks to translate from video data to object information. This algorithm can reduce terabytes of video data into a compact set of information such as object positions, object velocities, object count, and object classifications. Although the technology needed to design and implement these AI/ML algorithms is mature and provides an efficient workflow, there has existed a technology gap for the tools needed to install and operate these AI/ML capabilities in the real-world.

Computing and data handling frameworks, such as ADEPT, are designed to handle the challenge of deploying AI/ML capabilities because of their open architecture, high-performance, scalable, real-time computing and data handling approach. The ability to import, with drag & drop or through remote automation, ONNX format AI/ML models into the real-time computing framework eliminates the need for hand-coded deployment of AI/ML models and results in significantly reduced deployment costs. Real-world AI/ML deployment typically involves three distinct phases, as described in the table below.

AI/ML Capability Lifecycle	Overview
Phase 1 – Development & Training	An AI/ML algorithm/model is developed by selecting and refining the architecture, through a series of training and performance assessment cycles, using training and evaluation data sets. This effort is performed using a popular ML framework such as TensorFlow, PyTorch, Caffe2, etc. Output of phase 1 is an ONNX format AI/ML algorithm.
Phase 2 – Deployment & Operation	The next phase of deploying AI/ML capability involves receiving the ONNX format model, importing it into an ADEPT framework, connecting its inputs and outputs to data sources and other applications, building the AI/ML app, and deploying the app to an industrial edge server, or embedded computing platform.
Phase 3 - Sustainment & Adaptation	For most high-value AI/ML applications, the development process doesn't end after the system is deployed. Typical AI/ML capability sustainment will perform an on-going data collection activity used to deliver continuous learning application updates. This phase involves going back to the AI/ML tool of choice, e.g. TensorFlow, running analysis and training activities, generating an updated ONNX model, verifying and validating this model, and deploying the improved model within the system.

XI. T&E as a Cloud-Based Enterprise Service

As an organization matures in their application of open test automation, the load of effort tends to shift through the following stages:

- Developing Open Automation - Shift from time spent performing test manually to developing modular and reusable test automation assets
- Mature Open Automation - Shift from time spent developing test language assets to managing test asset revision and extension, cybersecurity roles and permissions management, and campaign analysis
- As T&E organizations mature, the demand for a cloud-based enterprise T&E automation service grows. An enterprise T&E service manages roles, tool permissions, and data access for the test enterprise, connected with various IT systems using organizational-standard mechanisms and cybersecurity policies. A complete enterprise T&E automation solution has the following two high-level layers:
- Enterprise Layer – Cloud-based test campaign management, user management, test asset management, test analysis, and test-as-a-service platform
- Framework Layer – Connectivity and computing frameworks, on-premises real-time and cloud-based virtual, running on a common computing platform

XII. What do Virtual V&V and Digital Twins Have in Common?

Virtual V&V and Digital Twins are both part of a broad strategy for digital engineering and are both relatively new technologies within the field of digital engineering. Although it wasn't called a DT a decade ago, the use of DT for MBSE System Development, as part of a modernized T&E automation capability, has been around for more than two decades. The application of DT for what-if analysis and live health monitoring, however, are relatively new concepts. But in order to make What-If-Analysis and Live Monitoring DTs a reality, complex systems organizations are likely going to need to leverage assets and technology being developed for their Virtual V&V testing capabilities. The goal of Virtual V&V is to use a digital version of the system to perform valuable testing and evaluation. That digital replica needs to be computationally efficient to support fast-iteration testing and low test cycle times and needs to have appropriately high levels of digital detail and fidelity. What-If-Analysis DTs and Live Monitoring DTs have the same capability requirements as Virtual V&V, and for most complex systems organizations it is a sensible strategy to remove traditional silos and share the Virtual V&V digital model assets as an input to the What-If-Analysis and Live Monitoring DT development and sustainment teams.

XIII. Cybersecurity for T&E Automation

T&E automation software and technologies must comply with and support necessary functions within the applicable cybersecurity standards, with NIST 800-171 used most commonly for aerospace and defense T&E automation applications. Data access control, audit mechanisms, configuration management, user authentication, role based function access, maintenance, system protections, physical protections, and system integrity are aspects of T&E automation that must be considered and accommodated for within:

- Design and implementation of T&E automation software technology
- Project-specific deployment on a T&E automation development

T&E automation software making use of standard Windows and Linux cybersecurity functions and mechanisms can typically be configured to support T&E organization cybersecurity policies.

XIV. Modernizing Airworthiness Testing

The T&E digital transformation methodologies discussed throughout this paper are used across the global commercial aircraft market in varying degrees. Some aircraft manufacturers make heavy and comprehensive use of all the methods discussed, on all active and future aircraft development programs. Other aircraft manufacturers show excellence in one or more areas of T&E digital transformation and less maturity in others. The authors of this paper have been encouraged to see steady investment by the aircraft industry over the past decade to improve the way aircraft are tested, evaluated, and certified. However, we expect to see significant changes in airworthiness standards and guidance coming from organizations such as the FAA and their peers and partners in the coming decade to ensure more consistent use of these highly effective testing methods and technologies. For those organizations well-through their T&E digital transformation, these changes to standards will have little impact. For those with a great digital transformation journey ahead, a significant initial increase in investment spending on test asset and test process modernization can be expected. If done well, initial investment levels can be expected to fall as automation-driven efficiencies begin generating a return on investment.

XV. Conclusions

Investment in digital engineering capabilities, including T&E modernization and automation, is growing at a significant rate within commercial and military complex systems organizations. The US DoD and the US Air Force, in particular, have clearly stated their intention to aggressively modernize and leverage new digital technologies including virtual test methods and DT capabilities. Annual spending on T&E, as part of RDT&E, within the US DoD is estimated to be between $5.3 billion and $21.32 billion spent annual on development testing alone. The adoption of modern T&E automation technology and the addition of advanced methods including Virtual, SIL, and PIL testing offer great potential to achieve dramatic cost savings. Silos have historically existed between development testing and operational testing. Adopting a T&E modernization strategy that removes silos between development testing and operational testing, and removes siloes between each type of development testing, and includes a harmonized T&E automation strategy with maximum reuse of MBSE T&E assets has shown significant cost-benefit within industry-leading commercial aerospace and defense organizations.

A central component of a T&E modernization strategy is the test automation, analysis, and reporting language that provides the bulk of T&E automation capability and allows for the incorporation of advanced technologies including Machine Learning and Digital Twin. Real-time and non-real-time virtual computing and connectivity frameworks provide the glue that connects legacy test assets, advanced testing capability, cloud-based enterprise systems using COTS computer equipment and Windows/Linux operating systems.

The goal of Virtual V&V is to use a digital version of the system to perform valuable testing and evaluation. That digital replica needs to be computationally efficient to support fast-iteration testing and low test cycle times and needs to have appropriately high levels of digital detail and fidelity. What-If-Analysis DTs (type 2) and Live Monitoring DTs (type 3) have the same capability requirements as Virtual V&V, and for most complex systems organizations it is a sensible strategy to remove traditional silos and share the Virtual V&V digital model assets as an input to the What-If-Analysis and Live Monitoring DT development and sustainment teams.

Digital engineering deployments involve leveraging model assets that include valuable intellectual property and therefore must go to extra lengths in the design and implementation of cybersecurity protections. Data access control, audit mechanisms, configuration management, user authentication, role based function access, maintenance, system protections, physical protections, and system integrity are aspects of T&E automation that must be considered and accommodated.

We expect to see significant changes in airworthiness standards and guidance coming from organizations such as the FAA and their peers and partners in the coming decade to ensure more consistent use of these highly effective testing methods and technologies. For those organizations well-through their T&E digital transformation, these changes to standards will have little impact. For those with a great digital transformation journey ahead, a significant initial increase in investment spending on test asset and test process modernization can be expected. However, if done well, initial investment levels can be expected to fall as automation-driven efficiencies begin generating a return on investment.

XVI. References

[1] S.C. James, C. Savaglio, “The Expanded Reach of Simulation Based Aircraft System Verification and its Software Capability Requirements”, AIAA Modeling and Simulation Technologies Confernce, 16-20 June 2014, Atlanta, GA.

[2] B. Hale, “Managing Multi-protocol Network Definitions in Aircraft Simulation and Test”, AIAA 2018-3430, Simulation Architecture and Technologies, AIAA SciTech, 24 June 2018.

[3] E. Carrillo de Albornoz-Braojos et al., “Developing Airplane Systems Faster and with Higher Quality through Model-Based Engineering”, Boeing Research & Technology, Europe. May 2017. URL: https://www.boeing.com/features/innovation-quarterly/may2017/feature-technical-model-based-engineering.page

[4] Department of Defense, Small Business Innovation Research (SBIR) Program X20.1, Air Force 20.1 Special Topics Solicitation, August 2020.

[5] Department of Defense, Small Business Innovation Research (SBIR) Program X20.3, Commercial Solutions Opening (CS) AF Ventures Topics Solicitation, December 2019.

[6] T. Hitchens, “Air Force Plans First Digital Engineering Pitch Day”, Breaking Defense, URL: https://breakingdefense.com/2020/12/air-force-plans-first-digital-engineering-pitch-day, December 2nd, 2020.

[7] FY 2021 Defense Budget. URL: https://www.defense.gov/Explore/Spotlight/FY2021-Defense-Budget/

[8] Statista Market Report. “Global industrial automation software market in 2018 and 2023, by type”, URL:

[9] IoT Analytics. “IoT Platforms: Market Report 2015-2021”, URL: https://investor.ptc.com/static-files/1227d8cb-b383-455d-859c-0174815f86c0

[10] Research and Markets, "Global Automation Testing Market Analysis, Trends, and Forecasts 2019-2025", URL: https://www.businesswire.com/news/home/20191212005665/en/Global-Automation-Testing-Market-Analysis-Trends-Forecasts

[11] S. Zhu, et al., "A formal approach using SysML for capturing requirements in avionics domain", Chinese Journal of Aeronautics, Vol. 32, Issue 12, Dec 2019. Pages 2717-2726.

[12] J, Lasalle, F. Peureux, F. Fondement, "Development of an automated MBT toolchain from UML/SysML Models", Innovations in Systems and Software Engineering, December 2011. 7(4):247-256.

[13] H. West, "AFWERX Is Smart Risk For Innovative Solutions", January 2018. URL: https://www.afwerx.af.mil/stories/afwerx-smart-risk.html

[14] J. Judson, "Fixing relationships: How US Army Futures Command is working with small biz, academia", July 2019. URL: https://www.defensenews.com/land/2019/07/22/can-futures-command-change-the-armys-relationship-with-academia-and-small-business/

[15] M. Bold, M. C. Roth, "A New Era of Acquisition", January 2020. URL: https://asc.army.mil/web/news-alt-jfm20-a-new-era-of-acquisition/

[16] Applied Dynamics International, "Understanding the Real-Time Executive Daemon (rtxd) for Linux", May 16, 2019.

[17] Applied Dynamics International, "Cyber-Physical Product Dev using Industrial Linux Servers – Aircraft Case Study", June 21, 2019.

[18] B. Marr. The 5 Biggest Technology Trends Disrupting Engineering and Design in 2020. Forbes Magazine, Nov. 2019. [Online]. URL: https://www.forbes.com/sites/bernardmarr/2019/11/22/the5-biggest-technology-trends-disrupting-engineering-and-design-in2020/#5d754a3a676a

[19] J. Moyne, E. Balta, I. Kovalenko, Y. Qamsane, and K. Barton, "The digital twin in the manufacturing ecosystem of the future," 1st International Workshop on Smart Manufacturing Modeling and Analysis, 2019.

[20] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," The International Journal of Advanced Manufacturing Technology, vol. 94, no. 9-12, pp. 3563–3576, 2018.

[21] Fei Tao, Fangyuan Sui, Ang Liu, Qinglin Qi, Meng Zhang, Boyang Song, Zirong Guo, Stephen C.-Y. Lu & A. Y. C. Nee (2019) Digital twin-driven product design framework, International Journal of Production Research, 57:12, 3935-3953, DOI: 10.1080/00207543.2018.1443229

[22] Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. Systems 2019, 7, 7.

[23] J. Soldatos, "Digital Twins: What are They and How are They Shaking Up Enterprise Maintenance?", URL: https://www.prometheusgroup.com/posts/digital-twins-what-are-they-and-how-are-they-shaking-up-enterprise-maintenance#:~:text=Digital%20Twins%20are%20also%20used,most%20effective%20asset%20management%20strategy.

[24] General Electric, ''The digital twin: Compressing time to value for digital industrial companies,'' Gen. Electr., Boston, MA, USA, White Paper, 2018.

URL: https://www.ge.com/digital/sites/default/files/download_assets/The-Digital-Twin_Compressing-Time-to-Value-forDigital-Industrial-Companies.pdf

[25] E. Balta, D. Tilbury, K. Barton, "A Digital Twin Framework for Performance Monitoring and Anomaly Detection in Fused Deposition Modeling", 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE).

[26] K. M. Alam and A. El Saddik, ''C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems,'' IEEE Access, vol. 5, pp. 2050–2062, 2017.

[27] K. Ding, F. T. S. Chan, X. Zhang, G. Zhou, and F. Zhang, ''Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors,'' Int. J. Prod. Res., vol. 57, no. 20, pp. 6315–6334, Oct. 2019.

[28] P. Aivaliotis, K. Georgoulias, and G. Chryssolouris, ''The use of digital twin for predictive maintenance in manufacturing,'' Int. J. Comput. Integr. Manuf., vol. 32, no. 11, pp. 1067–1080, Nov. 2019.

[29] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, ''Digital twin in industry: State-of-the-art,'' IEEE Trans. Ind. Informat., vol. 15, no. 4, pp. 2405–2415, Apr. 2019.

[30] G. Hilton, J. Needham., "Welcome to Boeing's factory of the future", Boeing Sheffield. URL: https://www.boeing.com/features/innovation-quarterly/feb2019/btj-global.page

XVII. Acknowledgments

Authors would like to thank the Barton-Tilbury Research Institute for collaboration on Digital Twin technology and architectures, Machine Learning technology, and more.