

An Interface Transition Approach

Beyond Functional Equivalence

Ms. Patrice A. Lyons
Workshop on DLT scalability
and interoperability
International Telecommunication Union
Geneva, Switzerland
August 2, 2019

Le retour by René Magritte, 1940

Musées royaux des Beaux-Arts de Belgique, Bruxelles / photo : J. Geleyns



Example of Static Form of Expression

Dynamic Form of Expression

- Where the art work, *Le retour*, is incorporated in a video game computer program called *The Doves of Peace Negotiators*, the work may be represented in digital form without change: the functional equivalent; however, it may also be altered to add new features when the program is executed by players: capable of behavior when processed.
- The Doves may fly, have encounters with other game elements, and move beyond the clouds to different worlds. The computer program may also be updated or altered to increase the speed of play, to permit the introduction of new story lines, or to enable players to interact with others to carry out negotiations in the Internet or other computational environments.
- *Le retour* has now become dynamic rather than static: it has moved beyond the functional equivalent of the original art work expressed with paint and fixed on canvas.

Documents Fixed on Paper

- Like a work of art on canvas, there are numerous legal documents traditionally fixed on paper, where consideration is being given to the representation of the data or other information in digital form that may be dynamic. Examples include: warehouse receipts, bills of lading, checks, securities, and bonds.
- Take the Certificate of Title to a vehicle where information about the ownership, model, vehicle identification number (VIN), and distance traveled are recorded in an official registry. If this title information is represented in digital form and structured as a digital object (aka digital entity), a user, if authorized, may interact with this record to update the information – perhaps to note when the vehicle is sold.
- The official record must be protected from unauthorized tampering or hacking—say the odometer reading is changed. If the digital entity is encrypted using public key technology, certain risks may be mitigated, particularly where the digital entity is part of a linked chain of title going back to the manufacturer.

Role of Transition Spaces

The introduction of a transition space for protection and other purposes has proven useful in many situations.

Consider the airlock. It is a small cylindrical area located on a spacecraft's mid-deck. An astronaut dons a space suit inside and the pressure is gradually decreased to the point where the astronaut goes through the outer airlock hatch for an Extravehicular Activity (EVA), e.g., a space walk. This transition space serves to protect the astronaut from the effects of the environmental differential.



Image credit: NASA
<https://creativecommons.org/licenses/by-nc/2.0>

Security Interface Program

- When a service is provided to a ledger, sometimes called a “DLT oracle” by others, careful security measures are required to prevent fraud or other malicious actions. For a system that allows for interoperability, outside services should only be supported if they can be accomplished securely.
- Rather than allowing direct access to a ledger that may not provide adequate security, an interface program could be generated to serve as a secure transition space where credentials may be authenticated, access permissions verified, and type checking via dry runs on all method calls made by the interface program to the service facility.
- Based on work at CNRI in the 1980s, the notion of a “bastion object” was developed to provide what may be viewed as an interface transition space for security purposes. It was defined in a CNRI patent application, now abandoned, as “an object created by a Knowbot operating system and which establishes a restricted interface to a system object.” (*System for Distributed Task Execution*, Pub. No.: US 2006/0136923 (2006))

Defensive Procedures

- Called a “sandbox for dynamic code,” a bastion for another object—
“the original” has also been described at <https://github.com/enthought/Python-2.7.3/blob/master/Lib/Bastion.py> as *“an object that has the same methods as the original but does not give access to its instance variables. Bastions have a number of uses, but the most obvious one is to provide code executing in restricted mode with a safe interface to an object implemented in unrestricted mode.”*
- While experience in the Python community has shown the need for hardening such interface programs against intrusion by determined hackers, and the module has been removed in Python 3, perhaps new cryptographically-based procedural methods can meet the security challenges previously encountered. The Python 3.0.2 Bastion module is still available at <https://docs.python.org/2/library/bastion.html>.
- Much work is required to enable commerce in dynamic new forms of expression: an interface transition approach may be helpful.