



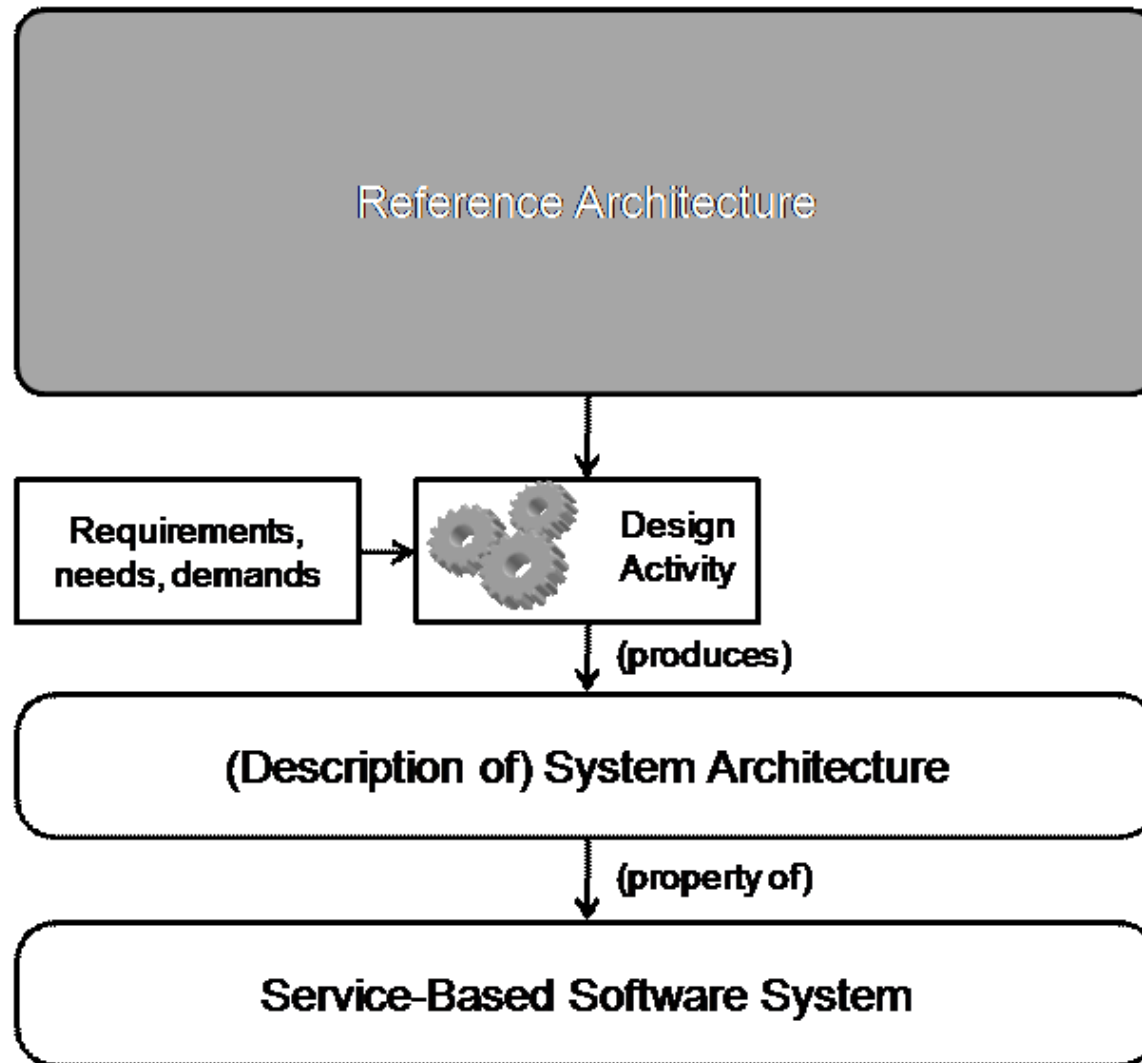
# NEXOF-RA

Ricardo Jimenez-Peris  
NEXOF-RA Research Director  
Universidad Politecnica de Madrid

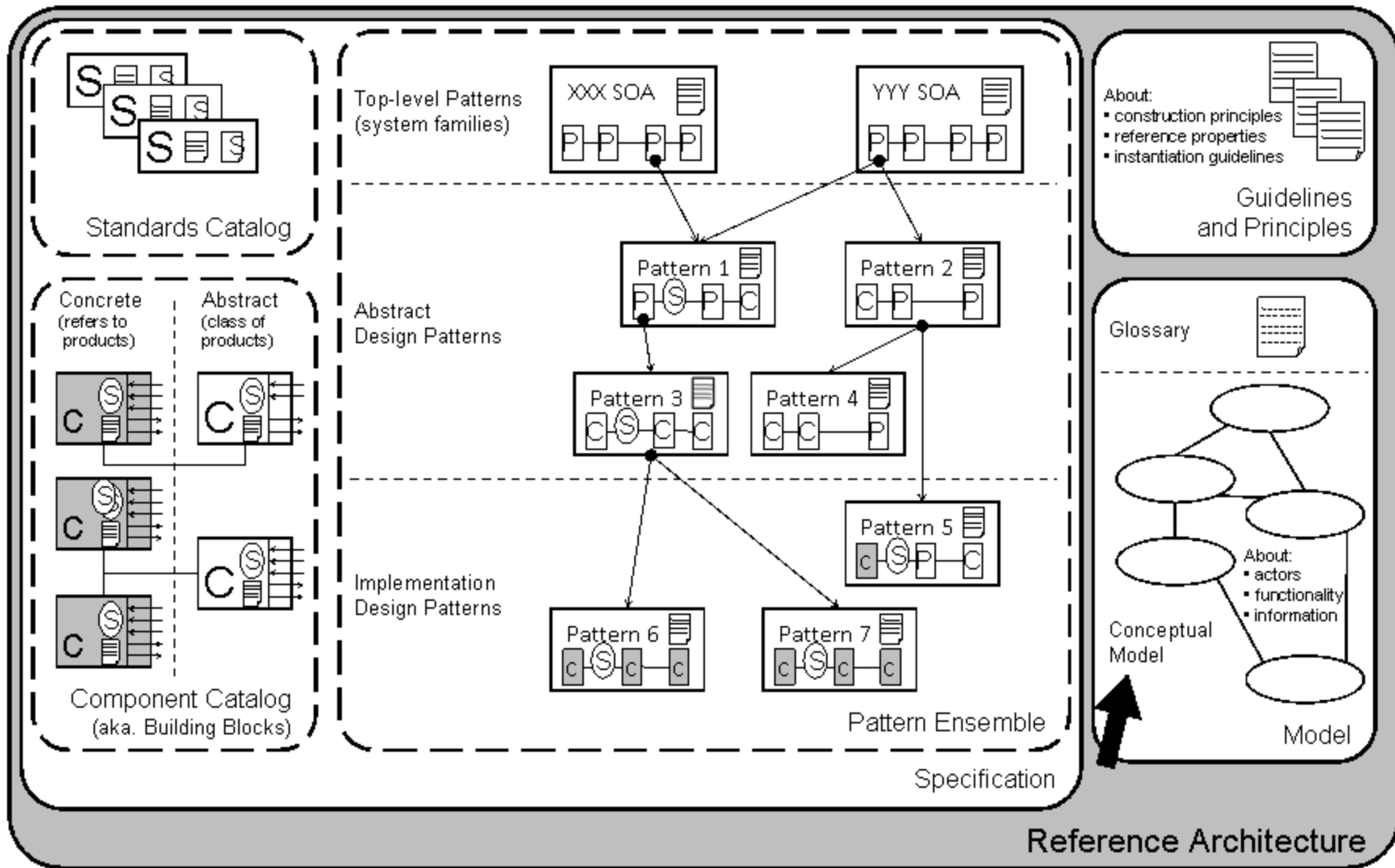
Internet of Services –  
Future Internet Assembly  
Brussels, 10<sup>th</sup> June 2009



- NESSI defined NEXOF with the aim of providing a framework for building service-based software systems.
- NEXOF-RA aims to build the reference architecture for NEXOF:
  - Regardless the size of the business;
  - Domain independent;
  - Technology independent.
- Open nature:
  - Open Construction Process.



- Generality of the architecture to address different service domains and neutrality with respect technology, domain and business size were extremely challenging.
- They have been achieved by resorting to architectural patterns:
  - Functional patterns: Provide patterns on how to interconnect different functionalities to achieve more complex functionalities.
  - Non-functional patterns: Patterns that address cross-cutting non-functional aspects and can be applied to different architecture fragments in different contexts.



- NEXOF-RA is focusing on the scalability and availability of SOI.
- It addresses from Infrastructure as a Service to more traditional SOI as multi-tier architectures.
- The described architectural patterns target medium scalability in clusters with 10s of nodes.
- Moving to the Internet of Services and cloud computing raises a number of open challenges.

- Many efforts in cloud computing for the IoT are focusing on new paradigms.
- Scalability is achieved by overconstraining the infrastructure to enable linear scalability.
- Most cloud infrastructure offer low-medium level abstractions for application development.
- Consistency and atomicity issues are pushed to the application developer.
- Unfortunately, most application developers are unable to solve these hardcore problems.

- The challenges identified by NEXOF-RA are:
  - Develop higher level abstractions for cloud computing.
  - Cloud computing infrastructure providing higher guarantees regarding atomicity and consistency.
  - Better abstractions for eventual consistency.
  - Cloud computing providing similar abstractions as multi-tier SOI.