2020

ExxonMobil

&

CPLANE.ai

Orchestration of an Open Industrial Control System

THE FOUNDATION FOR CONTINUOUS OPERATIONS

STEVE BITAR	R&D PROGRAM MANAGER	EXXONMOBIL RESARCH & ENGINEERING
JOHN CASEY	CTO & COFOUNDER	CPLANE.AI
DAVID DEBARI	INSTRUMENT ASSOCIATE	EXXONMOBIL RESEARCH & ENGINEERING
BRANDON WILLIAMS	COFOUNDER	CPLANE.AI

Table of Contents

Executive Summary

ExxonMobil and CPLANE.ai, a Silicon Valley software company, created a joint initiative to demonstrate the benefits of system orchestration to manage the digital lifecycle of industrial automation systems. As IT/OT Convergence and Industrial Internet of Things (IIoT) adoption accelerates, new system management challenges must be addressed. This pilot shows how orchestration can simplify the management of these new, multi-vendor systems while improving security and reliability.

Pilot Goals

- Demonstrate the "art of the possible" with system orchestration technology; specifically, to automate the provisioning and initiation of a multi-vendor, open system.
- Reduce the complexity and effort of deploying an Industrial Control System (ICS) by orders of magnitude from conventional methods.
- Perform a full ICS deployment with little to no IT expertise.
- Conform as much as possible to open standards and practices as defined by the Open Process Automation™ Standard (O-PAS).
- Demonstrate the suitability of standards such as OPC-UA, DMTF Redfish, and OASIS TOSCA in an open ICS environment.

Pilot Overview

The pilot consisted of a heterogeneous mix of IT/OT technologies simulating a chemical processing plant. The system started with industrial IT compute devices (replacing legacy DCSs and PLCs) powered-on and connected to an Ethernet network as well as being connected to specific I/O networks (based upon the design of the plant). The CPLANE.ai Industrial Orchestrator was used to fully install and configure all process control software and control logic to bring the simulated plant to an operational state. This was all performed in approximately *ten minutes* with just a few clicks of the mouse compared to an estimated 50-100 person-hours by conventional methods.

Key Findings

- Multi-vendor, open process automation can be integrated into a wholistic system using system orchestration technology.
- Open standards make inter-operability of a heterogeneous system much easier to manage and more reliable to implement and operate.
- Integration with current OT solutions requires cooperation and collaboration from vendors but is achievable with spectacular results. An integrated hybrid-architecture of IT/OT digital assets can be managed in one cohesive framework.
- System orchestration is critical for accelerating innovation and adoption of converged IT/OT systems, particularly in an open, multi-vendor and interoperable control system.

Purpose and Goals of The Orchestration Pilot

This pilot was conceived and executed in the context of rapid changes in industrial automation. Large industrial manufacturers like Merck, Dupont, Shell, BASF, Georgia Pacific, and ExxonMobil are making new demands on traditional process automation vendors for converged IT/OT solutions (the unification of Information Technologies and Operational Technologies into a single system of management and control) that are open, interoperable, and inherently secure. The standards for these Open Systems are being developed by global standards bodies like the Open Process Automation™ Forum and NAMUR.

The specific purposes and goals of the pilot are as follows:

1. Demonstrate the automated provisioning and initiation of a multi-vendor, converged IT/OT control system from a pre-software-installation state, to a fully operational and ready state – the "Startup Phase."

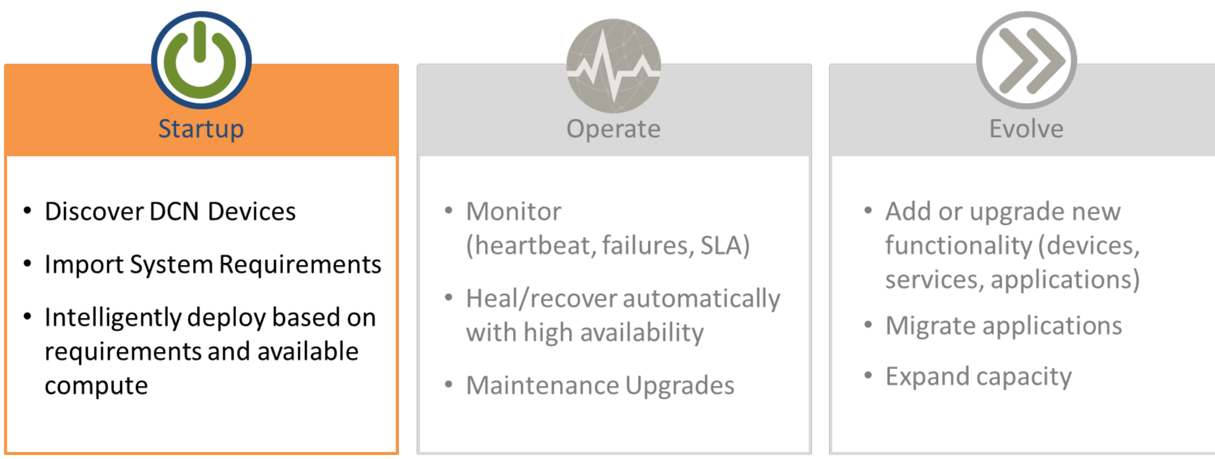

Figure 1: Three phases of plant lifecycle are managed by system orchestration. This pilot focused on the Startup Phase since it provides foundational capabilities required for the Operate and Evolve Phases.

2. Demonstrate orders of magnitude reduction in the complexity and effort required to perform this "Startup Phase" of both IT and OT systems when compared to existing, manual software installation processes.
3. Perform this "Startup Phase" with little or no IT expertise from the perspective of the system operator.
4. Prove that an orchestration platform can provide hands-off deployment of a converged IT/OT system that conforms to the open, multi-vendor specifications outlined in the Open Process Automation™ Standard.

5. Prove that DMTF Redfish can reliably provide physical system meta-data to facilitate correct deployment of OT software applications.
6. Specifically demonstrate the use of Cloud technologies and standards like OASIS TOSCA for orchestration.

What is "System Orchestration?"

System orchestration is the art and science of providing "digital life cycle management" to large, complex systems. It has long been used in large-scale telecommunications networks, sophisticated enterprise data centers, and web-scale data centers. Orchestration tools and techniques are used to allow a single operator, or even no operator, to manage huge and complex digital infrastructures. Industrial orchestration manages all compute elements, software stacks, control applications, networks, and containers as a single, integrated system. As next-generation industrial control systems transition to a rapidly maturing and increasingly complex digital technology stack, system orchestration customized for industrial systems is a key element in delivering "system-ness" from open components from multiple vendors.

Introduction to CPLANE.ai

ExxonMobil collaborated with CPLANE.ai, a Silicon Valley software company, to utilize CPLANE.ai's advanced orchestration platform for this pilot demonstration. Its orchestration software is used to solve complex digital lifecycle problems in managing large Edge Computing, Distributed Clouds, and Industrial Internet of Things (IIoT) environments and has been customized to meet the unique needs of industrial systems. CPLANE.ai has been actively involved in the Open Process Automation™ Forum over the last three years and is a leading contributor to the Open Process Automation™ Standard.

The Pilot Plant Description (the Digital Infrastructure)

The pilot plant for this demonstration simulates a chemical mixing and heating process involving several plant assets: Reactor Batch Processor, Heat Exchanger, Product Storage Tanks, and a Water Chiller.

The pilot infrastructure for this demonstration consists of 14 individual compute devices that represent 13 "Distributed Control Nodes" (DCN's), and a single "Advanced Computing Platform" (ACP). The DCN's run the control loops for the industrial process while the ACP hosts the Human Machine Interface (HMI) application as well as the IEC-61499 engineering design tool[1]. The compute devices are a heterogeneous combination of different microprocessors (Intel X86 and ARM), with different configurations of RAM and storage, from different manufacturers. These devices are also divided between two locations, approximately half of the compute devices are in New York and the other half in California, to represent a truly distributed topology.

The compute devices are connected on the northbound side to an Ethernet network, while the southbound side is connected to a (simulated) Fieldbus network with simulated sensors and actuators. The two operational sites (CA and NY) are connected through a VPN connection via a third site, the CPLANE.ai datacenter in California. The entire digital infrastructure was managed (orchestrated) from a single location in California.

The Pilot Demonstration

The goal of the pilot is to use automation (via the CPLANE.ai Industrial Orchestrator) to correctly install all the software necessary to deploy the control system of a simulated chemical plant. The CPLANE.ai Orchestrator will complete this "Startup Phase" operation in

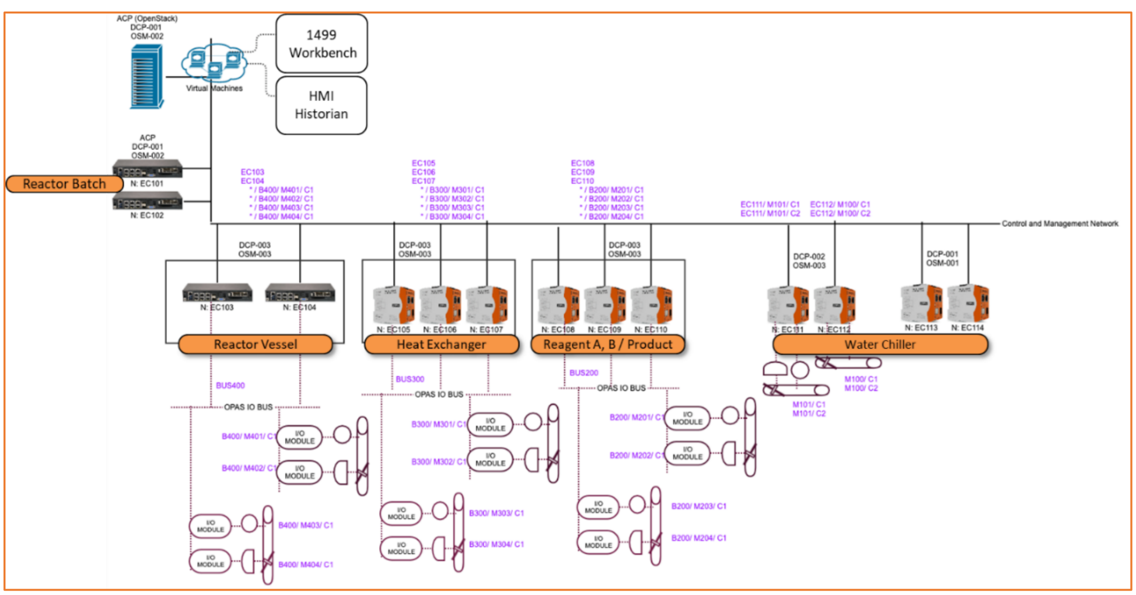

Figure 2: The topology of the pilot infrastructure.

[1] Schneider Electric https://blog.se.com/machine-and-process-management/2020/05/06/is-iec-61499-the-missing-link-for-industry-4-0/

approximately *ten minutes*. Manual installation of a similar system often takes a team of 2-3 engineers *several days* or a week.

1. The initial state of the control system:
    - Sensors and actuators (simulated) are connected via an I/O bus or directly to the DCN's.
    - IEC-61499 function blocks have already been created and compiled for the process in the engineering design tool.
    - All DCN & ACP devices are connected to an Ethernet network for both the control and data planes.
    - All DCN & ACP devices are powered on and are preloaded with Linux and DMTF Redfish client (simulated in this demonstration with CPLANE.ai client software).
    - The CPLANE.ai Orchestrator is running on the same control network as the process automation system on a dedicated X86 based device running Linux.
    - The IEC-61499 engineering design tool is connected to the CPLANE.ai Orchestrator.
2. The demonstration begins with the HMI running, but showing that it is not receiving sensor data nor does it have knowledge of the digital infrastructure underlying the control system.
3. In the **first step** of the three-step deployment process, the operator initiates discovery of the digital infrastructure with a single mouse click in the CPLANE.ai Orchestrator. The Orchestrator then polls all the digital devices on the control network, registering each device into the Orchestrator's information model database. Additional details of each digital device are ingested into the Orchestrator's information model using the DMTF Redfish protocol as well as other standard IT discovery protocols. This discover process takes approximately 90 seconds to perform.
4. The Orchestrator then builds a digital topology model of the physical infrastructure including in-depth data describing the physical infrastructure and its state. This topology model will be used to make intelligent decisions on exactly where each software element of the control system will be installed. Now, the Orchestrator effectively has a "digital twin" of the control system's digital infrastructure in its memory.
5. The CPLANE.ai Orchestrator is then able to display all of the digital infrastructure of the control system including rich detail for every connected device. Examples of details in the information model include:
    - Microprocessor type and manufacturer

- Device manufacturer and model number
- Available device RAM and storage
- Utilized and available CPUs
- Device IP address
- Network zone
- OS type and version

6. The **second step** of the three-step deployment process is to "program" the CPLANE.ai Orchestrator with all of the requirements for the control system's "digital life-cycle." This "programming" step is performed by uploading one or more OASIS TOSCA documents that contains all the system requirements in a structured, reusable data format (YML). Examples of the requirements in a OASIS TOSCA[2] document includes:
    - Device requirements for software applications (e.g. microprocessor type, available RAM, OS type and version, etc.)
    - Policies that govern system life cycle (e.g. software applications that are prerequisites for other software applications, high-availability requirements , security zone requirements, affiliations with specific I/O locations, etc.)

7. The Orchestrator is now programmed with the "system engineering knowledge" in its memory; and, using sophisticated algorithms, can make intelligent decisions on which application should be installed on which device. And, equally as important, the Orchestrator understands exactly which steps must be taken in the correct order to install all of the software for the entire system in an automated and deterministic manner.
8. A converged IT/OT system like a modern, open process automation system can only be effectively and efficiently managed within a wholistic framework. The orchestration platform has merged all the requirements, policies, procedures, workflows, and state-management into a single fabric. This single management fabric is now able to take action in multiple dimensions (detailed below) with a deep understanding of dependencies and interdependencies of the entire digital infrastructure including:
    - Supervisory applications (HMI, Historian)
    - Process control applications
    - Compute layer (OS, Docker, virtual machines)
    - Network security
    - Networking

[2] OASIS TOSCA Overview https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca#overview

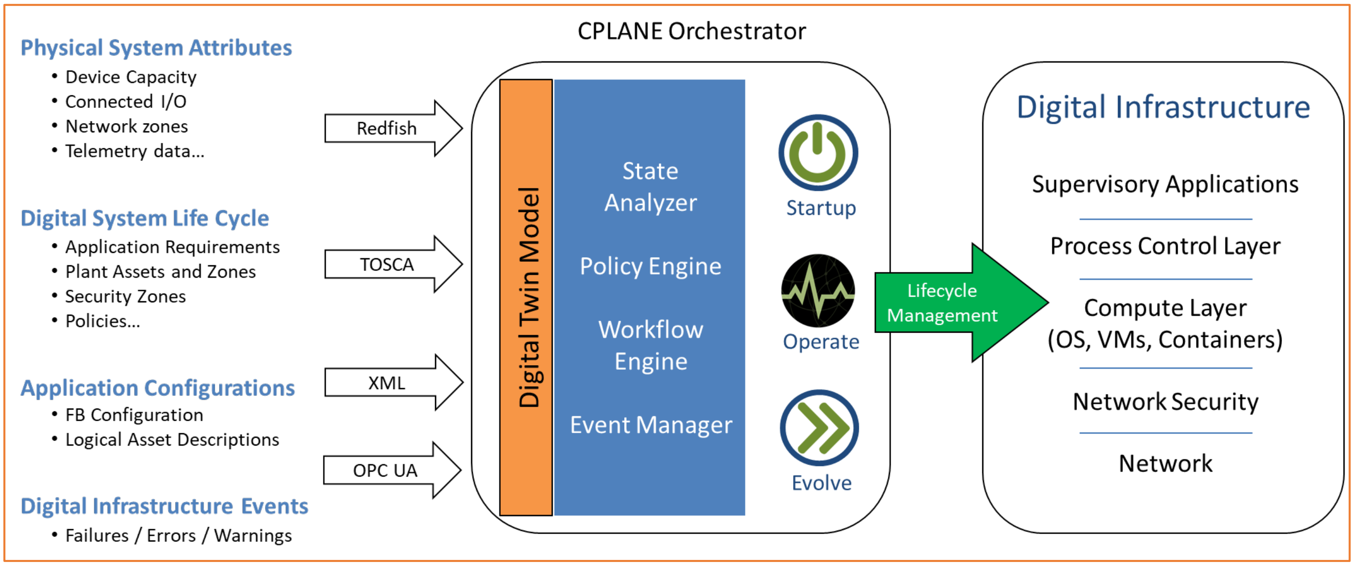

Figure 3: The CPLANE.ai Orchestrator receives input from multiple sources and manages multiple layers of infrastructure as a single fabric.

9. The **final step** is to activate the Orchestrator to deploy the ICS software. The operator simply pushes the “Deploy” button and the provisioning and installation operations commence.
10. The Orchestrator, in parallel, performs the following actions on all of the DCN’s and ACP nodes of the system:
    - Examines the current state of each device as well as the entire system.
    - Calculates the final desired state of each device and the whole system.
    - Evaluates the policy constraints (the “rules”) of each device (networking, security, prerequisites, etc.).
    - Calculates all the steps necessary to deploy the entire system without violating any of the constraints.
    - This calculation then programs a workflow engine to execute each step both in sequence and in parallel where permitted.
    - The workflow engine (internal to the Orchestrator) takes actions such as:
        i. calls premade scripts (like Ansible and Python) to direct installation routines
        ii. dynamically creates custom scripts where needed to instantiate a change in state of a device
        iii. cleans up devices to allow for new software installations

- Simultaneously, the Orchestrator monitors the digital infrastructure for events that signal successful actions taken, or unexpected errors that require the Orchestrator to take action or perhaps notify the operator.

11. Some of the actions performed on the system by the Orchestrator include:
    - Deployment of OPC-UA client and server instance on 12 DCN's
    - Deployment of the OPC-UA discovery services (LDS) on a single DCN
    - Deployment of Docker containers on 12 DCN's that will host the engineering runtime
    - Deployment of the engineering runtime on 12 DCN's (in their respective Docker container)
    - Directing the 61499 engineering design tool to install function blocks on specific DCN's once the engineering runtime environment is ready
12. This deployment process takes approximately *five minutes* to perform. The end result is a fully deployed control system.
13. Switching back to the HMI, the first thing that is noticeable is that there are values in the sensor displays showing that OPC-UA has been installed correctly, is connected to the right I/O, and is successfully sending data to the HMI.
14. Using the HMI, the (simulated) chemical process is started. Each DCN successfully executes its function block code to make the chemical process operate as expected.
15. This whole "Startup Phase" operation takes approximately *ten minutes* from the initial state as described above to a fully operational state where the HMI is able to initiate control functions.

Engineering Principles Applied to Pilot Demonstration

- System startup (the "Startup Phase") is not the end goal of using system orchestration. The primary goal is to maintain continuous plant operations during change, failure, or upgrade as specified by digital-lifecycle policies. The "Startup Phase", demonstrated in this pilot, lays the foundation for future phases to demonstrate continuous plant operations.
- Automated management of complex digital systems has been done before. Converged IT/OT control systems must follow the success of other industries' successful management of large, complex systems. Orchestration has been widely adopted to manage the lifecycle of global telecommunications networks, automate the world's largest Cloud data centers, and to accelerate the deployment and maintenance of software applications for every large business.

- Do not just “make it work”; to avoid costly “dead ends,” choose a standards-based framework that is designed for industrial automation, rather than the currently “hot” technology.

Key Learnings from the Pilot

1. System-ness for Open Systems is achievable using system orchestration. Multi-vendor, open process automation can be integrated into a wholistic system using the technologies and techniques of orchestration.
2. Open standards make inter-operability orders of magnitude easier to manage and more reliable to implement. These open standards will continue to evolve as technologies evolve; helping to future proof your investment.
3. Integration with current OT solutions requires cooperation and collaboration from vendors but is achievable with spectacular results. Shift thinking from IT and OT, to an integrated hybrid-architecture of IT/OT digital assets managed in one cohesive framework.
4. System orchestration is critical for accelerating innovation and adoption of converged IT/OT systems.