Modelling Quantum Software: An Annotated Bibliography

Luis Mariano Bibbo, Alejandro Fernandez, José Manuel Suárez y Oscar Pastor

TLISC - Agosto 2024

Our Hypothesis

- Quantum computing is going to have a big impact on software engineering.
  - New Languages, Environments, New Paradigm

  What will Quantum Software modelling mean?

  Can they be addressed with traditional design tools?

Purpose

- Examine several published works that analyse quantum software modelling
- Generate brief annotations of each work highlighting:
  - What software engineering resources they provide
  - At what stage in the software development process they are applied
- Selection of works:
  - We explored some papers and started to go through the references and related work.

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

Design approach

- Math oriented
  - “A Graph-Based Approach for Modelling Quantum Circuits” D. Alonso, P. Sánchez, and B. Alvarez,

Representation of quantum circuits as graphs. And a metamodel that allows them to transform graphs into other graphs. M2M Transformation

  - “Quantum Software Models: The Density Matrix for Classical and Quantum Software Systems Design,”I. Exman and A. T. Shmilovich,

Design procedure starting point for both classical and quantum software systems is Von Neumann’s quantum notion of Density Operator and its Density Matrix representation.

Design approach

- Workflow oriented, Optimization, Transformation and Mapping
  - B. Weder, U. Breitenbucher, F. Leymann, and K. Wild, “Integrating quantum computing into workflow modeling and execution”

A modelling extension for imperative workflow languages (BPMN) to enable the integration of quantum computations and ease the orchestration of classical applications and quantum circuits.

  - J. Luo and J. Zhao, “Formalization of Quantum Intermediate Representations for Code Safety,”.

Standard LLVM (Low Level Virtual Machine) compiler

  - T. Jin and J. Zhao, “ScaffML: A Quantum Behavioral Interface Specification Language for Scaffold,”

Provides insights into the use of assertions in quantum languages (such as Scaffold).

Design approach

- Conceptual Models
  - L. S. Barbosa, “Software engineering for ’quantum advantage’,”:

    Proposes to deepen the study of three lines of quantum software engineering: Models, Architectures and Properties.

  - R. Juárez-Ramírez, et all, “A Taxonomic View of the Fundamental Concepts of Quantum Computing– A Software Engineering Perspective,”:

    Introduces the main concepts of quantum computing and classifies them. Presents a useful taxonomy to introduce the subject.

  - S. Ali and T. Yue, “Modeling Quantum programs: Challenges, initial results, and research directions,”

    Propose on model-based engineering of quantum programs and assert that they must be platform independent.

Design approach

- Conceptual Models including Model Driven
  - C. A. Pérez-Delgado and H. G. Perez-Gonzalez, “Towards a Quantum Software Modeling Language,”:

    Extend UML class diagram to model quantum characteristics in a class design.

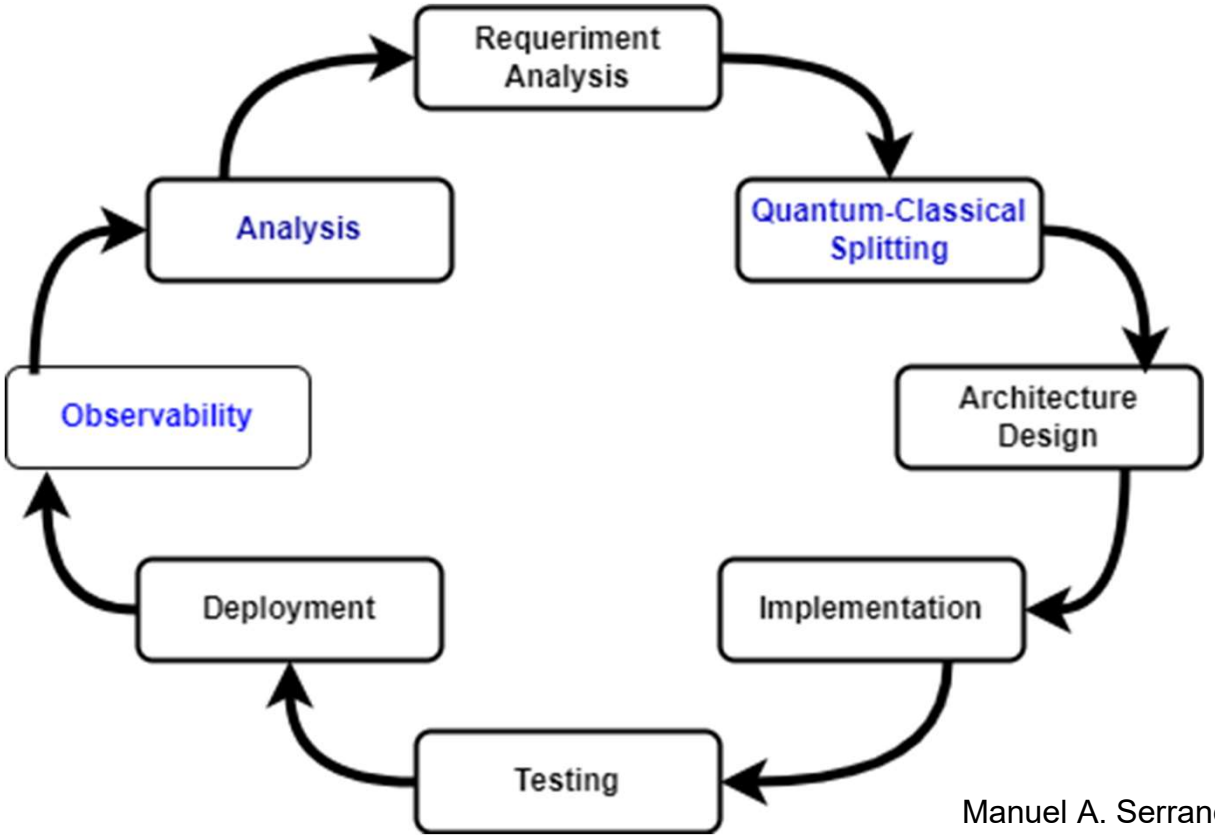
  - R. Pérez-Castillo and M. Piattini, “Design of classical-quantum systems with UML,”

    Propose a Q-UML profile (UML extension) for modelling quantum elements together with classical elements.

  - F. Gemeinhardt, A. Garmendia, and M. Wimmer, “Towards Model-Driven Quantum Software Engineering,” and F. Gemeinhardt, A. Garmendia, M. Wimmer, and R. Wille, “A Model-Driven Framework for Composition-Based Quantum Circuit Design,”F. Gemeinhardt, M. Eisenberg, S. Klikovits, and M. Wimmer, “Model-Driven Optimization for Quantum Program Synthesis with MOMoT,”

    Model-driven methods to produce quantum software. These papers propose modeling languages and transformers that allow the construction of quantum-specific functionalities.

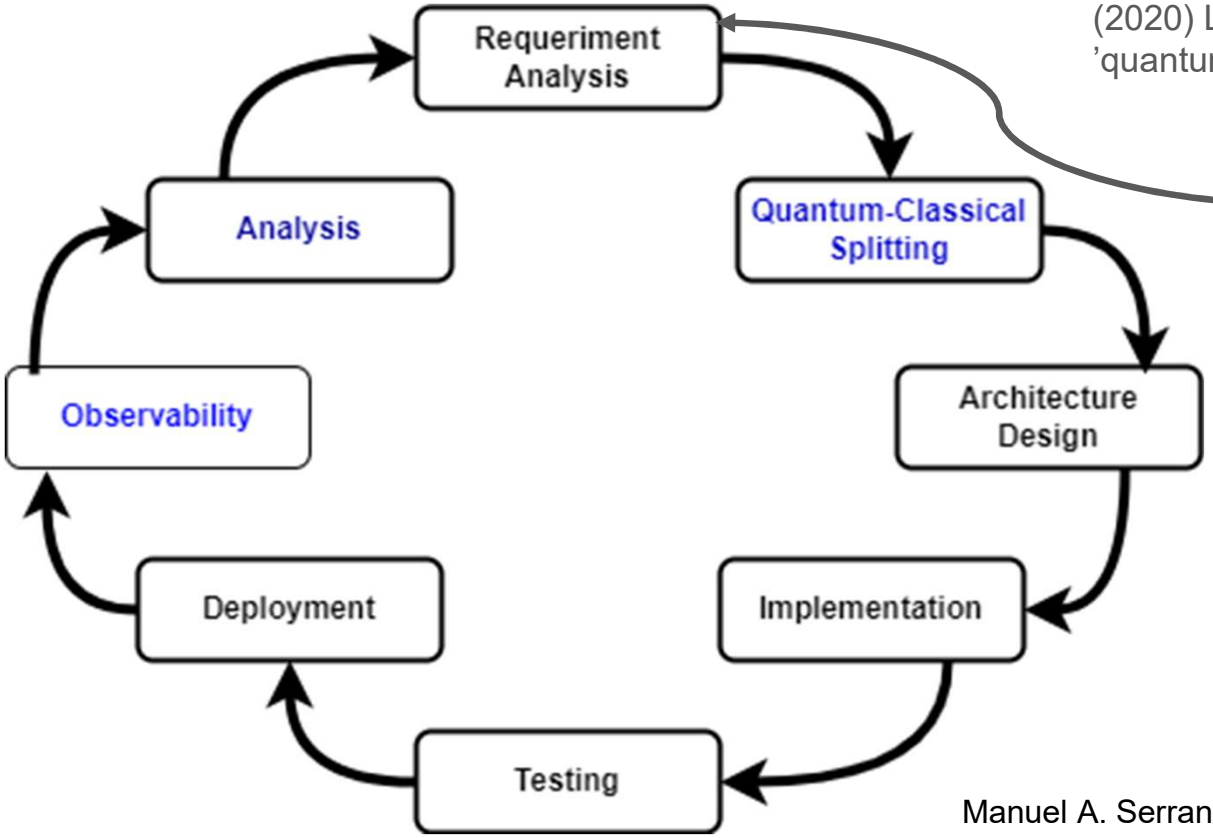
Eje en el de Ciclo de vida



Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

Luis Mariano Bibbo

Eje en el de Ciclo de vida

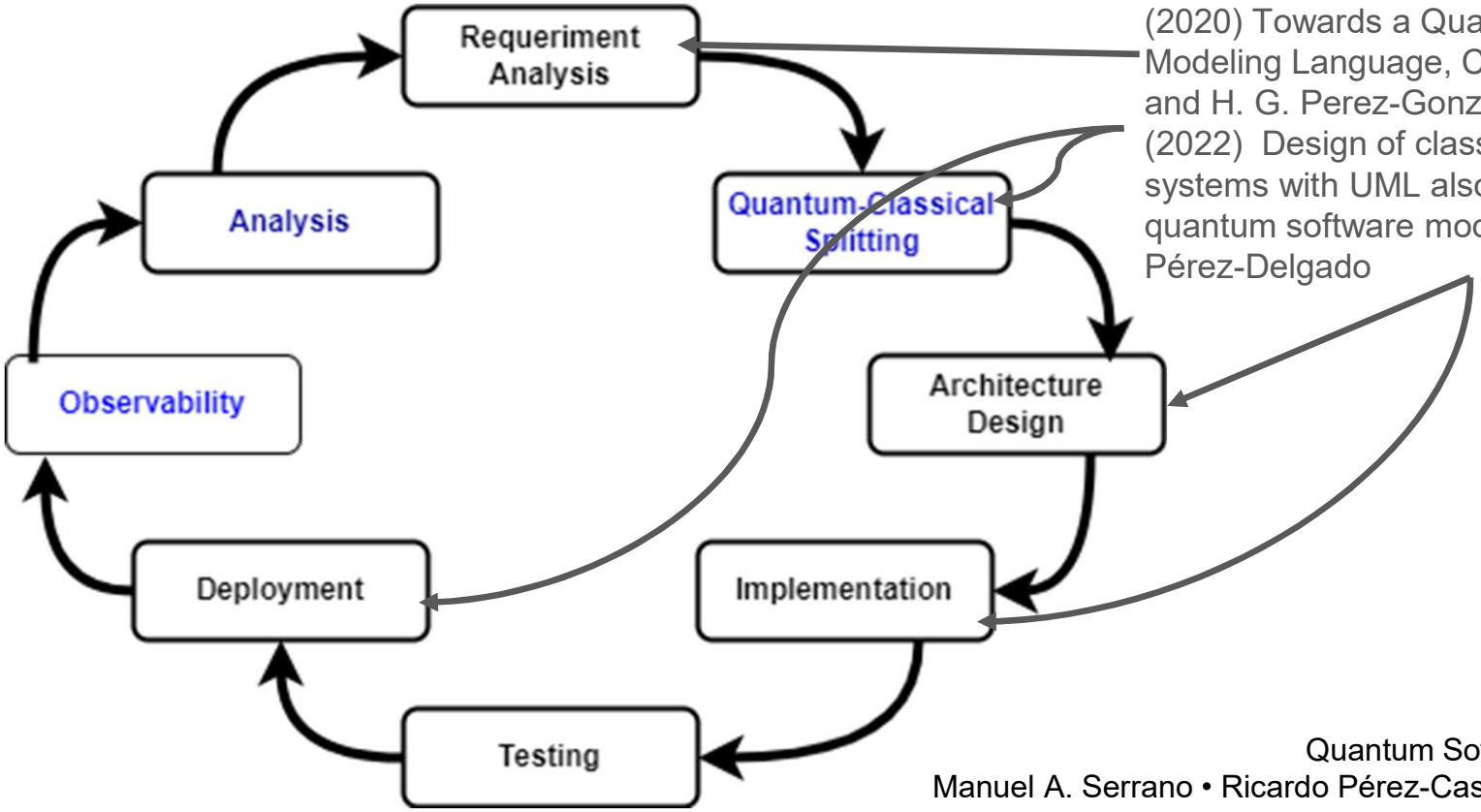


(2019) F. Leymann. Towards a Pattern Language for Quantum Algorithms:
(2020) L. S. Barbosa, “Software engineering for ’quantum advantage’,”

(2020) L. Nallamothula, “Selection of quantum computing architecture using a decision tree approach,”
(2020) S. Ali and T. Yue, “Modeling Quantum programs: Challenges, initial results, and research directions,”
(2023) R. Júarez-Ramírez, et.al“ A Taxonomic View of the Fundamental Concepts of Quantum Computing–A Software Engineering Perspective,”

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

Eje en el de Ciclo de vida

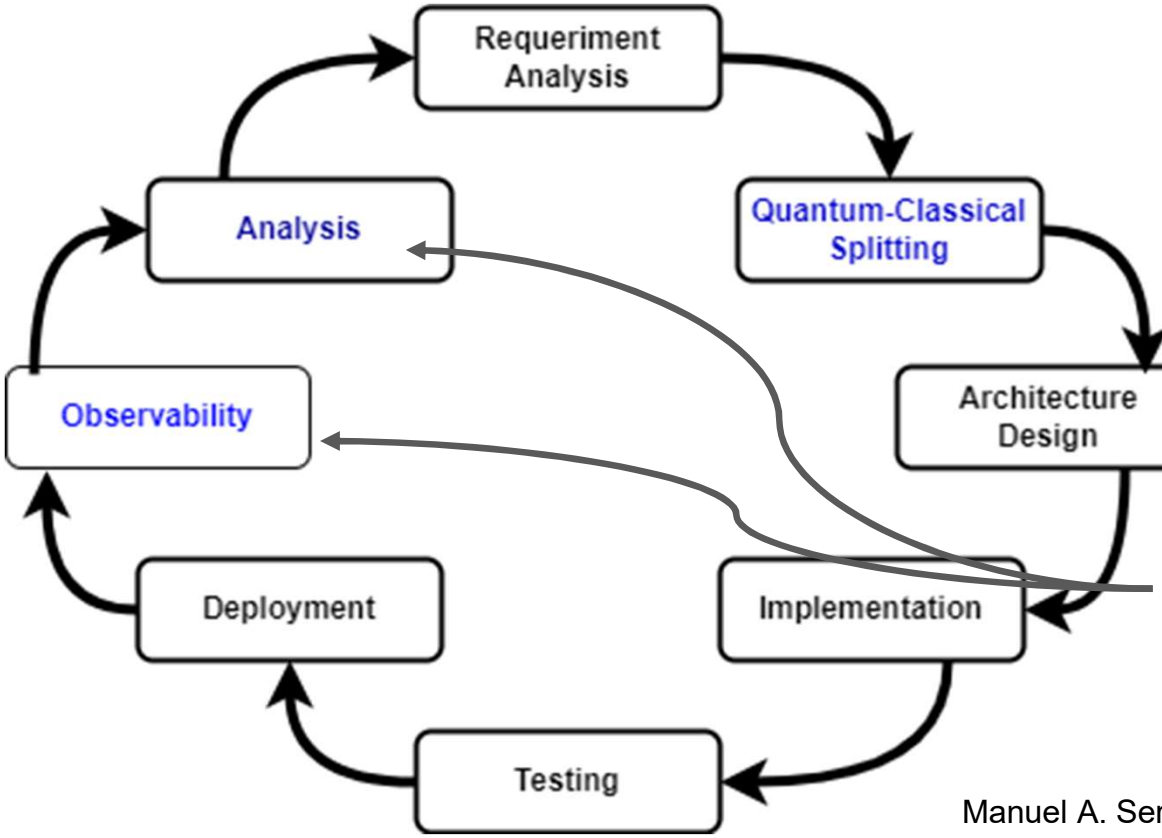


(2020) Towards a Quantum Software Modeling Language, C. A. Pérez-Delgado and H. G. Perez-Gonzalez

(2022) Design of classical-quantum systems with UML also in Chapter 6; A quantum software modeling language. Pérez-Delgado

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

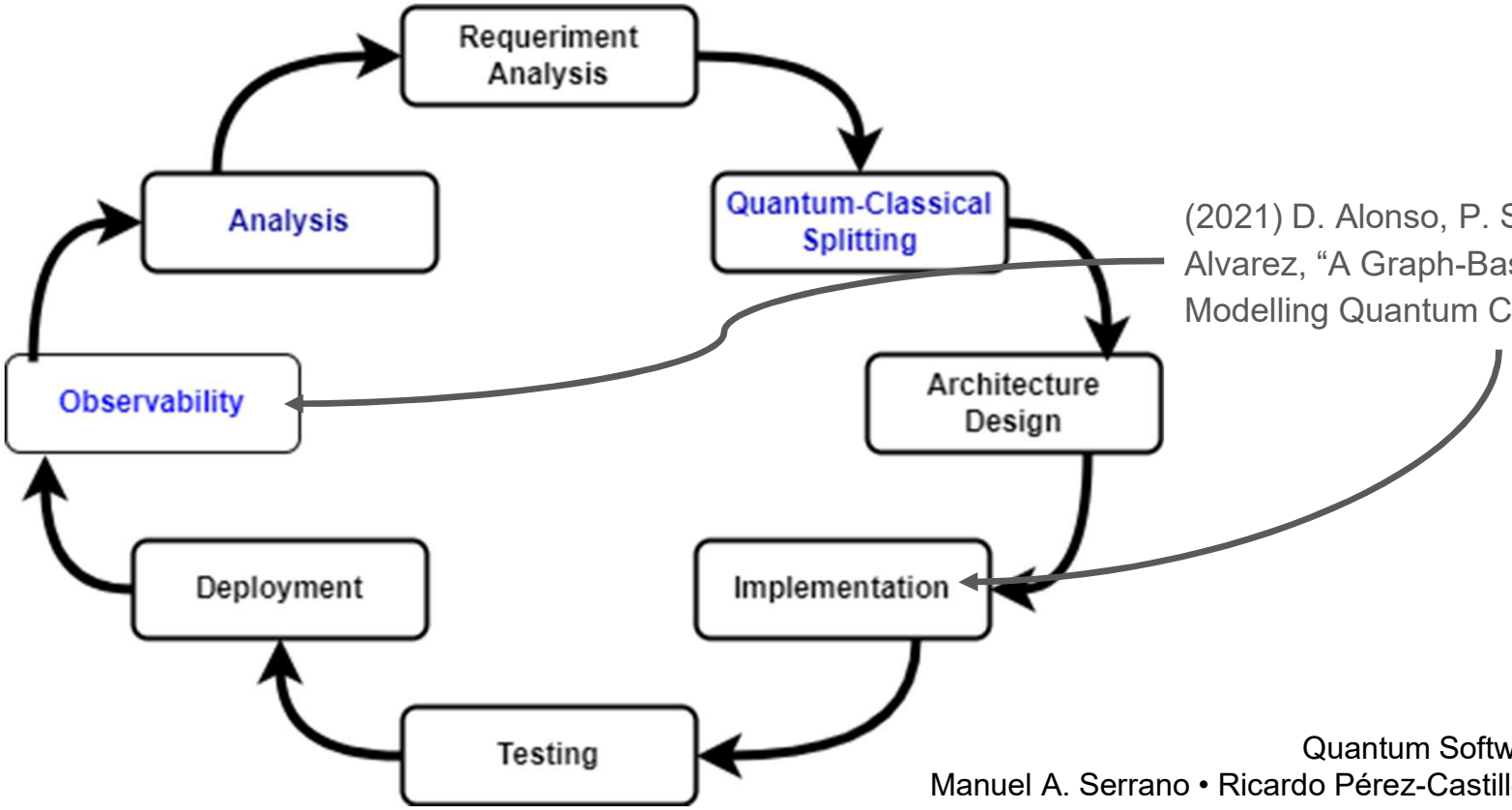
Eje en el de Ciclo de vida



(2020) B. Weder, U. Breitenbucher, F. Leymann, and K. Wild, “Integrating quantum computing into workflow modeling and execution

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

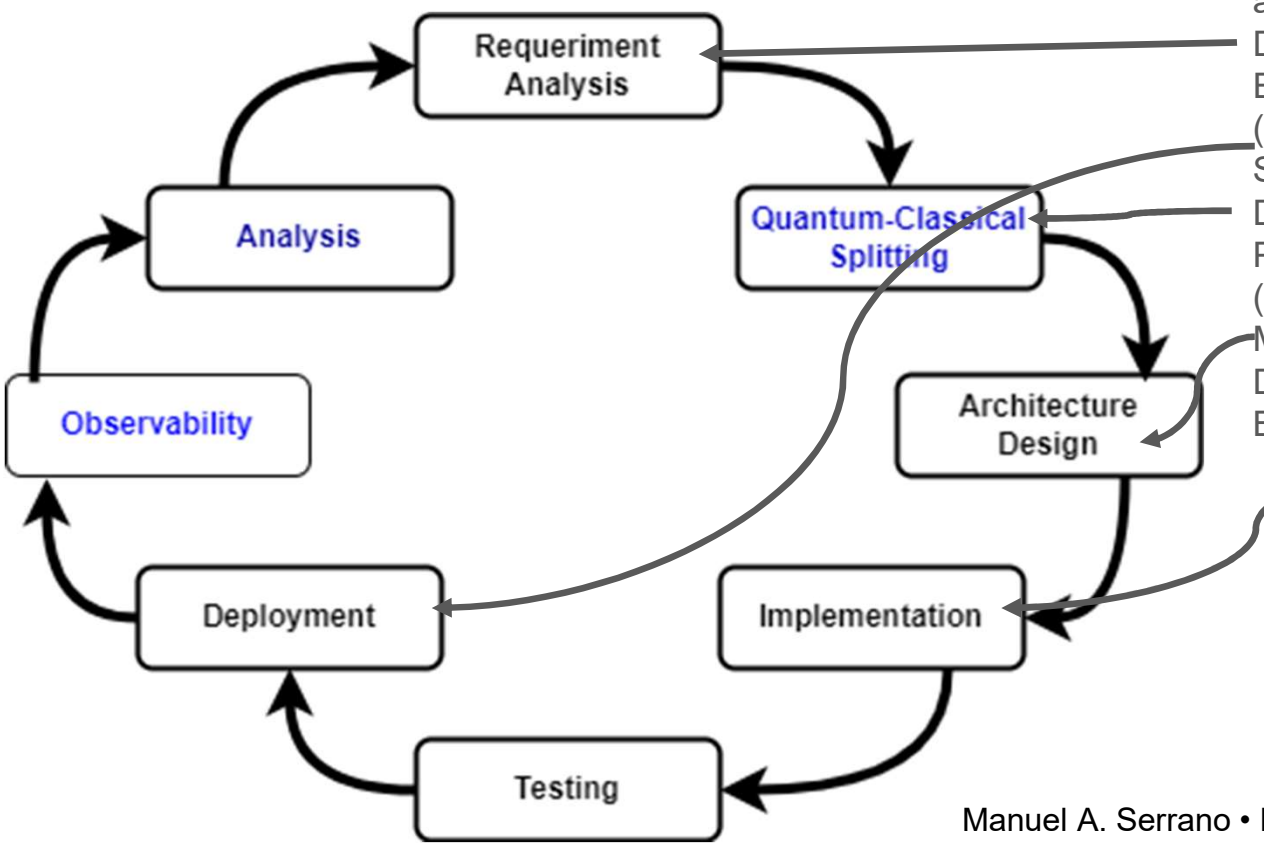
Eje en el de Ciclo de vida



(2021) D. Alonso, P. Sánchez, and B. Alvarez, “A Graph-Based Approach for Modelling Quantum Circuits,”

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

Eje en el de Ciclo de vida



(2021) F. Gemeinhardt, A. Garmendia, and M. Wimmer, “Towards Model-Driven Quantum Software Engineering,”
(2023) F. Gemeinhardt, M. Eisenberg, S. Klikovits, and M. Wimmer, “Model-Driven Optimization for Quantum Program Synthesis with MOMoT,”
(2024) F. Gemeinhardt, A. Garmendia, M. Wimmer, and R. Wille, “A Model-Driven Framework for Composition-Based Quantum Circuit Design,”

Quantum Software Engineering
Manuel A. Serrano • Ricardo Pérez-Castillo • Mario Piattini
Chapter 4: **Quantum Software Development Lifecycle**
Benjamin Weder, Johanna Barzen, Frank Leymann, and Daniel Vietz

Conclusion and Future Work

For each work we made a brief commentary and tried to answer - what it tries to model (Design Effort) and - in which stage of the lifecycle of the development of a quantum system.

- Most of the papers are Conceptual Models; the works of Perez Delgado (UML Profile) and Gemeinhardt, Felix et all. that aim to work with MDE techniques where they provide model-to-code transformations stand out
- Most of the work is focused on the stages of Requirements Analysis Quantum-Classical Splitting Design and Implementation. There is a deficit in the stages of Analysis of results and Observability

For the future

- Firstly, conducting a Systematic Literature Review (SLR) to formalize this research
- Secondly, identifying and analyzing existing gaps within the discipline and proposing innovative solutions to find methodologies and tools to improve existing practices in this area.

- Thanks

Luis Mariano Bibbo