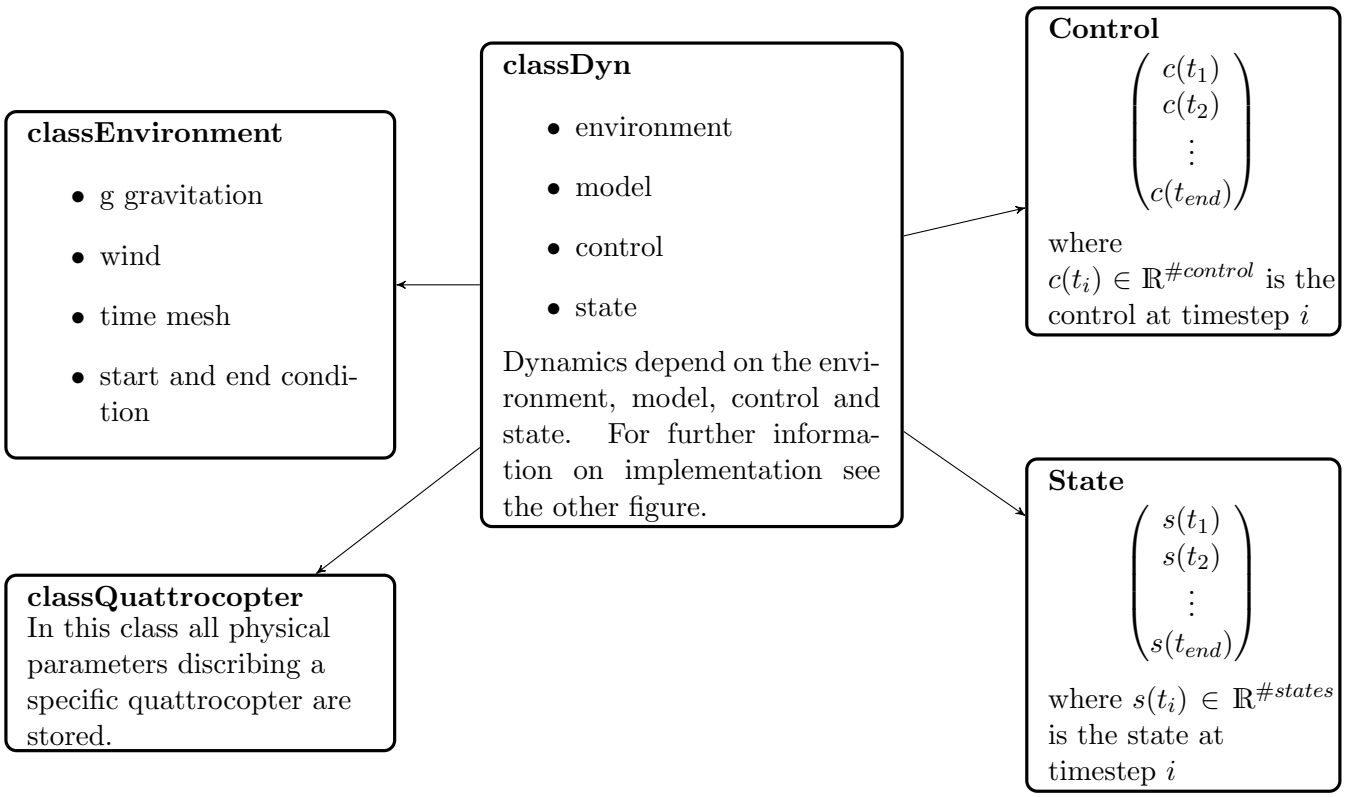
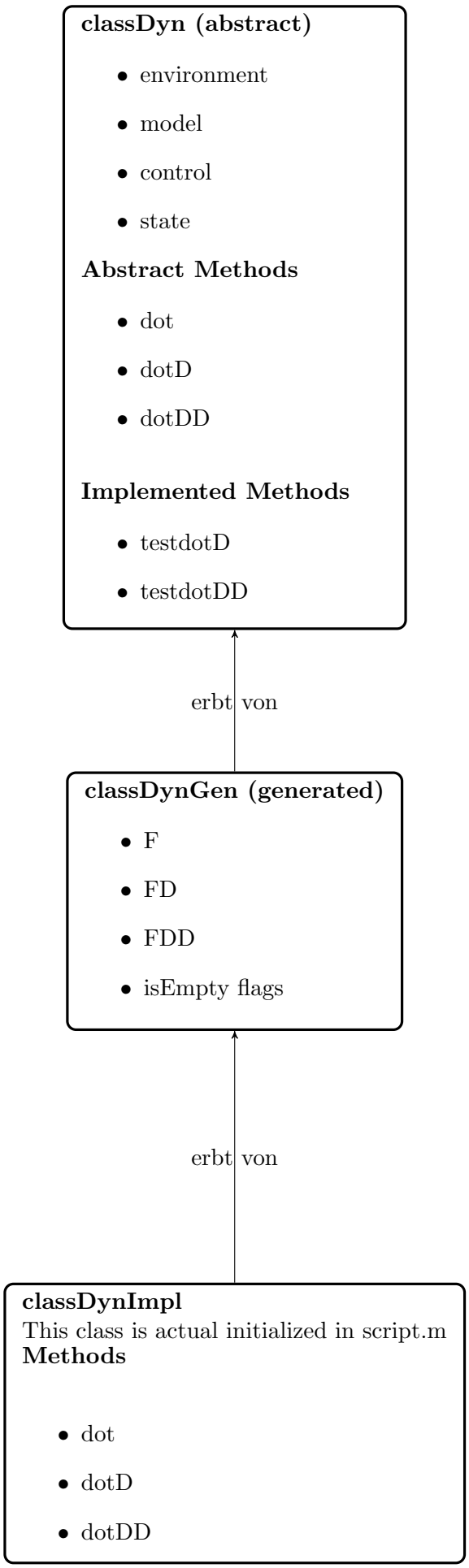

classEnvironment
• g gravitation
• wind
• time mesh
• start and end condition
classDyn
• environment
• model
• control
• state
Dynamics depend on the environment, model, control and state. For further information on implementation see the other figure.
Control
$\begin{pmatrix} c(t_1) \\ c(t_2) \\ \vdots \\ c(t_{end}) \end{pmatrix}$
where $c(t_i) \in \mathbb{R}^{\#control}$ is the control at timestep $i$
State
$\begin{pmatrix} s(t_1) \\ s(t_2) \\ \vdots \\ s(t_{end}) \end{pmatrix}$
where $s(t_i) \in \mathbb{R}^{\#states}$ is the state at timestep $i$
classQuattrocopter
In this class all physical parameters discribing a specific quattrocopter are stored.


classDyn (abstract)
environment
model
control
state
Abstract Methods
dot
dotD
dotDD
Implemented Methods
testdotD
testdotDD
erbt von
classDynGen (generated)
F
FD
FDD
isEmpty flags
erbt von
classDynImpl
This class is actual initialized in script.m
Methods
dot
dotD
dotDD