TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

sedaotoioo

software engineering dependability

Validation of Self-Adaptive Systems' Safety Requirements at Design Time

Rasha Abu Qasem
Chair of Software Engineering: Dependability
University of Kaiserslautern, Germany
abuqasem@cs.uni-kl.de

Peter Liggesmeyer
Chair of Software Engineering: Dependability
University of Kaiserslautern, Germany

Rasha Abu Qasem

- Bachelor in Information Technology from Damascus University, Syria
- Masters of Science in Software Engineering from University of Kaiserslautern, Germany
- Researcher and PhD candidate at the Chair of Software Engineering: Dependability at the University of Kaiserslautern, Germany

Specification of Safety Requirements of Self-Adaptive Systems (SAS)

- There are several attempts to target uncertainty in requirements of SAS
- Safety requirements of SAS are not properly addressed
- Lack of guidance on how to specify safety requirements that are properly traceable to the architecture design and to failure propagation models
- No clear methods neither recommendations –to our best knowledge- explain how to elicit or manage safety requirements of SAS

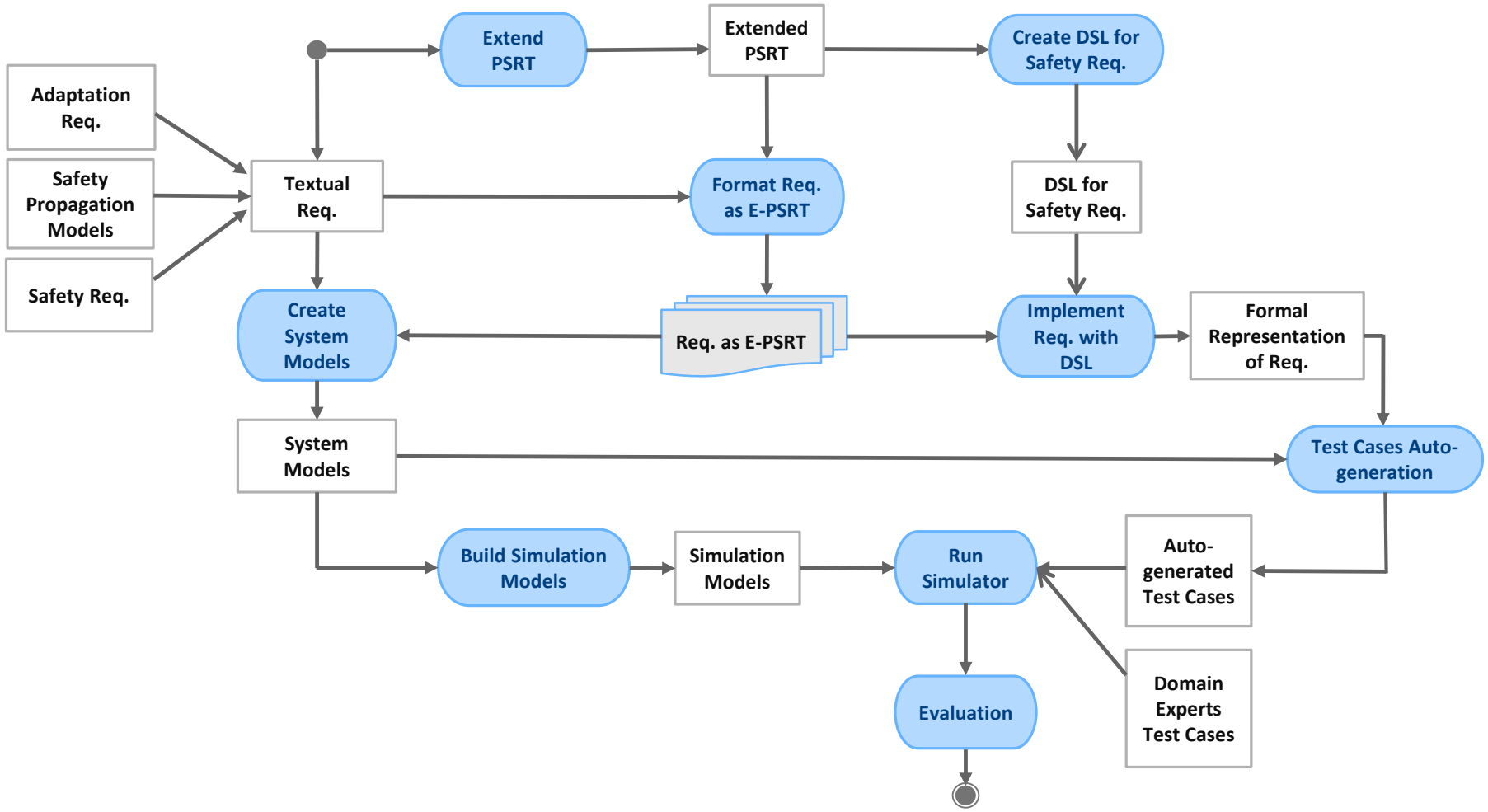
Validation of Safety Requirements of SAS at Design time

- The earlier the error is detected the less impact it has on the development's cost and effort
- Trace errors and locate potential design flows before the actual implementation takes place
- Provide the system designer and system analysts with a systematic method to validate the system design and architecture at design time

Auto-generation of Test Cases to Verify the Safety of the SAS

- Documented test cases are essential for testing large and complex systems such as SAS
- Usually test cases are derived manually from the textual requirements
- Manual test case generation is a time consuming and error-prone process
- Test case generation proved to be a powerful approach to reduce the cost of testing as well as to assure the requirements' coverage
- Non of the test cases' auto-generation approaches –to our best knowledge- has addressed the safety requirements of SAS

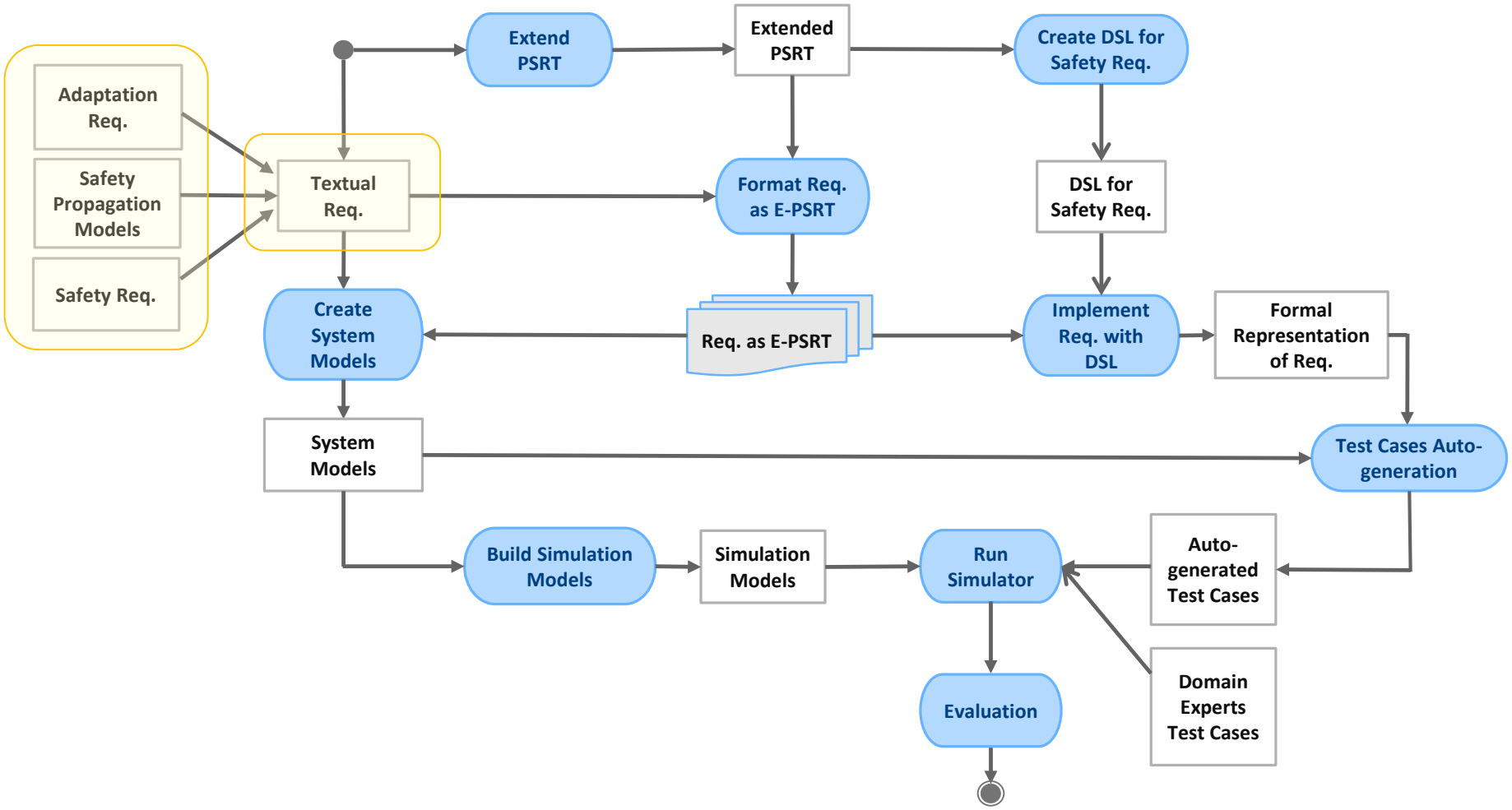
Validation of Self-Adaptive Systems' Safety Requirements at Design Time



This approach:

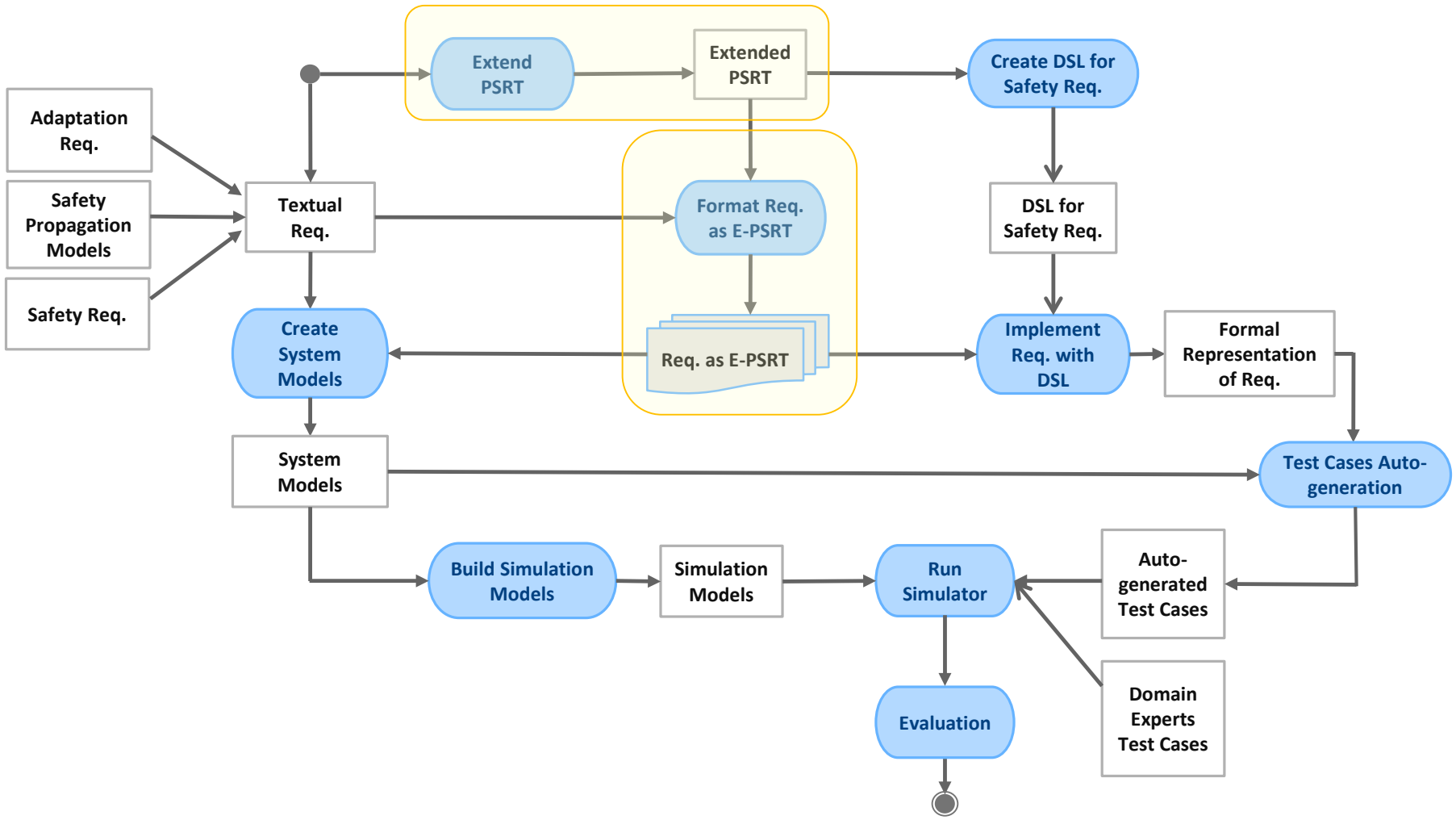
- Is a means to evaluate the safety requirements of a self-adaptive system at design time
- Assesses the system behavior in critical events
- Gives a clear guidance about the quality of the architecture design
- Checks the adherence of the architecture design to the overall safety requirements of SAS

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



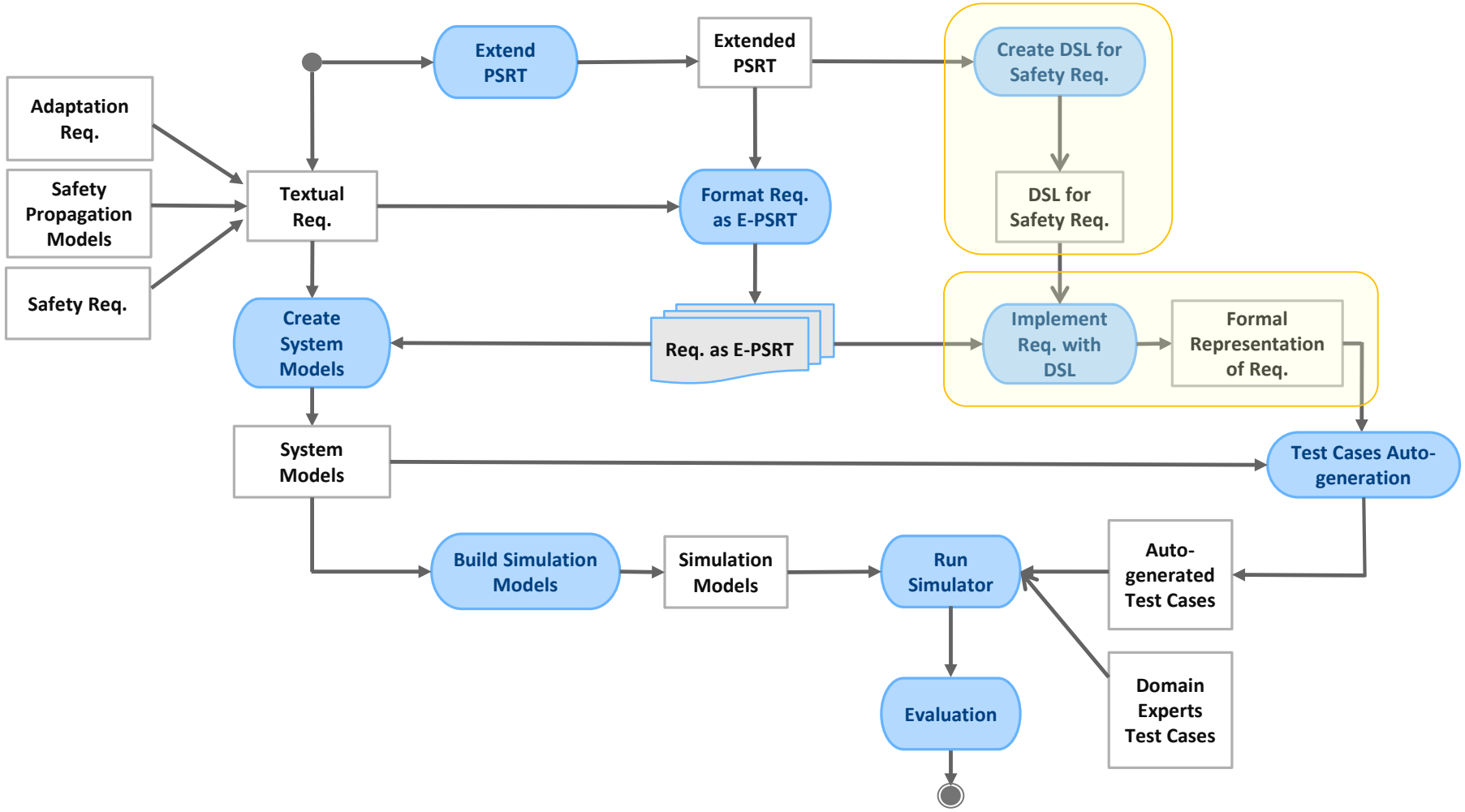
- Starting point is the textual requirements
- Input of the approach:
  - The adaptation requirements
  - Safety requirements
  - Safety propagation models

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



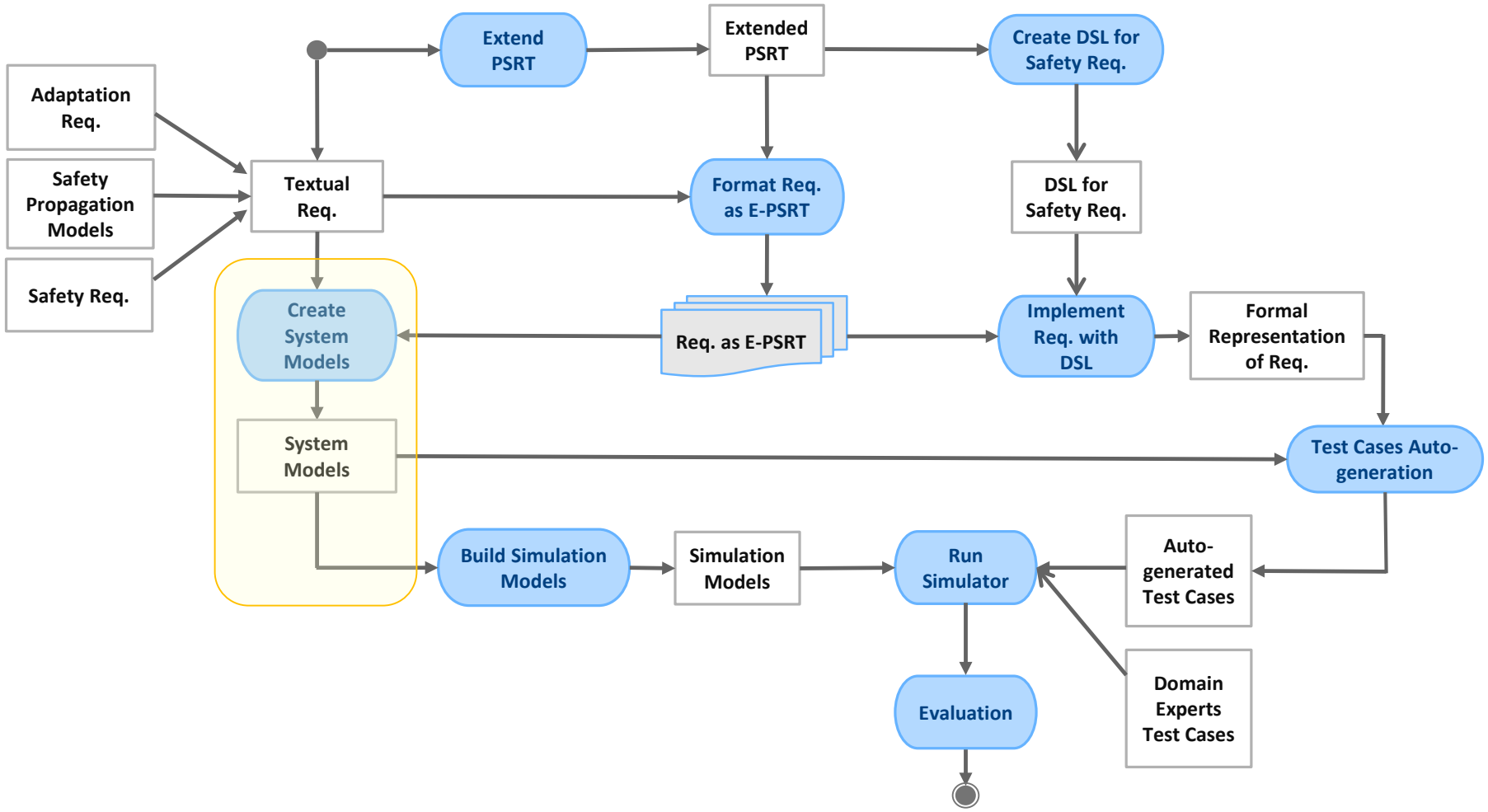
- Parametrized Safety Requirements Templates(PSRT) provide guidelines on how to specify the safety requirements of a system
- PSRT assure consistency and traceability of the safety requirements
- Extend the PSRTs to (E-PSRTs) helps to:
  - specify the safety requirements and the adaptation scenarios of SAS
  - identify inconsistent safety requirements

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



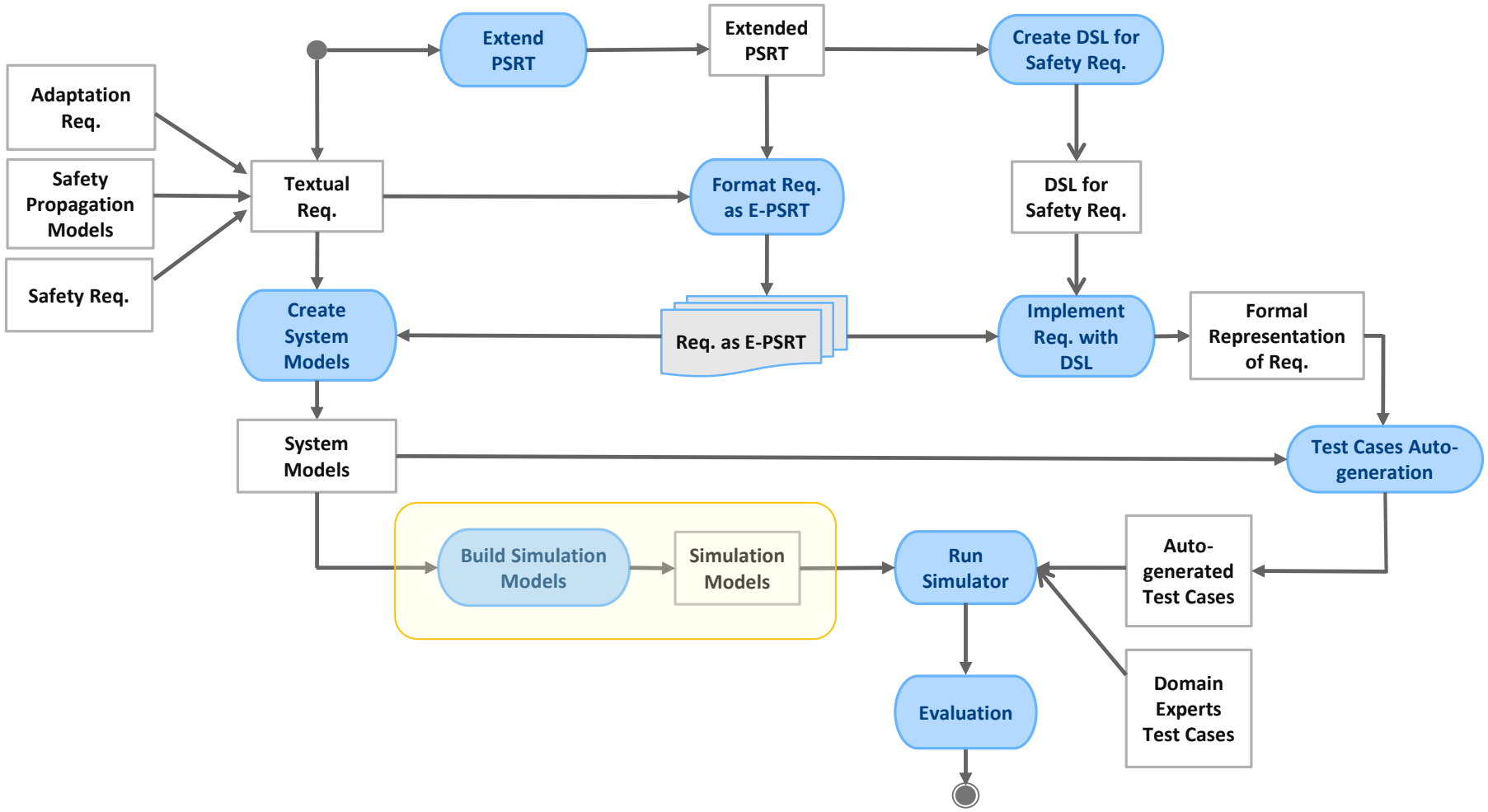
- Design a language to represent the adaptation scenarios and safety requirements of SAS
- Parse the safety requirements to a formal structure

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



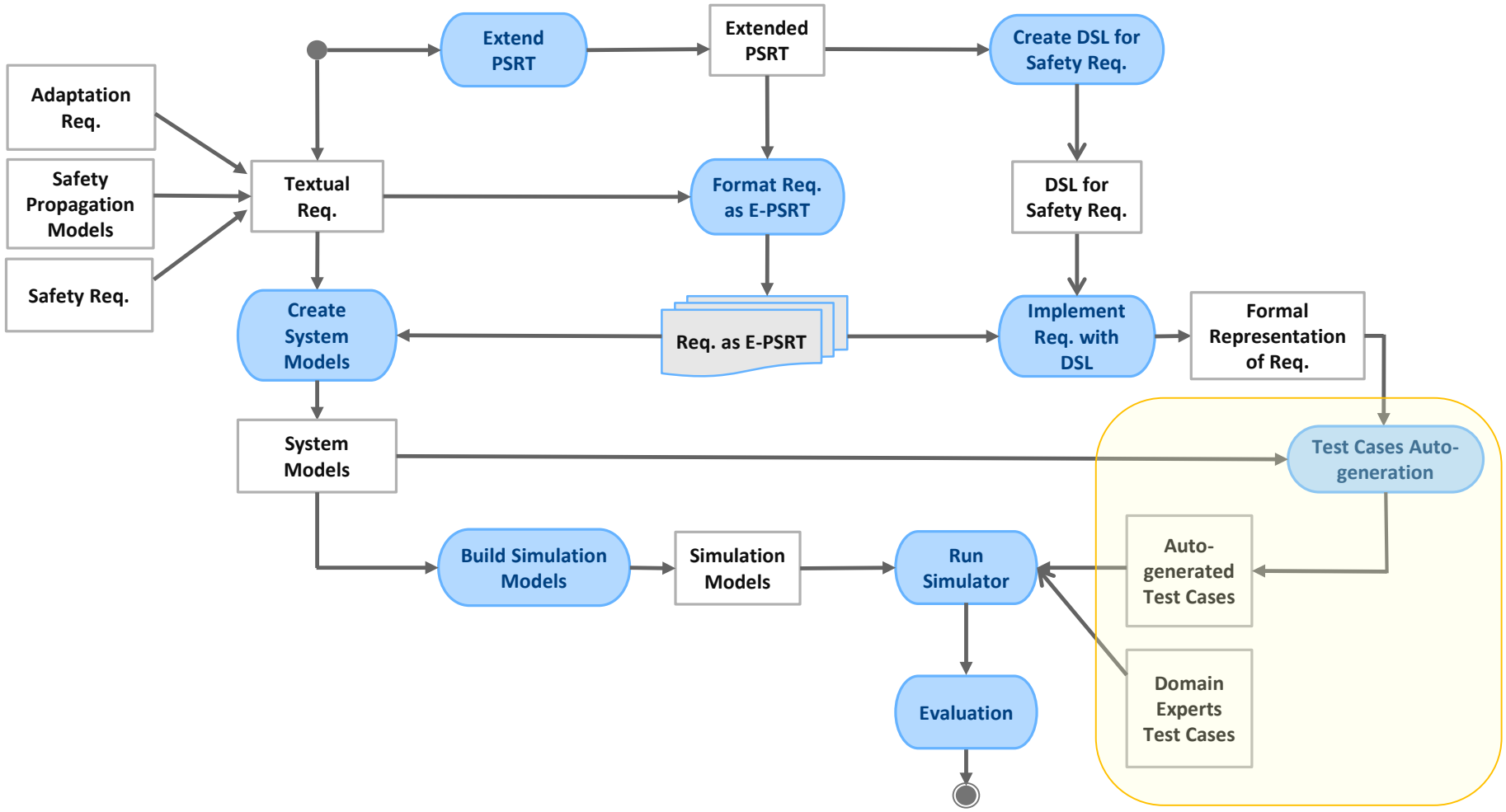
- Create the system models and architecture
- This step can be conducted manually or in a semimanual fashion

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



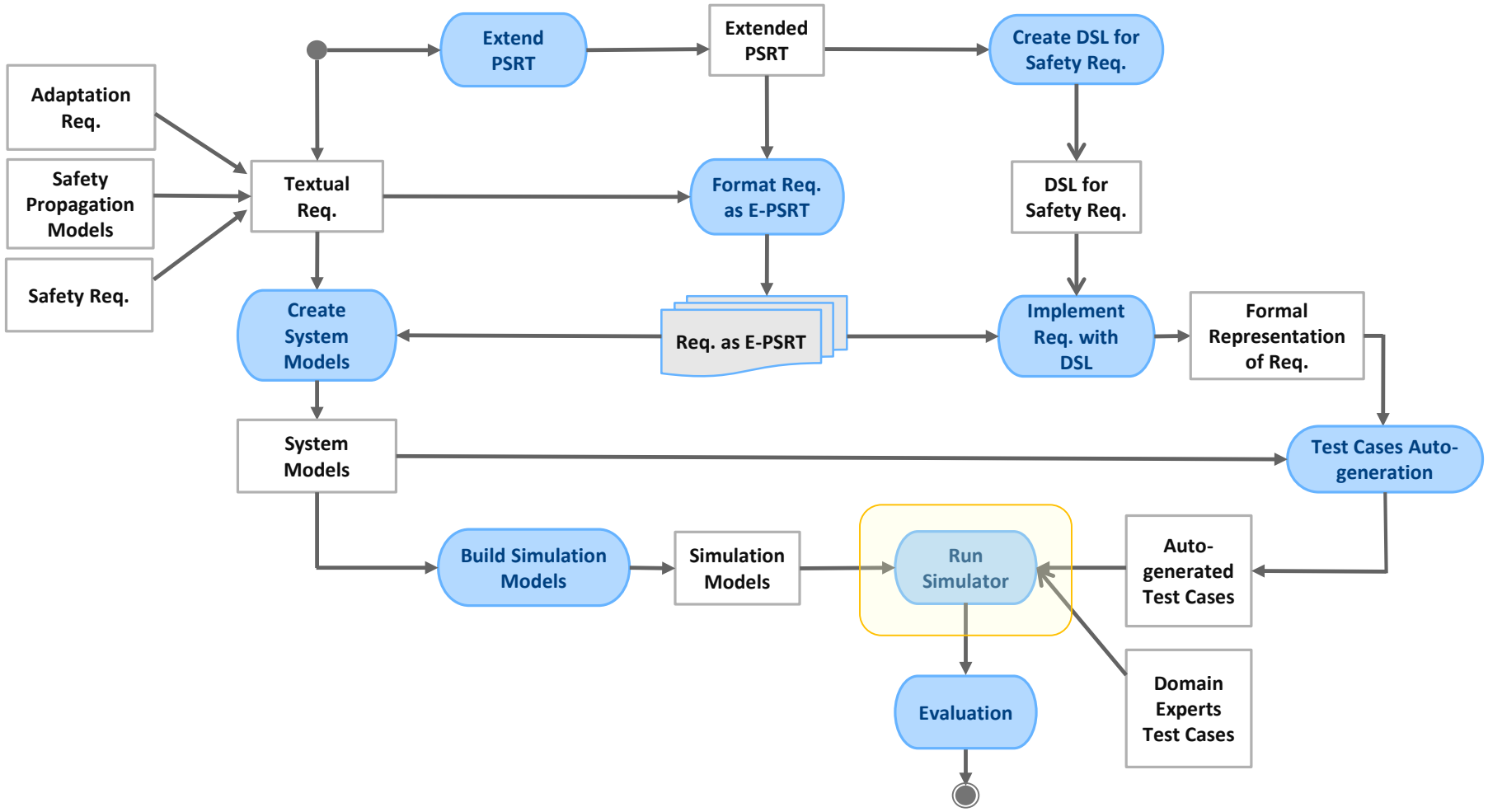
- Build the simulation models based on the system models
- This step is conducted manually or in a semimanual fashion

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



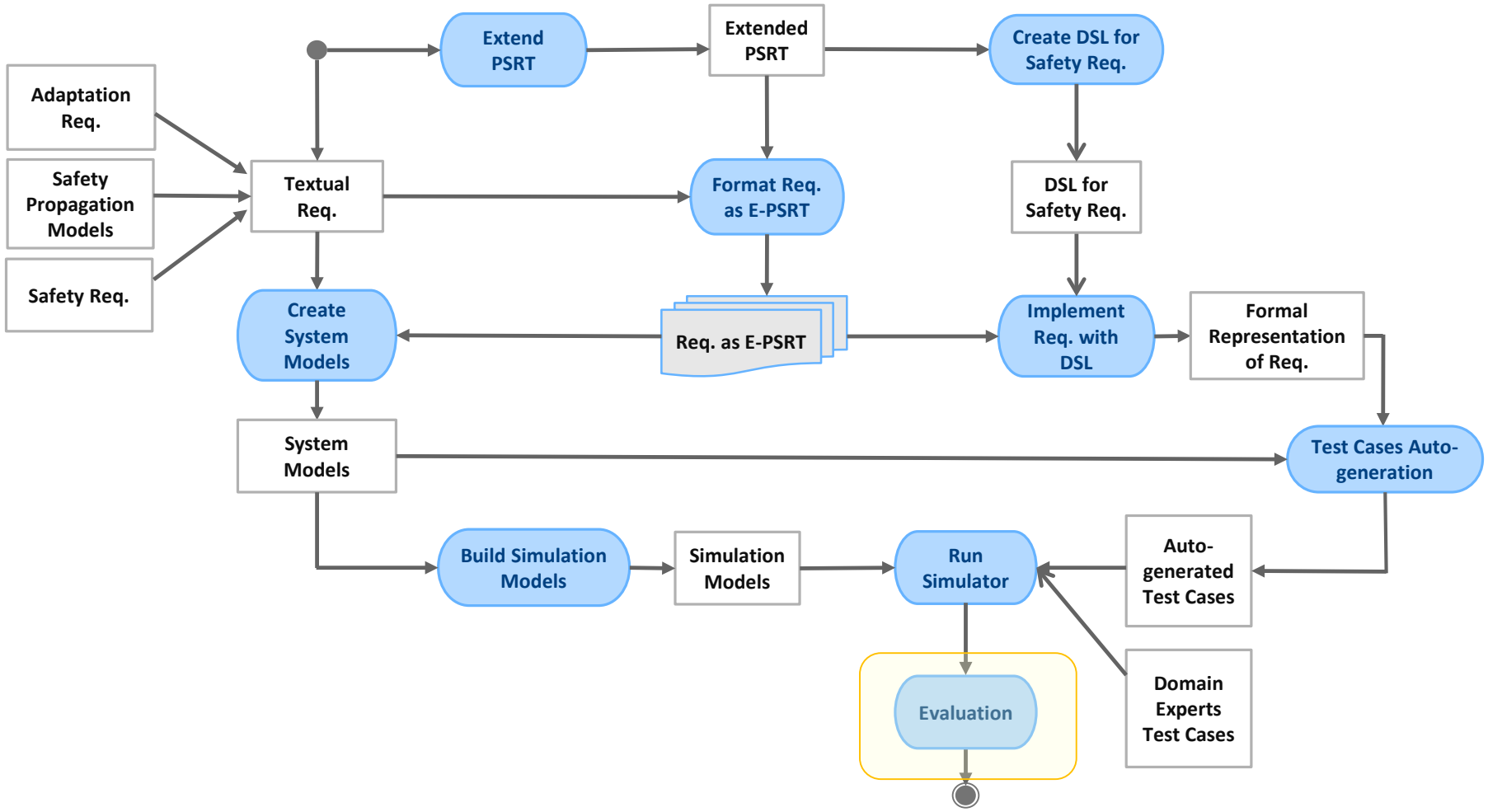
- Deriving test cases in a systematic way
- Assure coverage of all systems' safety requirement
- Inject failures

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



- Run the simulation models in a simulator
- Feed the simulator with:
  - auto-generated test cases
  - domain experts' test cases
  - simulation models

Validation of Self-Adaptive Systems' Safety Requirements at Design Time



- Monitor the simulator behavior
- Compare the expected behavior with the resulted/simulated behavior
- Raise alarm in case of deviations or hazardous behaviors
- Trace the behaviors back to the initial requirements
- Perform the needed updates

Conclusion

- Building adaptable systems in safety-critical environments is a challenging task
- Our proposed approach addresses some of these challenges in requirements elicitation and system design phases
- We first tackle the problem of specifying safety requirements of SAS and how we integrate them in the adaptation strategies
- We generate adequate test cases to test the expected behavior of the system which enable us to get an early feedback before system implementation
- We use a simulator to run the generated test cases at design time to identify the flaws in safety requirements