

Verteilte Prozesse und Aktoren: Ein Paradigma für Alles?

Der erste Teil dieser Präsentation beschäftigt sich mit dem theoretischen Aktorenmodell. Zunächst werden vier Prozesskommunikations-Paradigma vorgestellt, um das Aktorenmodell einordnen zu können. Dann folgt eine Charakterisierung von Aktoren, wie die Verarbeitung, das Speichern und die Kommunikation, als fundamentale Einheit der Arbeit von Berechnungen. Es wird auch aufgezählt was Aktoren in erste Linie können müssen, um Ihrem Namen gerecht zu werden. Es wird erläutert was im Zusammenhang mit dem Aktor Modell das Asynchrone Message Passing ausmacht und was dessen Vorteile sind, sowie das jeder Aktor sein eigenen Zustand im Laufe der Zeit einnimmt. Es soll hierbei auch deutlich gemacht werden, dass es in diesem Modell keinen globalen Systemzustand gibt. Zum Ende der kurzen Abhandlung wird noch ein kurzer Abriss zur Geschichte zu diesem theoretischen Modell präsentiert. Ein kurzer geschichtlicher Überblick über die Entwicklungsgeschichte von Erlang wird ebenfalls mitgeliefert.

Der zweite Teil dieser Präsentation stellt die funktionale Programmiersprache Erlang mit dem Framework OTP (Erlangs de facto Standardbibliothek) vor. Es werden zwei wesentliche Bereiche von Erlang/OTP vorgestellt und erklärt. Es werden die Prinzipien der Supervision und das Schichten von Prozessen beschrieben, zwei Bereiche die unter Verwendung des OTP-Framework auf einfache Art und Weise zu implementieren sind. Es wird deutlich, dass mit Hilfe dieses Frameworks sehr schnell und einfach produktive Software erstellt werden kann.

Zum Ende dieser Präsentation werden noch die beiden Frameworks Akka und CAF (C++ Actor Framework) vorgestellt, sowie ein kleiner Abriss über Pony mitgeliefert.

Distributed Processes and Actors: One Paradigm to Rule them All?

The first part of this presentation describes the theoretical actor model. At first there is an introduction into four process communication paradigms, so the actor model can be better classified. Then follows an explanation about what makes an actor an actor, like the three fundamental things that an actor has to embody: processing, storage and communication. And what an actor can do, like creating more actors or sending messages to other actors. It also will be described the advantages of Asynchronous Message Passing as the important part of this model and further more that there is no global state, each actor has his own state due to time. In the end of this part there is listed an overview about the history of the theoretical actor model and the practical conclusion with Erlang.

The second part of this presentation includes an introduction into the functional programming language Erlang with its framework OTP (the so called „Standard Library“ of Erlang). It will be presented two essential fields of Erlang/OTP. The principals of Supervision and Layering Processes beeing described here, which could be easily implemented with using the OTP-Framework. In conclusion it becomes clear, that using this framework helps building productive software fast.

In the end of this presentation the two frameworks Akka and CAF (C++ Actor Framework) being introduced and a short outline about Pony.