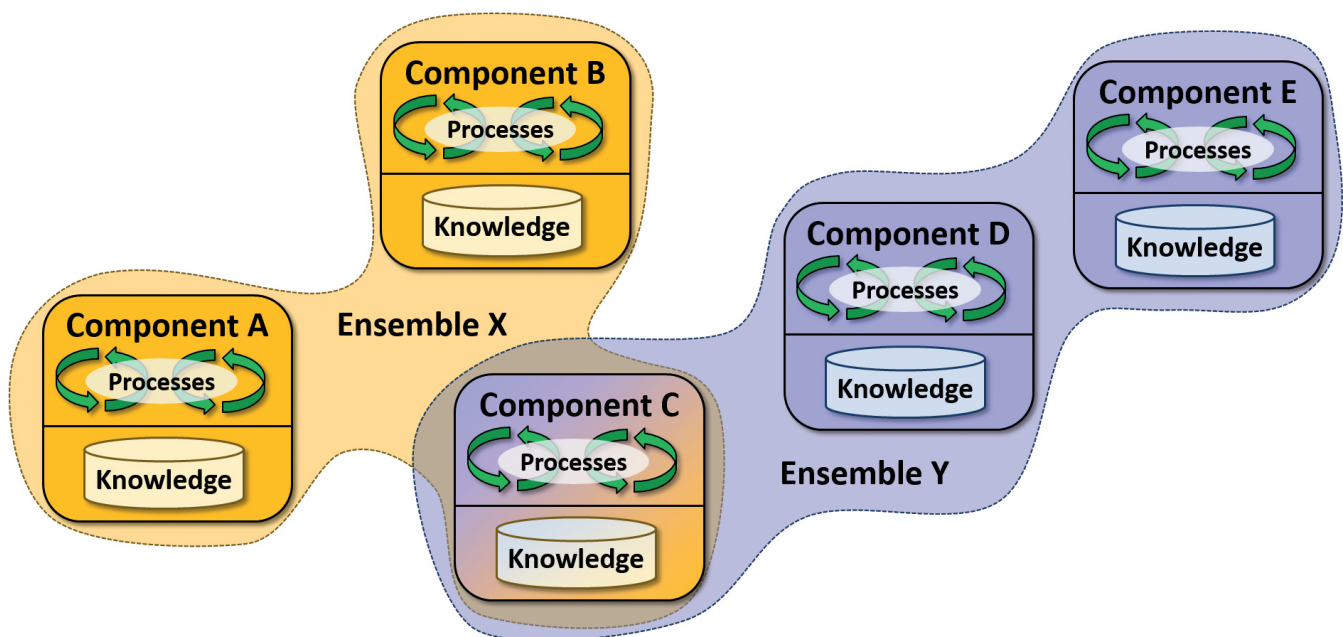


DEECo is a component framework where both component and ensemble are first class concepts. Each component constitutes state, referred to as knowledge, and behavior, expressed in terms of a set of processes. To achieve component interaction, components are dynamically composed into ensembles.

Membership of a component in an ensemble is declaratively expressed in terms of the membership condition, defined on component interfaces, which provide a partial view of component knowledge. Members of an ensemble interact in terms of implicit knowledge exchange, which is handled by the execution environment.



In DEECo, a component operates autonomously, based solely on its own knowledge, which is implicitly updated depending on the ensembles the component belongs to at a particular moment. A single component can be involved in several ensembles at the same time.

In order to bring the DEECo concepts closer to regular software development for evaluation and experimentation, its execution environment (jDEECo) has been created in Java. The DEECo concepts (components and ensembles) are mapped to Java via an internal domain-specific language realized by Java annotations. The core responsibility of the runtime environment – execution of knowledge exchange – is implemented in two ways: via distributed tuple-space middleware or via periodic broadcast, supporting deployment on mobile ad-hoc networks (MANETs).

Further Information:

- [User Guide](#)
- [Download](#)