ADDING PREDICTIVE CAPABILITY TO THE O-PAS TECHNICAL STANDARD

Date: January 2021

Objective: Provide recommendations that enable a predictive capability to the O-PAS Technical Standard.

Why: The ability to reason about critical quality attributes (performance, safety, security, etc.) before hardware and software components are acquired from the marketplace can reduce testing and commissioning time. With predictive capability, this can occur virtually and confidently as O-PAS conformant control systems can be shown in the specification of components, their assembly and integration to achieve the high-level requirements of the end user as the control systems are designed and validated ***before*** acquisition commitments. That means, that when the actual control systems are acquired and put into place, test and commissioning becomes a verification activity of that which *was designed* rather than "*as built*".

How: Predictive capability requires all aspects of a target, end-user control system to be specified. Such specificity would ultimately be extended to not only the hardware of the control system (actuators, human-machine interfaces, networks, compute platforms) but also to the underlying software frameworks to which the components must operate and other relevant software "touch points" for the end-to-end threads of critical control. The O-PAS technical standard would need to be "open" itself, to the expression of such properties that would support reasoning about a variety of quality attributes as well as the integration of reasoning capabilities that act on those properties to result, or predict that outcome of such reasoning so that it can be compared to the high-level, end user requirements.

What: Proposed is an exploratory step to first understand the boundaries O-PAS technical standard as it pertains to incorporating a predictive capability (e.g., expression of component properties such as behavior, timing, expression of O-PAS conformant assemblies of control system's components (hardware and software), and the incorporation of '*reasoning frameworks*' which can be instantiated (and perhaps productized) to confidently reason about specific end user quality attribute requirements from such assemblies of O-PAS components). The result of this exploratory step would be recommendations that comprehend an "open" predictive capability to the O-PAS Technical Standard for the committee to consider.

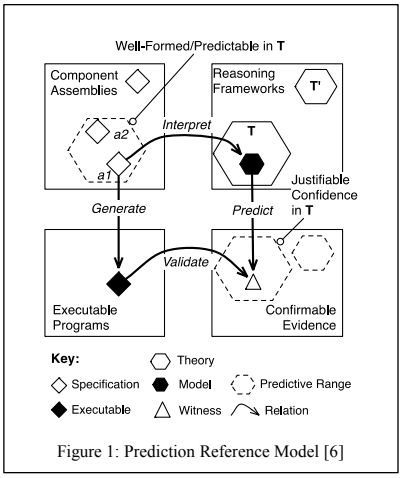

Figure 1: Prediction Reference Model [6]

Methodology: Current, and prior, SEI work in model-based software engineering [1][2], system and architectural analysis [3], and predictable techniques [4][5] will serve as a basis for the proposed exploratory work. Figure 1 illustrates the techniques that would be applied in the proposed work. Component assemblies (upper left of Figure 1) have been encoded in AADL and PECT in a number of domains. That encoding not only expresses the end user application, but also the properties of the constituent hardware and software components. The properties are those needed for the quality attributes critical to the end user application. From there (moving to the upper right of Figure 1), the assemblies and properties were interpreted by reasoning frameworks to make predictions. In SEI work, those same

component assemblies (in the upper left) served as input to generative technology to create executable programs (in the lower left) that would be deployed in a test or operational environment as a means to validate the actual behavior of the end user application. Actual observations that fell within the predictive range emminating from one or more reason frameworks (moving to the lower right of Figure 1) of the *designed* end user application would serve a objective, confirmable evidence that the user application as designed behaved as expected. It is important to point out that the prediction itself was made without any executable programs in hand. The key to making a quality attribute prediction is to know, *a priori*, the salient component properties of the constituents of the assembly used for the end user application.

Approach: For this exploratory project, O-PAS, itself, would serve as the means for encoding end user applications as component assemblies along with the needed properties of those components sufficient for interpretation in reasoning frameworks to make prediction(s). Targeted for the end user application would be a portion of a real-time monitoring thread of a Separation Tower simulated in the OPA Test Bed. The critical quality attribute for this end user application could be the latency of observed, monitored values of values within the tower processes (for instance the values displayed on the HMI appear no later than some tolerance). Components expressed in the component assembly would include those at all tiers of the OPA [8] (to include, but not limited to DCNs, ACPs, PLCs and other relevant "touch points" in the OPA for the end user application). To the extent possible, reasoning frameworks from prior SEI work [7] would be reused to generate predictions for the critical quality attributes of the end user application. Although actual implementation generation is beyond the scope of this exploratory project, it is expected that runtime measurements of components (simulated or not, virtualized or not) to obtain actual values within the OPA Test Bed would be captured and used in a validation exersize to compare with the prediction(s) results from the reasoning frameworks.

Output: The exploratory project would serve as a basis to support a working hypothesis that the O-PAS can serve as a means to not only support the construction of process automation applications in an open-systems manner, but it can also support the expression of important properties of components used in process automation applications. The output from this project would identify the gap between what is today's O-PAS standard and future versions of O-PAS that would enable predictive capabilities for users to predict the critical quality attributes of end user applications based on those specified in O-PAS compliant components before cost and effort is expended in commissioning such applications in the field.

Envisioned Steps:

1. Define concrete success criteria for the exploratory project
2. Define the demonstration OPA Test Bed end user application (e.g., Separation Tower Monitoring)
3. Define the critical quality attribute(s) of the end user application
4. Identify the supporting reasoning framework(s) for prediction(s)
5. Build the end user application component assembly specification (upper left of Figure 1)
6. Generate prediction(s) from reasoning framework(s) through interpretation of end user application component assembly specification (upper right of Figure 1)
7. Build (if necessary), instrument, and run the end user application in the OPA Test Bed (lower left of Figure 1)
8. Evaluate OPA Test Bed observation(s) with generated prediction(s) to validate "as built" end user application
9. Generate recommendations for the O-PAS Technical Standard

References

1. D. Gluch, J. Brockway, "An Introduction to Software Engineering Practices Using Model-Based Verification," CMU/SEI-99-TR-005, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=13371
2. G. Lewis, L. Wrage, "Model Problems in Technologies for Interoperability: Model-Driven Architecture," CMU/SEI-2005-TN-022, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005, https://resources.sei.cmu.edu/asset_files/TechnicalNote/2005_004_001_14510.pdf
3. P. Feiler, D. Gluch, "Model-Based Engineering with AADL: An Introduction to the SAE Architecture Analysis & Design Language," Addison-Wesley, ISBN: 9780321888945, 2012, https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=30284
4. K. Wallnau, "Obtaining the Benefits of Predictable Assembly from Certifiable Components (PACC)," White Paper, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005, https://resources.sei.cmu.edu/asset_files/WhitePaper/2005_019_001_29342.pdf
5. S. Hissam, M.Klein, J. Lehoczky, P. Merson, G. Moreno, K. Wallnau, "Performance Property Theories for Predictable Assembly from Certifiable Components (PACC)," CMU/SEI-2004-TR-017, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2004, https://resources.sei.cmu.edu/asset_files/TechnicalReport/2004_005_001_14411.pdf
6. K. Wallnau, "Predictability By Construction: Working the Architecture/Program Seam," Dissertation, Mälardalen University, School of Innovation, Design and Engineering, ISBN: 978-91-86135-80-5, Västerås, SE, 2010, http://www.ivica-crnkovic.net/phd/KurtWallnau/theSeamMain-04-07.pdf
7. L. Bass, J. Ivers, M. Klein, P. Merson, "Reasoning Frameworks," CMU/SEI-2005-TR-007, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005, https://resources.sei.cmu.edu/asset_files/TechnicalReport/2005_005_001_14588.pdf
8. The Open Group, "Open Process Automation Forum," The Open Group, 1995-2021, https://www.opengroup.org/forum/open-process-automation-forum

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM21-0049