

# Einführung in NaturalX

Dieses Kapitel enthält eine kurze Einführung in die komponentenbasierte Programmierung, und damit einhergehend die Benutzung der NaturalX-Schnittstelle und einem speziell für diesen Zweck vorgesehenen Satz Natural-Statements.

Folgende Themen werden behandelt:

- Warum NaturalX?
- 

## Warum NaturalX?

Auf Komponenten-Architektur basierende Software-Anwendungen bieten viele Vorteile gegenüber traditionellen Designs. Diese sind u.a. die Folgenden:

- Schnellere Entwicklung. Die Programmierer können Anwendungen schneller erstellen, indem sie die Software aus vorerstellten Komponenten zusammensetzen.
- Geringere Entwicklungskosten. Eine allgemeine Menge von Schnittstellen für Programme zur Verfügung zu haben, bedeutet weniger Arbeit bei der Integration der Komponenten in vollständige Lösungen.
- Höhere Flexibilität. Es ist leichter, Software für unterschiedliche Abteilungen innerhalb eines Unternehmens zu standardisieren, indem Sie einfach einige der Komponenten verändern, aus denen die Anwendung besteht.
- Reduzierte Wartungskosten. Im Falle eines Umstiegs auf eine neue Version ist es häufig ausreichend, einige der Komponenten zu ändern, anstatt die gesamte Anwendung ändern zu müssen.

Mit NaturalX können Sie komponentenbasierte Anwendungen erstellen.

Sie können NaturalX einsetzen, um einen komponentenbasierten Programmierstil zu pflegen. Allerdings können auf Großrechner- und UNIX-Plattform die Komponenten nicht verteilt werden und nur in einer lokalen Natural-Session laufen.