Candidate Solution

Go up to Terms and Definitions

Discussion

A Candidate Solution identifies a proposed means of satisfying a defined need that has been selected for evaluation.

A Candidate Solution can consist of or identify:

- A Distributed System
- A Node
- A Node Set
- A service
- A software component
- An Executable Artifact
- A hardware component
- A product
- A technology
- An architecture
- An implementation
- A Configuration
- A proposed modification
- A combination of components, resources, configurations, and services
- Another proposed means of satisfying the defined need

A Candidate Solution can identify:

- Its identity
- Its name
- Its description
- Its solution type
- Its intended purpose
- The defined need it proposes to satisfy
- Its provider
- Its owner
- Its applicable Version
- Its applicable Configuration
- Its constituent components
- Its applicable Nodes
- Its applicable Node Sets
- Its interfaces
- Its applicable Endpoints
- Its required Dependencies
- Its required Operational Resources
- Its applicable Deployment Targets
- Its evaluation characteristics

- Its applicable Test Definitions
- Its applicable Test Environments
- Its applicable Acceptance Criteria
- Its applicable Governance Policies
- Its evaluation status
- Its Evidence
- Its Provenance
- Its Traceability

A Candidate Solution becomes a selected Candidate Solution when an applicable Authority, Governance Policy, or Acceptance Criteria selects it for evaluation.

Selection for evaluation does not constitute approval, acceptance, acquisition, authorization, certification, or selection for deployment.

Definition

proposed means of satisfying a defined need that has been selected for evaluation

Source

Adapted from:

- Requirement
- Test Item
- Acceptance Criteria
- DIDO Reference Implementation Conceptual Model
- DIDO-TE draft Requirements Register
- DIDO-TE Broad Agency Announcement source material

Note

A Candidate Solution differs from a Test Item:

- A Candidate Solution identifies a proposed means of satisfying a defined need
- A Test Item identifies an item subject to testing

A Candidate Solution selected for testing can be a Test Item. A Test Item is not necessarily a Candidate Solution.

A Candidate Solution differs from a Distributed System:

- A Candidate Solution is defined by its status as a proposed means selected for evaluation
- A Distributed System is defined by its distributed structure and behavior

A Candidate Solution can be a Distributed System, a part of a Distributed System, or a proposed

change to a Distributed System.

A Candidate Solution differs from a Node:

- A Candidate Solution is the subject of an evaluation
- A Node is an architectural participant in a distributed system

A Candidate Solution can be realized by one Node, multiple Nodes, a Node Set, or a modification to a Node.

A Candidate Solution differs from a Configuration:

- A Candidate Solution identifies the proposed means under evaluation
- A Configuration identifies the arrangement and settings applicable to an item

A Configuration can form part of a Candidate Solution or distinguish one Candidate Solution from another.

A change to any of the following can create a distinct Candidate Solution when the applicable evaluation rules treat the change as material:

- Version
- Configuration
- Constituent component
- Dependency
- Interface
- Deployment Target
- Operational Resource
- Architecture
- Implementation
- Security control
- Governance constraint

The applicable Acceptance Criteria should establish the identity boundary of each Candidate Solution and identify which differences create a separate Candidate Solution.

A Candidate Solution can be:

- Proposed
- Selected for evaluation
- Under evaluation
- Evaluated
- Rejected
- Recommended
- Approved
- Accepted

These possible lifecycle conditions should not be treated as interchangeable. The applicable Governance Policies determine the permitted lifecycle conditions and transitions.

A favorable comparative evaluation does not by itself approve a Candidate Solution.

Approval, acceptance, acquisition, authorization, certification, and deployment require the applicable

Authority and controlling Governance Policies.

Example

An organization evaluates three implementations of a transaction-validation Node.

Each implementation:

- Provides the same required interface
- Processes the same Test Items
- Uses the same Test Argument Values
- Operates within an equivalent Test Environment
- Is evaluated using the same Test Definitions
- Is evaluated according to the same Acceptance Criteria

Each implementation is a separate Candidate Solution because it provides a different proposed means of satisfying the transaction-validation need.

DIDO-TE associates each Candidate Solution with its applicable Version, Configuration, Test Runs, Test Results, Evidence, Provenance, and Traceability.

The resulting comparative evaluation identifies relative differences among the Candidate Solutions but does not by itself approve or select one for deployment.
