<<Quantum>>

Use Case: applied to Use Case.
Class: applied to Class & Package (Element)
Sequence: applied to Lifeline
Deployment: applied to Artifact, Node, Component (only stereotype included)

A definition of quantum environments where quantum functionality is executed. To distinguish where functionality is performed is based on quantum software. <<Quantum>> Use Cases are related to quantum environments stereotyped with <<Quantum Computer>>, denoted as Actors. <<Quantum>> Use Cases are never directly related to human actors as classical systems typically manage them.

Like classical software components, quantum software components will be designed as classes and packages. These classes and packages define the functionality of creating and developing quantum circuits, algorithms, and related elements. The <<Quantum>> stereotype indicates they are related to quantum software. It can be used for entire packages or individual classes to identify which software parts are intended for quantum computing.

Structure diagrams show the basic structure of quantum software components; behavioural diagrams will show how the components behave and interact over time.

Quantum algorithms are shown as black box elements (details omitted) and are represented as simple classes. Quantum circuits can be modelled via activity diagrams, referenced in the overview diagram and connected to a <<Quantum>> class with a “refine” dependency, showing the activity diagram provides more detail on the behaviour of the quantum class.

<<Quantum>> stereotype is used to distinguish quantum hardware in a deployment diagram and quantum components within the quantum hardware.

Summary:

This stereotype is the most common and denotes any element related to quantum computing. For instance, a quantum algorithm is modelled as a class with the «Quantum» stereotype. Use case diagrams can distinguish quantum use cases from classical ones, although this decision should be made based on prior knowledge about the problem domain.

The «Quantum» stereotype can be used in different UML diagrams to model various static and dynamic aspects of hybrid information systems. For example, class diagrams denote classes designed for quantum software components, and deployment diagrams indicate the hardware used for quantum computations.

<<Quantum Computer>>

Use Case: applied to Actor
Sequence: applied to Actor

Quantum environments that represent software and hardware, and when a use case is performed, should be denoted as an actor with a <<Quantum Computer>> stereotype. They should be connected through <<Quantum>> Use Cases.

Summary:

This stereotype represents the quantum environment where a specific quantum functionality is executed. It is typically associated with an actor element in a use case diagram and connected through associations with «Quantum» use cases. This stereotype allows for the specification of the technology stack, including both software and hardware, involved in the quantum aspect of the hybrid system.

<<Quantum Driver>>

Class: applied to Class
Sequence: applied to Lifeline

Specific classes in the software will act as controllers and manage how quantum (or classical) software components are interacted with; these are called drivers. When applied to a quantum software component, these are represented as class elements with the stereotype <Quantum Driver>>, indicating their role in managing quantum operations.

The same stereotypes defined for classes (Class Diagram) are used for lifelines (Sequence Diagram) for quantum software. The <<Quantum Driver>> lifeline is responsible for creating an instance of a <<Quantum Request>> object, which in turn creates instances of optimisation and cost function objects for managing specific requests to the <<Quantum>> components.

Summary:

This stereotype defines the class drivers that manage invocations to the quantum software components from the classical software packages. It is typically applied to a class element in a class diagram and represents the bridge between the classical and quantum parts of the system.

<<Quantum Request>>

Use Case: applied to Include
Class: applied to Association Class, Class, Dependency and Operation
Sequence: applied to Lifeline and Message

Quantum requests from classical use case to quantum use case; <<Quantum Request>> can be applied under include relationships. It is a call to any component encoded as quantum software, which must run on a quantum device.

The response to the request could be quantum and needs transformation before being received as classical.

Requests from the <<Quantum Driver>> class to the <<Quantum>> class are modelled with the <<Quantum Request>> stereotype. The <<Quantum Request>> is optional and can be used with association classes and dependencies. First, using the <<Quantum Request>> with an Association Class between <<Quantum Driver>> and <<Quantum>> allows the link of cost and optimisation functions to the quantum request. Second, a <<Quantum Request>> dependency represents the same call more simplistically (without further information). When a <<Quantum Driver>> class triggers a quantum operation, the operation can be labelled with the stereotype <<Quantum Request>> to distinguish it from non-quantum operations of the same class.

Since multiple quantum requests can be made to a quantum algorithm in sequence diagrams, a loop fragment can be added to this part.

A <<Quantum Request>> stereotype is also used in a sequence diagram to denote whether messages between quantum and classical software are classical or quantum.

Summary:

This stereotype, used in different UML diagrams, represents a call to any component encoded as quantum software that must run on a quantum device. In use case diagrams, it can be applied to include relationships to denote quantum requests from ordinary use cases to quantum-based use cases. Sequence diagrams are used for messages exchanged between classical and quantum software components.

This stereotype allows for modelling quantum calls or requests from the «Quantum Driver» classes in classical software to the «Quantum» classes. It can be used with association classes, dependencies, and even operation elements to provide different levels of detail about the quantum request.

It is important to note that the request itself may not be inherently quantum. Instead, it indicates that the response to that request could be quantum and thus need some transformation for the classical part.

<<Quantum Reply>>

Sequence: applied to Message

In Sequence Diagrams, the <<Quantum Reply>> stereotype is used to denote that a quantum algorithm answer has been sent via a <<Quantum>> component. The cost function

will translate the quantum answers into a classical one, which is then sent back to the user as a reply.

Summary:

This stereotype denotes that the quantum algorithm's answer has been sent by the «Quantum» component. It is typically applied to a message element in a sequence diagram. It represents the flow of information from the quantum part back to the classical part of the hybrid information system.

<<Quantum Circuit>>

Activity: applied to Activity

Quantum algorithms are represented by a single compound activity with the <<Quantum Circuit>> stereotype. The entire circuit is defined as an activity, and the compound activity can be reused for other circuits, as often occurs in quantum programming

<<Qubit>>

Activity: applied to Activity Partition

The various activity partitions, graphically represented as horizontal swim lanes, can be defined in the parent activity by employing the <<Qubit>> stereotype. The circuit has as many activity partitions as the number of qubits used in the algorithm.

<<Quantum Gate>>

Activity: applied to Action

All the different quantum gates applied in the circuits are represented as action elements and placed in the respective swim lanes according to the qubit to which the gate is used or controlled.

Ordinary quantum gates (H, Y, Z) are represented as call operation actions plus the <<Quantum Gate>> stereotype. Conditional gates are represented by multiple action elements.

<<Measure>>

Activity: applied to Value Specification Action

Special operations, such as qubit measuring and qubit resetting, are represented by value specification action elements and their respective stereotypes <<Measure>> and <<Reset>>.

<<Reset>>

Activity: applied to Value Specification Action

Special operations, such as qubit measuring and qubit resetting, are represented by value specification action elements and their respective stereotypes <<Measure>> and <<Reset>>.

<<Controlled Qubit>>

Activity: applied to Send Signal Action

Control qubits are represented by send signal action elements with the <<Controlled Qubit>> stereotype, while the gate applied is defined by the counterpart element, accept event action, plus the <<Quantum Gate>> stereotype.