

## ARCHITECTURE

This checklist covers designs at the architecture level. The term element is used in this document to mean any sub-system, component, module, or other design element the architecture covers. See *CxStand\_Design* for a discussion of different design levels in CxOne.

### General

- ARC-1 Does the architecture conform to *CxTemp\_Architecture*?
- ARC-2 Does the architecture conform to *CxCheck\_Design*?
- ARC-3 Does the architecture communicate an adequate vision of the system that will direct further design activities?
- ARC-4 Is the architecture well organized and provide a concise system overview, background information, constraints, and a clear organizational structure for all downstream designs?
- ARC-5 Is the architecture designed to accommodate likely changes?
- ARC-6 Does the architecture stay above detailed design and user interface design activities?
- ARC-7 Do the dependencies between different architectural views hold together?

### Architecture Design

- ARC-8 Is the system architecture, including the data flows, control flows, high-level elements, and interfaces, clearly represented?
- ARC-9 Are fine grain element details omitted and left to subsequent design artifacts?
- ARC-10 Does the architecture cleanly decompose the top-level elements of the system?
- ARC-11 To the extent possible, is the architecture independent of the technology that will use used to implement it?
- ARC-12 Does the architecture take into account technology or other constraints that cannot be avoided, i.e., can the architecture be implemented for the target environment?
- ARC-13 Have you used round-trip design, selecting the best of several attempts? Are reasons listed for the discounted alternatives?
- ARC-14 Does the architecture differentiate between the problem-domain, the user-interface, task-management, and data-management? If not, is the lack explained and justified?

### Architectural Features

- ARC-15 Are external interfaces, including user interfaces, defined and justified?
- ARC-16 Is a level of robustness specified and justified?
- ARC-17 Is the architecture appropriately layered?
- ARC-18 Is the architecture loosely coupled and have good cohesion?
- ARC-19 Does the architecture set resource budgets for each element?

### Strategies

- ARC-20 Is an I/O strategy described and justified?
- ARC-21 Is an error-handling strategy described and justified?

### **Quality Attributes**

- ARC-22 Have the required quality attributes been listed by priority?
- ARC-23 Have technical quantitative limits been detailed for the quality attributes?
- ARC-24 Have technical solutions been identified to support the quality attributes?
- ARC-25 Have trade-offs been identified and accepted for quality attributes solutions?

### **Project Level Support**

- ARC-26 Does the architecture support the lifecycle selected by the project?
- ARC-27 Are the major system goals clearly stated?
- ARC-28 Is the system environment defined, including hardware, software, and external systems?
- ARC-29 Are necessary buy vs. build decisions included and justified?