Abstract Composite Knowledge Artifacts

Some Desirable characteristics

- Support for Heterogeneous content that is open ended and can evolve as technology does.
- A model that supports execution as the target level of representation to insure that the interpretation of knowledge can be uniform.
- Support for a full lifecycle include multiple layers of authoring, execution, deployment, and replacement.
- Support for multiple types of governance including authoring, certification, testing, and acceptance.
- Support for a packaging model that is physically separate from relationships between the components. This allows multiple relationship structures (e.g. trees and networks, etc.) to coexist without undue burden and avoids conflating the physical relationships of the components with their various logical relationships.
- Support for a multiple physical representational forms in term of both their physical and logical expression. In particular the ability have
- Allow artifacts in the same physical bundle or have the included by reference. Likewise the ability to define the same logical artifacts in different encodings such that a consumer code choose the best implementation for their environment (e.g. A java library vs. a c# library or the correct value set based on the code set used).
- Mechanisms that allow compositions to be formed that allow to composer to resolve execution, interpretation, and information conflicts. This envisions behavioral encapsulation and clearly define communications channels being essential. This includes concepts such a formal input and output, events, activation lifecycle, context binding, information sharing, and event propagation.
- To have a model that supports evolving bindings to new and emerging technologies.
- Ability to due automated integrity checking of a composite to insure that it still fulfills its intended use.
- Ability to override the intended use of an individual component.
- Have a system that facilitates and promotes reuse.
- Ability to use identifiers that are local to a composite. This allows the development of role based logics and facilitates reuse.
- Support commercial knowledge creation by supporting digital rights management and features such as service integration and opaque runtime implementations.