

From nutcracking to assisted driving

Stratified instrumental systems
and the modeling of complexity

Second International Engineering Systems Symposium
Engineering Systems: Achievements and Challenges

June 15-17, 2009 at MIT

Maarten Franssen Bjørn Jespersen

Section of Philosophy,
Faculty of Technology, Policy and Management,
Delft University of Technology

m.p.m.franssen@tudelft.nl

b.t.f.jespersen@tudelft.nl

Second International Engineering Systems Symposium
Engineering Systems: Achievements and Challenges

June 15-17, 2009 at MIT

Socio-technical system

Question

What is a *socio-technical* system?

Answer

A socio-technical system (up to some critical point) is a sophisticated *instrumental system*.

Socio-technical system

Any *instrumental system* has this structure:

IT⟨*user*, **CC**⟨*instrument*, *object*⟩⟩

IT: two-place intentional/causal relation

CC: two-place causal relation

user, instrument, object: **slots** for appropriate entities.

Instrumental systems: *process* and *product*

An instrumental system is of a *process* type:

$IT\langle user, CC\langle instrument, object \rangle \rangle$

whereas

$APP(IT\langle user, CC\langle instrument, object \rangle \rangle)$

indicates the *outcome* of an instrumental system, i.e. the transformed object, and is of a *product* type

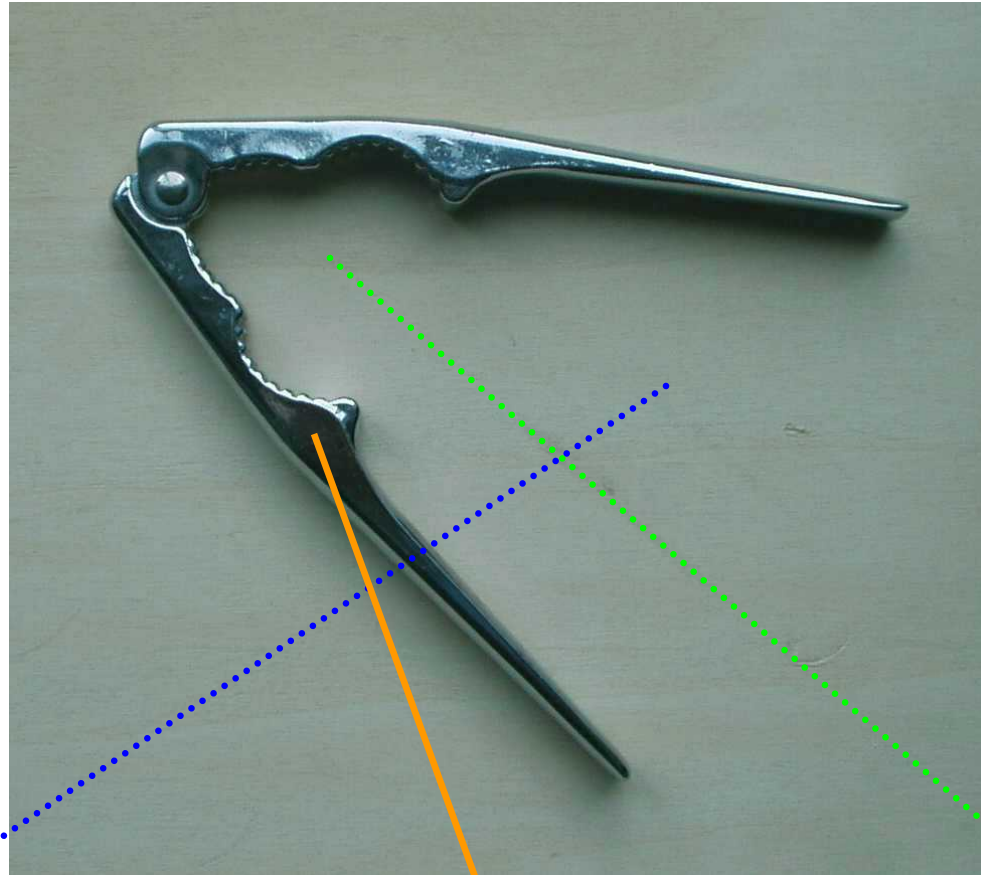
Instrumental system I: nutcracking

IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

the dynamics of nutcracking

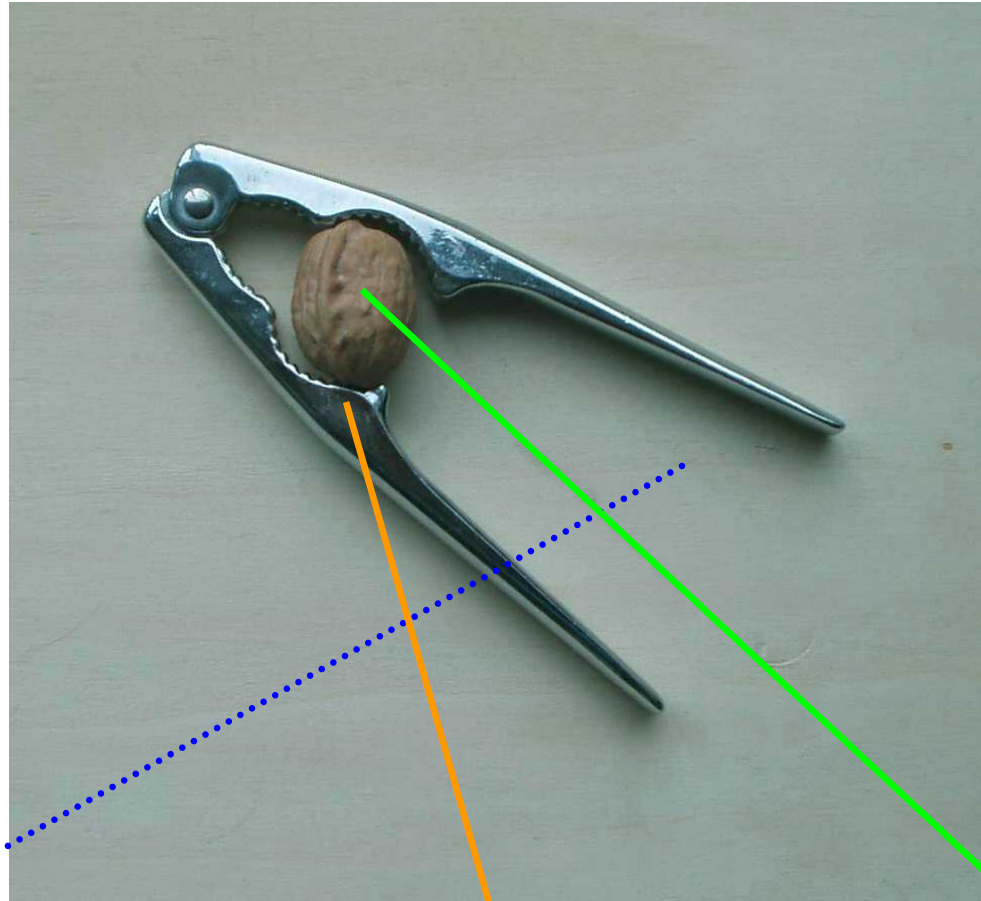
- *transformation* of whole nut at time t_1 into cracked nut at t_2 via the user's manipulation of the instrument according to procedure
- transformation of nut's properties *transforms* user's properties, from having a whole nut at t_1 to having a cracked nut at t_2
- both *user* and *object* must be part of the system in order for the user to transform the object
- the *transformed object* – the cracked nut – is no systemic part but merely a physical part, until removed

A nut-cracking instrument,
but not a nut-cracking system:



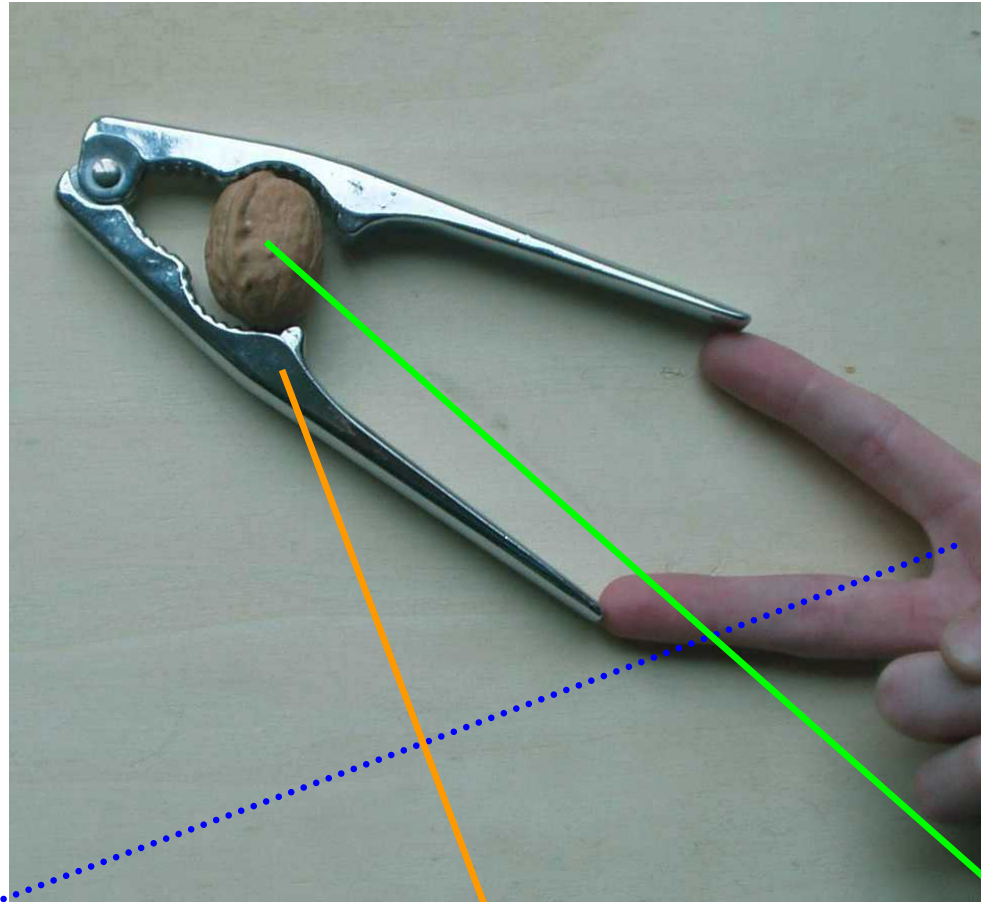
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Still not a nut-cracking system:



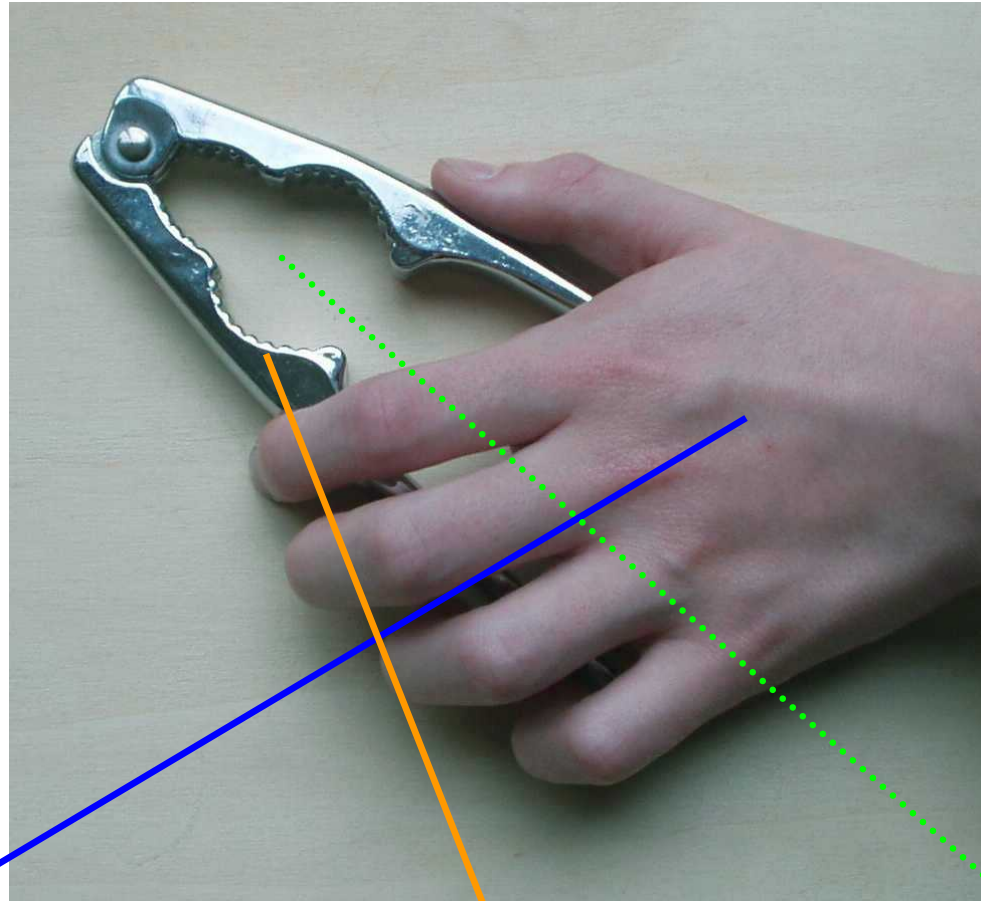
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Still not a nut-cracking system:



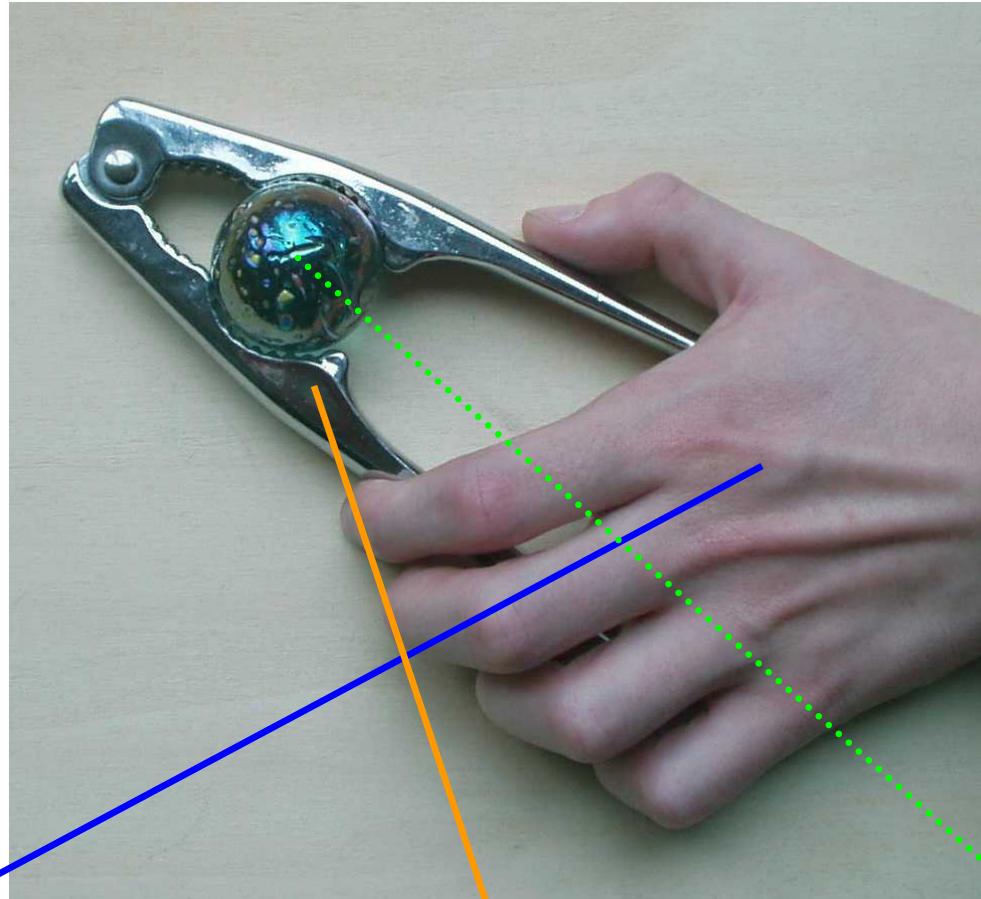
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Still not a nut-cracking system:



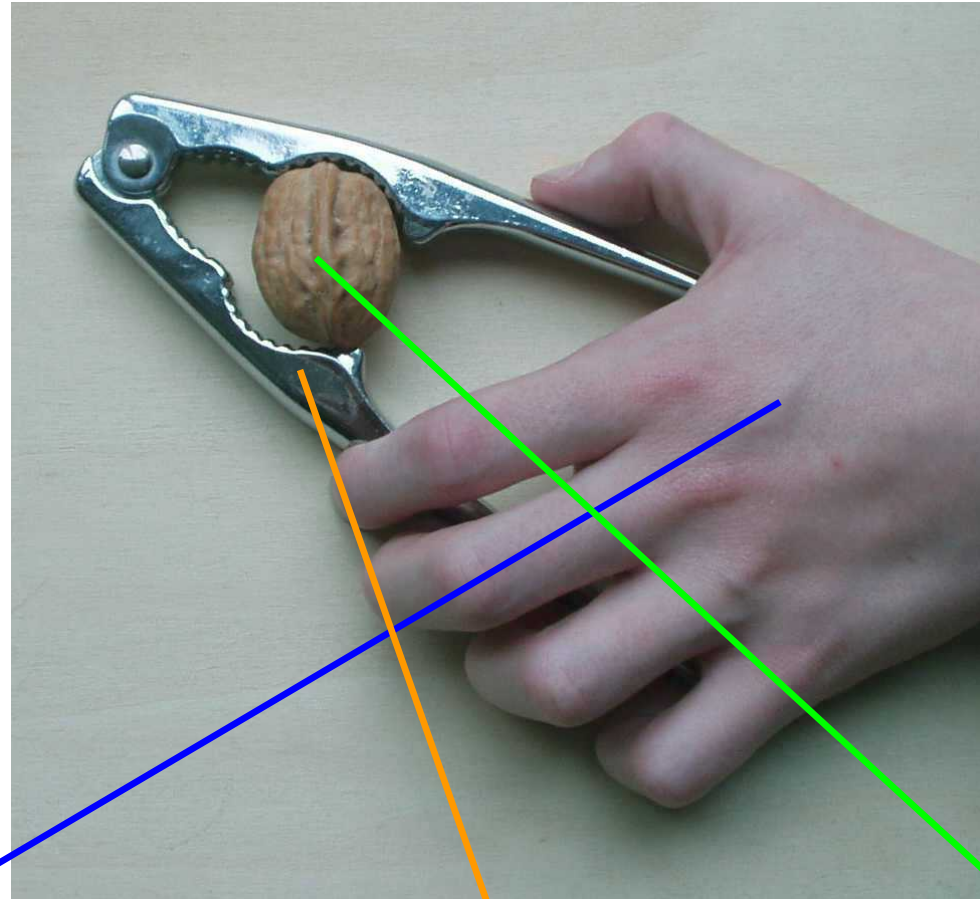
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Still not a nut-cracking system:



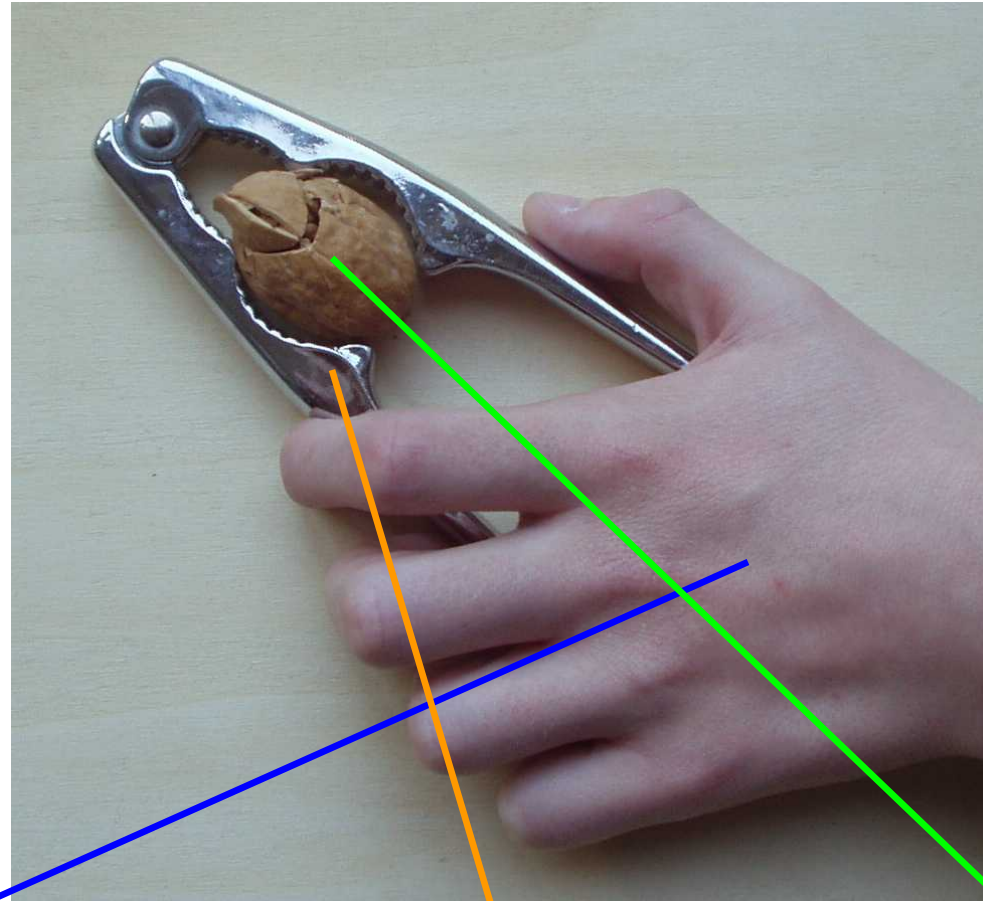
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

One form of an **idle** nut-cracking system:
an *operational* nut-cracking system:



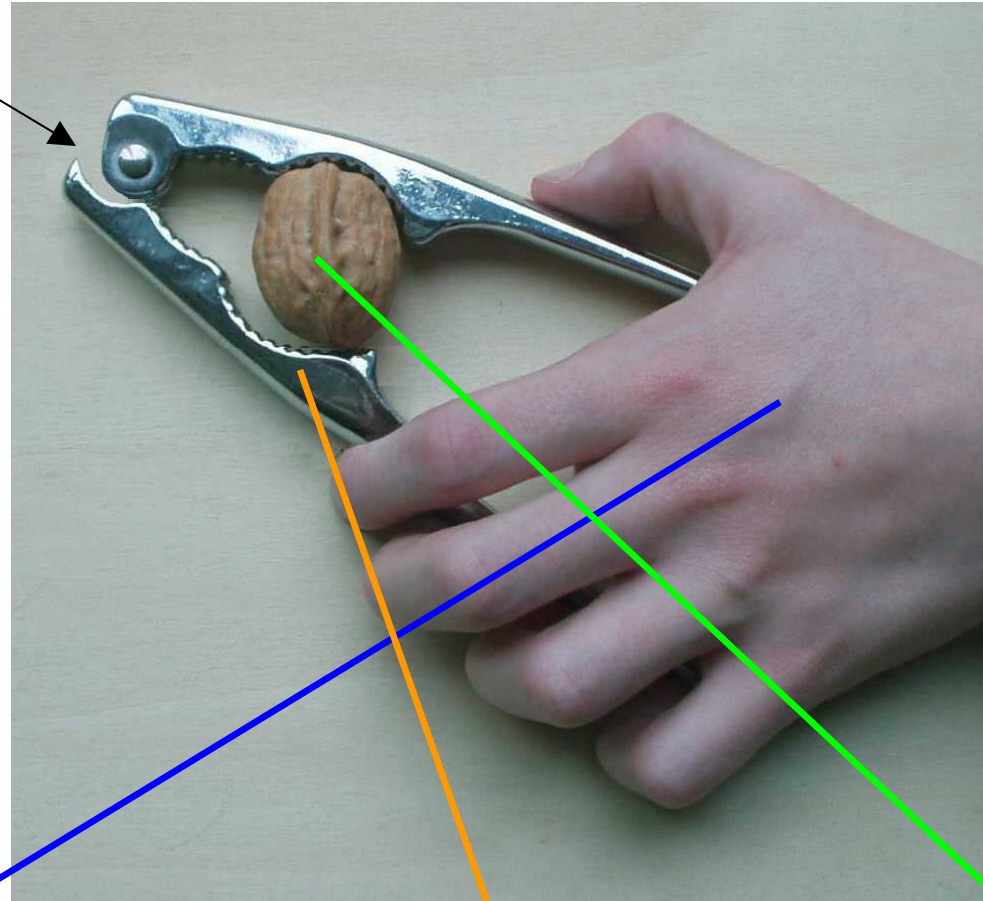
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

An active nut-cracking system:



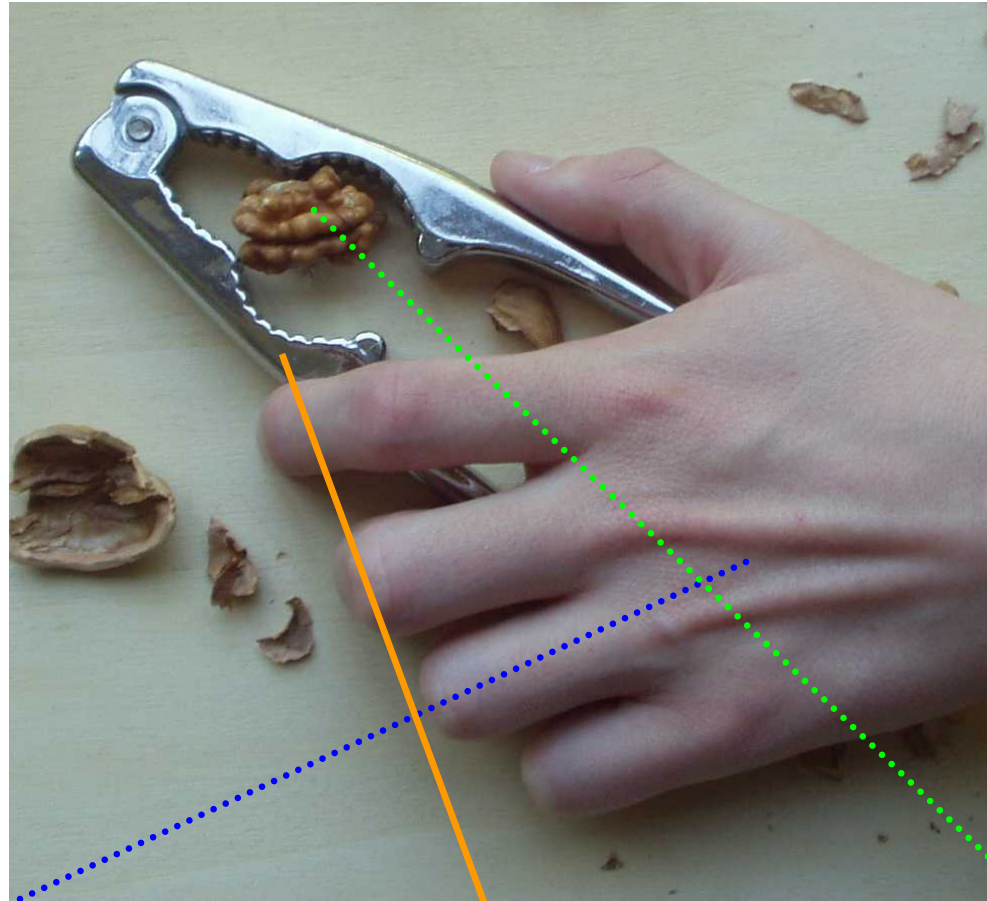
IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Another form of an **idle** nut-cracking system:
a *malfunctioning* nut-cracking system:



IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

No longer a nut-cracking system:



IT⟨*person*, CC⟨*nutcracking device*, *whole nut*⟩⟩

Instrumental system I': driving

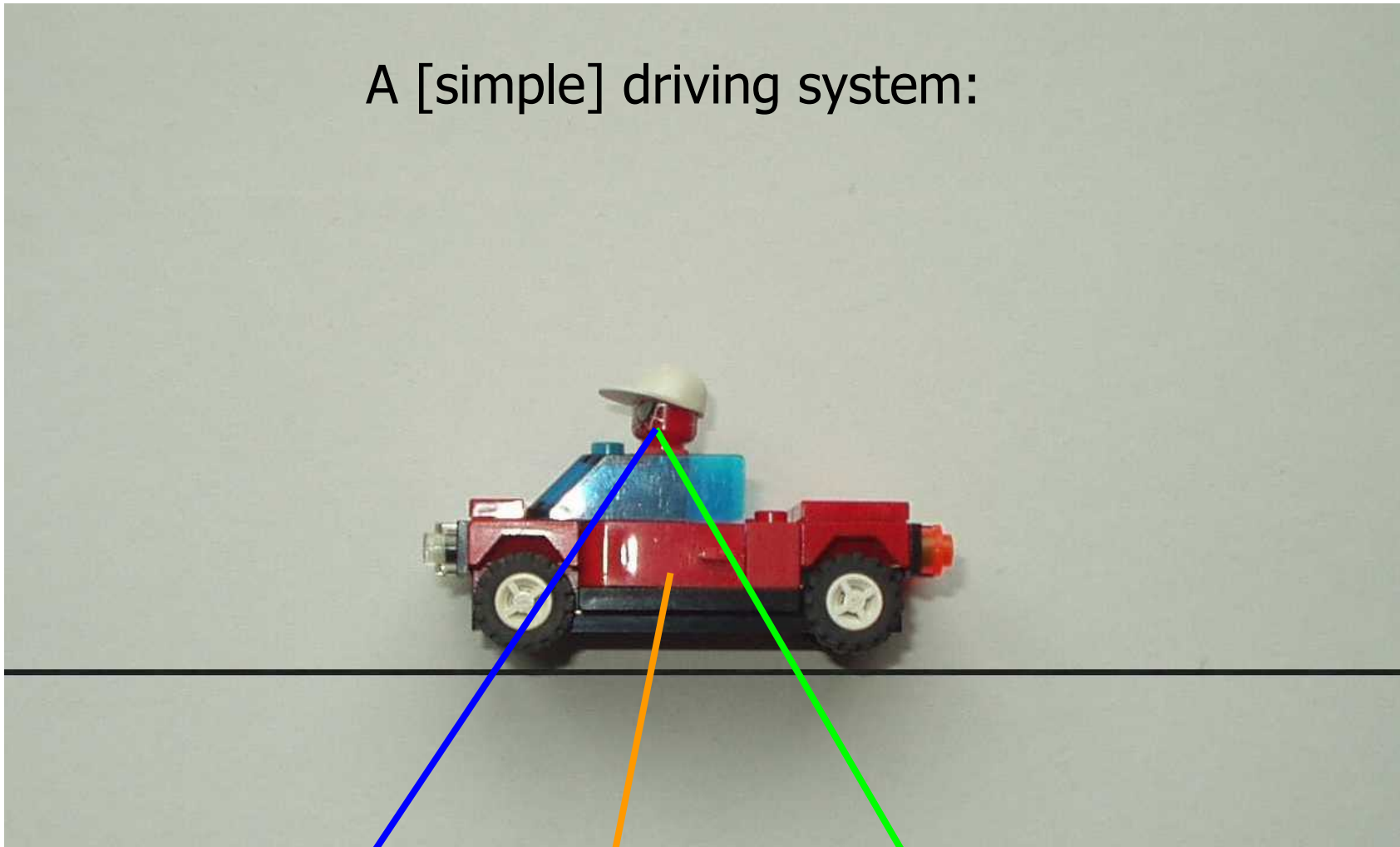
IT⟨*driver*, CC⟨*vehicle*, *passenger*(*n*)⟩⟩

A [simple] driving system:



$IT \langle driver, CC \langle vehicle, passenger(n) \rangle \rangle$

A [simple] driving system:



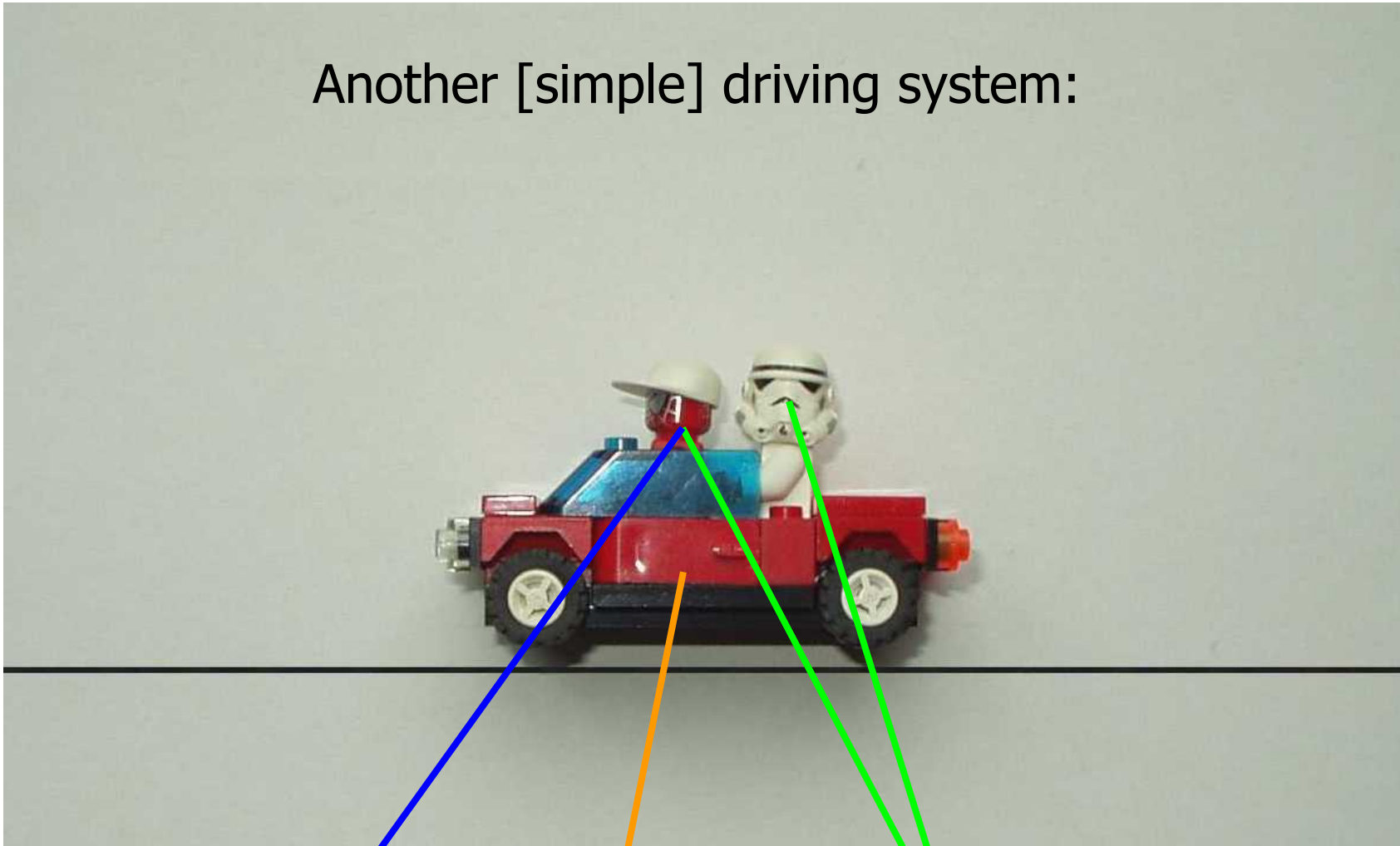
$IT\langle driver, CC\langle vehicle, passenger(n)\rangle\rangle$

Another [simple] driving system:



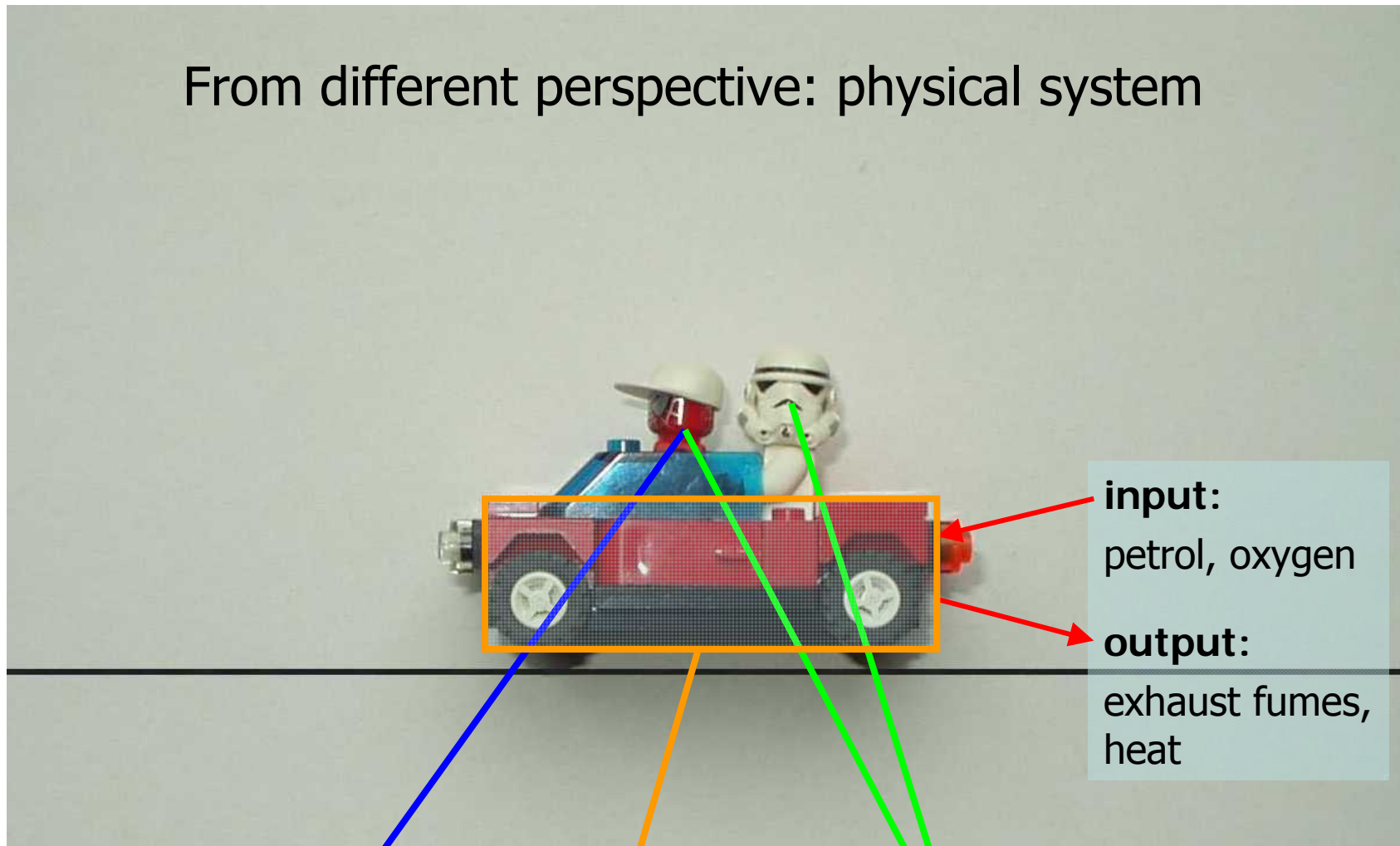
IT⟨driver, CC⟨vehicle, passenger(n)⟩⟩

Another [simple] driving system:



$IT\langle driver, CC\langle vehicle, passenger(n)\rangle\rangle$

From different perspective: physical system



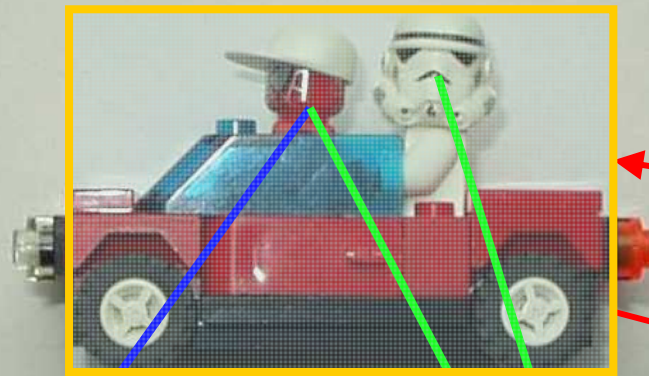
input:
petrol, oxygen

output:
exhaust fumes,
heat

$IT \langle driver, CC \langle \boxed{vehicle}, passenger(n) \rangle \rangle$

physical system

From different perspective: physical system



input:

petrol, oxygen,
food

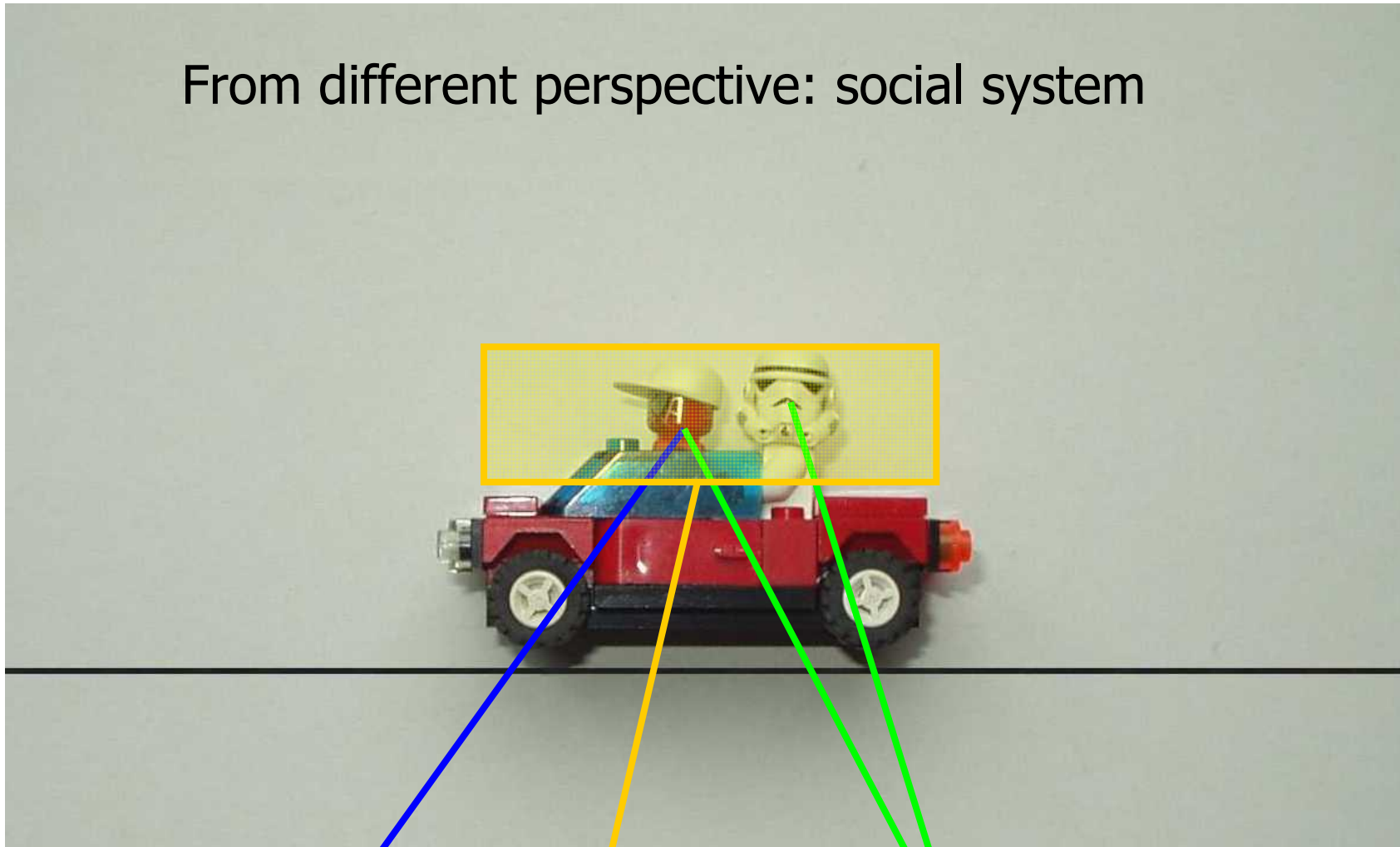
output:

exhaust fumes,
heat, sweat

IT⟨*driver, CC⟨vehicle, passenger(n)⟩⟩*

physical system

From different perspective: social system

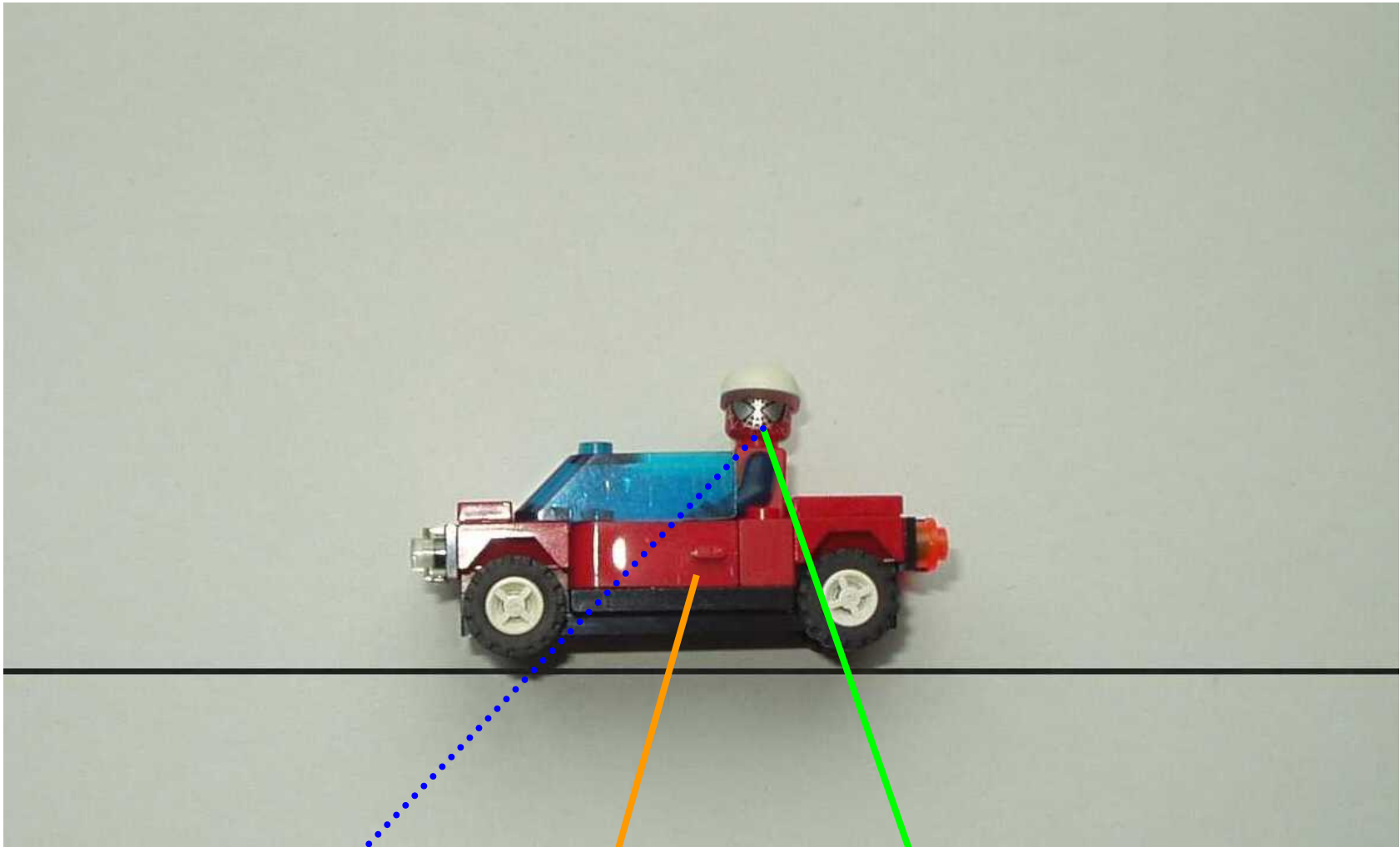


IT⟨*driver*, CC⟨*vehicle*, *passenger*(*n*)⟩⟩

social system



IT⟨driver, CC⟨vehicle, passenger(n)⟩⟩



$IT\langle driver, CC\langle vehicle, passenger(n)\rangle\rangle$

Instrumental system II: assisted driving

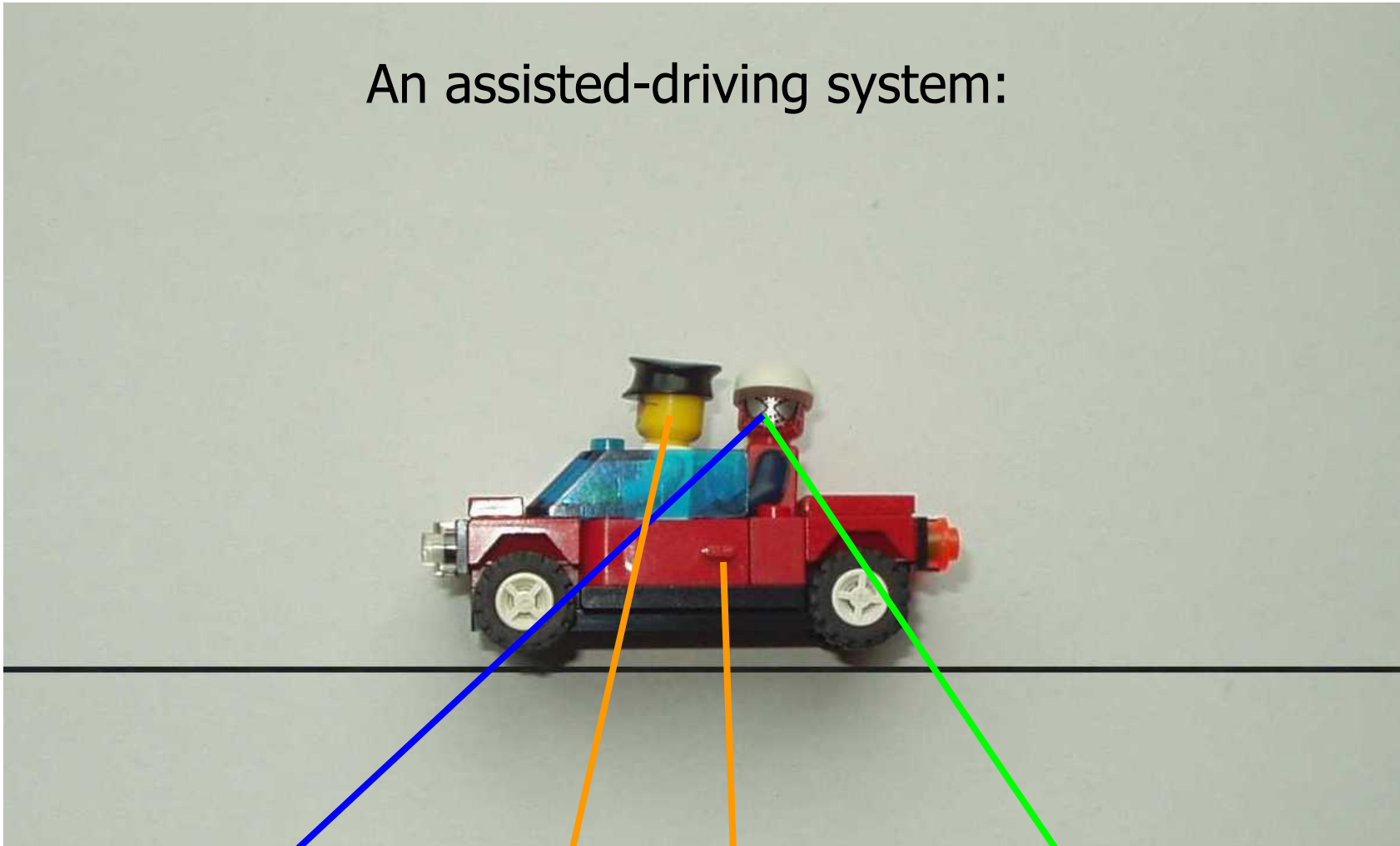
IT⟨*client*, CC⟨IC⟨driver, vehicle⟩, *passenger*(*n*)⟩⟩

An assisted-driving system:



$IT\langle client, CC\langle IC\langle driver, vehicle \rangle, passenger(n) \rangle \rangle$

An assisted-driving system:



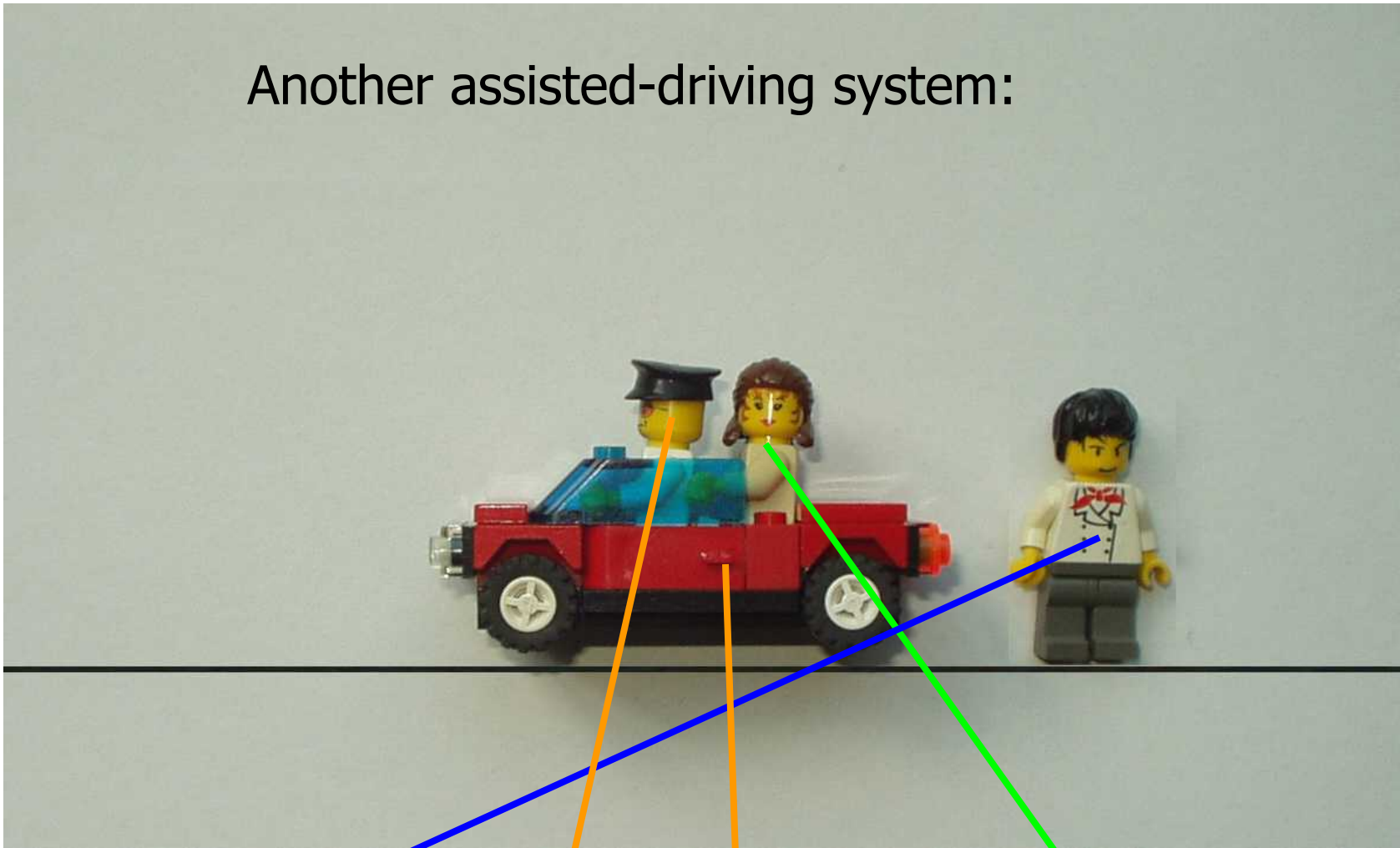
$IT\langle client, CC\langle IC\langle driver, vehicle \rangle, passenger(n) \rangle \rangle$

Another assisted-driving system:



$IT\langle client, CC\langle IC\langle driver, vehicle \rangle, passenger(n) \rangle \rangle$

Another assisted-driving system:



$IT\langle client, CC\langle IC\langle driver, vehicle \rangle, passenger(n) \rangle \rangle$

Taxonomy of nested instrumental systems

Three *primā facie* possible ways for a slot to be filled by an instrumental system:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩, object⟩⟩

IT⟨IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩, CC⟨instrument, object⟩⟩

On second thoughts, two possible ways for a slot to be filled by an instrumental system remain:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩, object⟩⟩

~~IT⟨IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩, CC⟨instrument, object⟩⟩~~

A particular that realizes a complete instrumental system is a *process* type and therefore not the right kind of thing to fill the user slot of an instrumental system.

Two possible ways for a slot to be filled by an instrumental system:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument, object'*⟩⟩, object⟩⟩

φ-ing-system-regulating system

Two possible ways for a slot to be filled by an instrumental system:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument, object'*⟩⟩, object⟩⟩

Benefiting-from- ϕ -ing system

Three possible ways for a slot to be filled by the *product* of an instrumental system:

IT⟨user, CC⟨instrument, APP(IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩)⟩⟩

IT⟨user, CC⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), object⟩⟩

IT⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), CC⟨instrument, object⟩⟩

Three *primā facie* possible **non-instrumental systems**:

REL_{UI} ⟨user, instrument⟩

REL_{UO} ⟨user, object⟩

REL_{IO} ⟨instrument, object⟩

Only two **non-instrumental systems** make sense:

REL_{UI} ⟨user, instrument⟩

~~REL_{UO} ⟨user, object⟩~~

REL_{IO} ⟨instrument, object⟩

A coupling of a *user* to an *object* without the mediation of an instrument does not make a system.

Six *primā facie* possible ways for a slot to be filled by non-instrumental systems IC⟨U, I⟩ or CT⟨I, O⟩:

IT⟨user, CC⟨instrument, IC⟨user', instrument'⟩⟩⟩

IT⟨user, CC⟨IC⟨user', instrument⟩, object⟩⟩

IT⟨IC⟨user, instrument'⟩, CC⟨instrument, object⟩⟩

IT⟨user, CC⟨instrument, CT⟨instrument', object'⟩⟩⟩

IT⟨user, CC⟨CT⟨instrument', object'⟩, object⟩⟩

IT⟨CT⟨instrument', object'⟩, CC⟨instrument, object⟩⟩

On second thoughts, four ways for a slot to be filled by non-instrumental systems $IC\langle U, I \rangle$ or $CT\langle I, O \rangle$ remain:

$IT\langle \text{user}, CC\langle \text{instrument}, IC\langle \text{user}', \text{instrument}' \rangle \rangle \rangle$

$IT\langle \text{user}, CC\langle IC\langle \text{user}', \text{instrument} \rangle, \text{object} \rangle \rangle$

~~$IT\langle IC\langle \text{user}, \text{instrument}' \rangle, CC\langle \text{instrument}, \text{object} \rangle \rangle$~~

$IT\langle \text{user}, CC\langle \text{instrument}, CT\langle \text{instrument}', \text{object} \rangle \rangle \rangle$

$IT\langle \text{user}, CC\langle CT\langle \text{instrument}', \text{object} \rangle, \text{object} \rangle \rangle$

~~$IT\langle CT\langle \text{instrument}', \text{object} \rangle, CC\langle \text{instrument}, \text{object} \rangle \rangle$~~

Again, process-type entities are ruled out as *users*

Six *primā facie* possible ways for a slot to be filled by the **product** of a non-instrumental system $IC\langle U, I \rangle$ or $CT\langle I, O \rangle$:

IT⟨user, CC⟨instrument, $APP(IC\langle user', instrument' \rangle) \rangle \rangle$

IT⟨user, CC⟨ $APP(IC\langle user', instrument \rangle)$, object⟩⟩

IT⟨ $APP(IC\langle user, instrument' \rangle)$, CC⟨instrument, object⟩⟩

IT⟨user, CC⟨ instrument, $APP(CT\langle instrument', object \rangle) \rangle \rangle$

IT⟨user, CC⟨ $APP(CT\langle instrument', object \rangle)$, object⟩⟩

IT⟨ $APP(CT\langle instrument', object \rangle)$, CC⟨instrument, object⟩⟩

On second thoughts, only the ways where a slot is filled by the product of a $CT\langle I, O \rangle$ system remain:

~~IT⟨user, CC⟨instrument, APP(IC⟨user', instrument'⟩)⟩⟩~~

~~IT⟨user, CC⟨APP(IC⟨user', instrument'⟩), object⟩⟩~~

~~IT⟨APP(IC⟨user, instrument'⟩), CC⟨instrument, object⟩⟩~~

IT⟨user, CC⟨instrument, APP(CT⟨instrument', object'⟩)⟩

IT⟨user, CC⟨APP(CT⟨instrument', object'⟩), object⟩⟩

IT⟨APP(CT⟨instrument', object'⟩), CC⟨instrument, object⟩⟩

IC⟨U, I⟩ has no object that is transformed, so no product.

Generating instrumental systems of arbitrary complexity:

(i) $IT\langle U, CC\langle I, O \rangle \rangle$ is an instrumental system;

(ii) In any instrumental system:

(1) I can be filled by $IT\langle U, CC\langle I, O \rangle \rangle$ to give an instrumental system;

(2) O can be filled by $IT\langle U, CC\langle I, O \rangle \rangle$ to give an instrumental system;

(3) I can be filled by $IC\langle U, I \rangle$ to give an instrumental system;

(4) O can be filled by $IC\langle U, I \rangle$ to give an instrumental system;

(5) I can be filled by $CT\langle I, O \rangle$ to give an instrumental system;

(6) O can be filled by $CT\langle I, O \rangle$ to give an instrumental system;

(7) U can be filled by $APP(IT\langle U, CC\langle I, O \rangle \rangle)$ to give an instrumental system;

(8) I can be filled by $APP(IT\langle U, CC\langle I, O \rangle \rangle)$ to give an instrumental system;

(9) O can be filled by $APP(IT\langle U, CC\langle I, O \rangle \rangle)$ to give an instrumental system;

(10) U can be filled by $APP(CT\langle I, O \rangle)$ to give an instrumental system;

(11) I can be filled by $APP(CT\langle I, O \rangle)$ to give an instrumental system;

(12) O can be filled by $APP(CT\langle I, O \rangle)$ to give an instrumental system.

Summary

- Instrumental systems are *dynamic* input/output *transformation* systems
- Instrumental systems are *hybrid* systems, encompassing *intentionality and causality*
- Instrumental systems can become *arbitrarily stratified*, hence complex, through *substitution* of components by systemic entities

Examples

Two possible ways for a slot to be filled by an instrumental system:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument, object'*⟩⟩, object⟩⟩

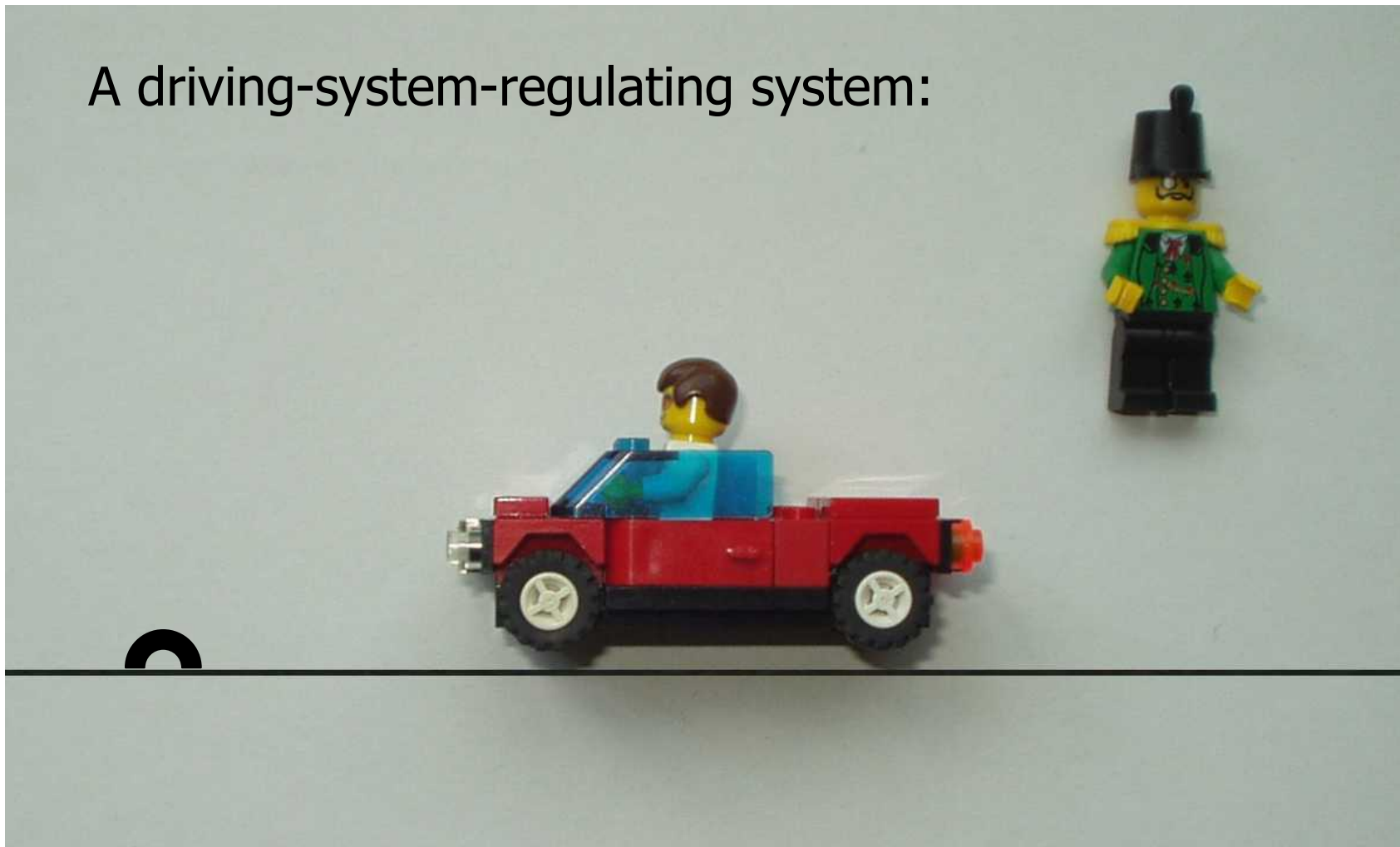
φ-ing-system-regulating system

A driving-system-regulating system:



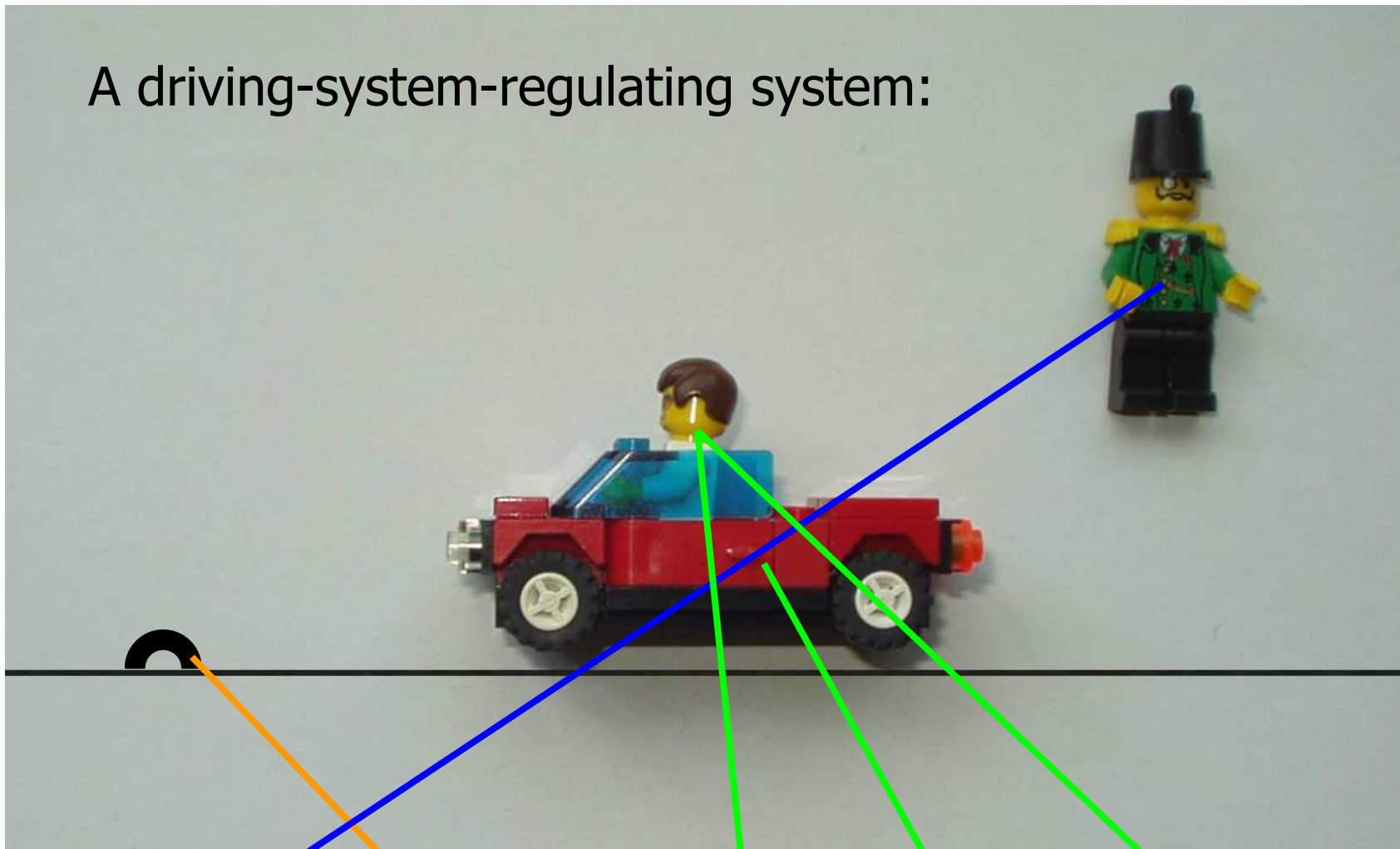
IT⟨system owner, CC⟨instrument, IT⟨driver, CC⟨vehicle, passenger(*n*)⟩⟩⟩⟩

A driving-system-regulating system:



IT⟨system owner, CC⟨instrument, IT⟨driver, CC⟨vehicle, passenger(*n*)⟩⟩⟩⟩

A driving-system-regulating system:



IT⟨system owner, CC⟨instrument, IT⟨driver, CC⟨vehicle, passenger(*n*)⟩⟩⟩⟩

Two possible ways for a slot to be filled by an instrumental system:

IT⟨user, CC⟨instrument, IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩⟩⟩

IT⟨user, CC⟨IT⟨*user'*, CC⟨*instrument, object'*⟩⟩, object⟩⟩

Benefiting-from- ϕ -ing system

A benefiting-from-driving system:



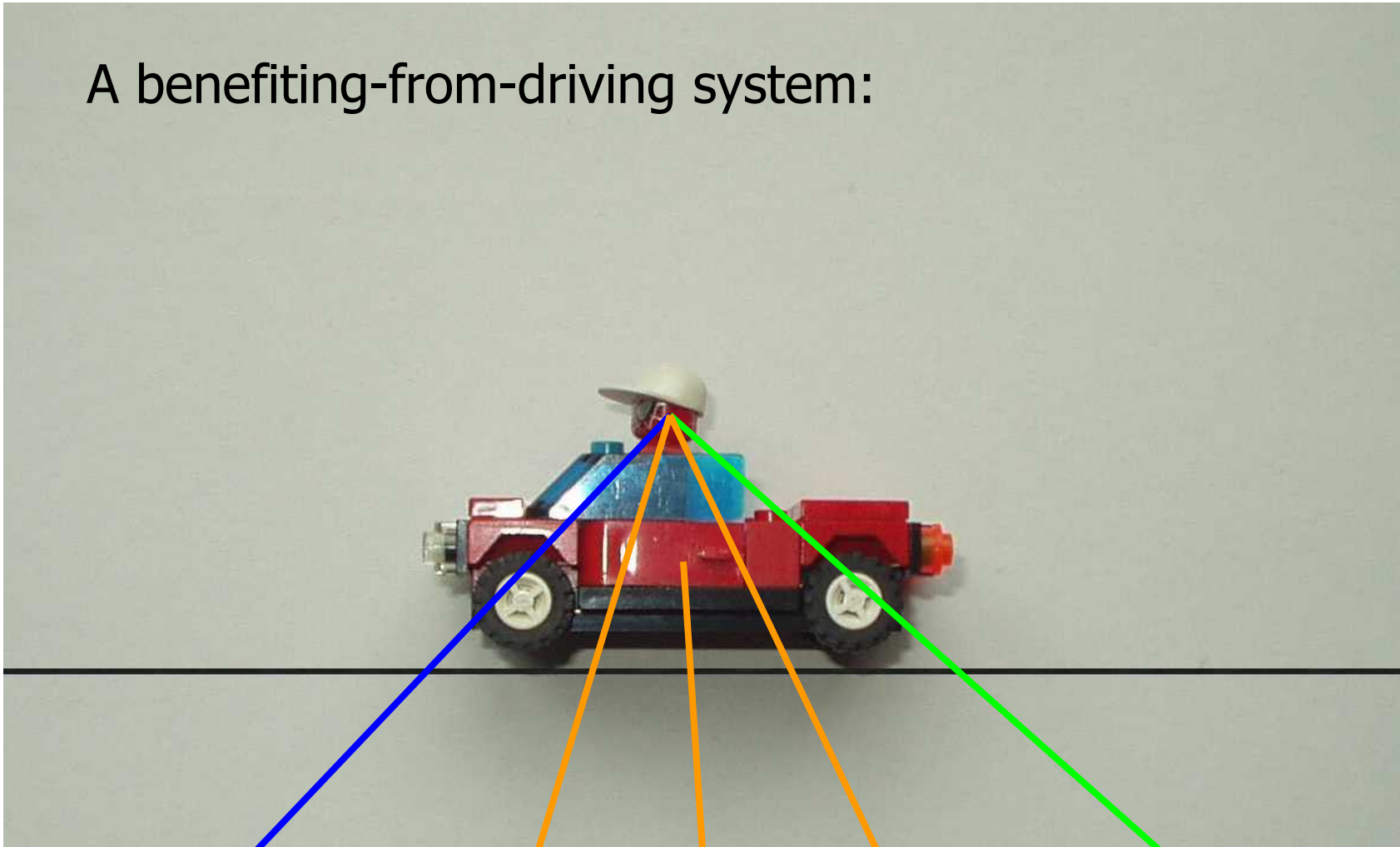
IT⟨system owner, CC⟨IT⟨driver, CC⟨vehicle, passenger(n)⟩⟩, beneficiary⟩⟩

A benefiting-from-driving system:



IT⟨system owner, CC⟨IT⟨driver, CC⟨vehicle, passenger(n)⟩⟩, beneficiary⟩⟩

A benefiting-from-driving system:



IT⟨system owner, CC⟨IT⟨driver, CC⟨vehicle, passenger(n)⟩⟩, beneficiary⟩⟩

A benefiting-from-assisted-driving system:



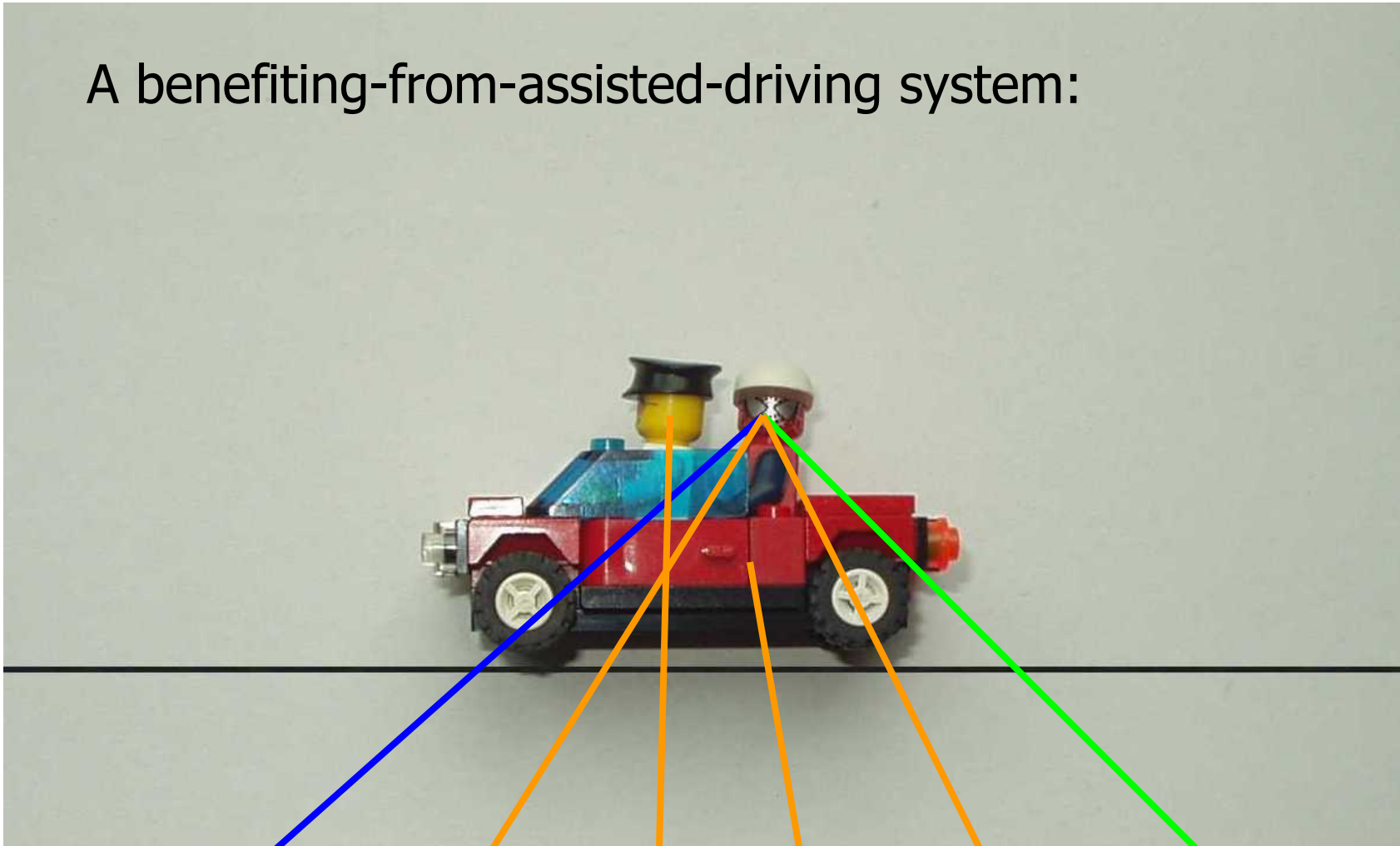
IT⟨syst. owner, CC⟨IT⟨client, CC⟨IC⟨driver, vehicle⟩, passenger(*n*)⟩⟩, beneficiary⟩⟩

A benefiting-from-assisted-driving system:



IT⟨syst. owner, CC⟨IT⟨client, CC⟨IC⟨driver, vehicle⟩, passenger(*n*)⟩⟩, beneficiary⟩⟩

A benefiting-from-assisted-driving system:



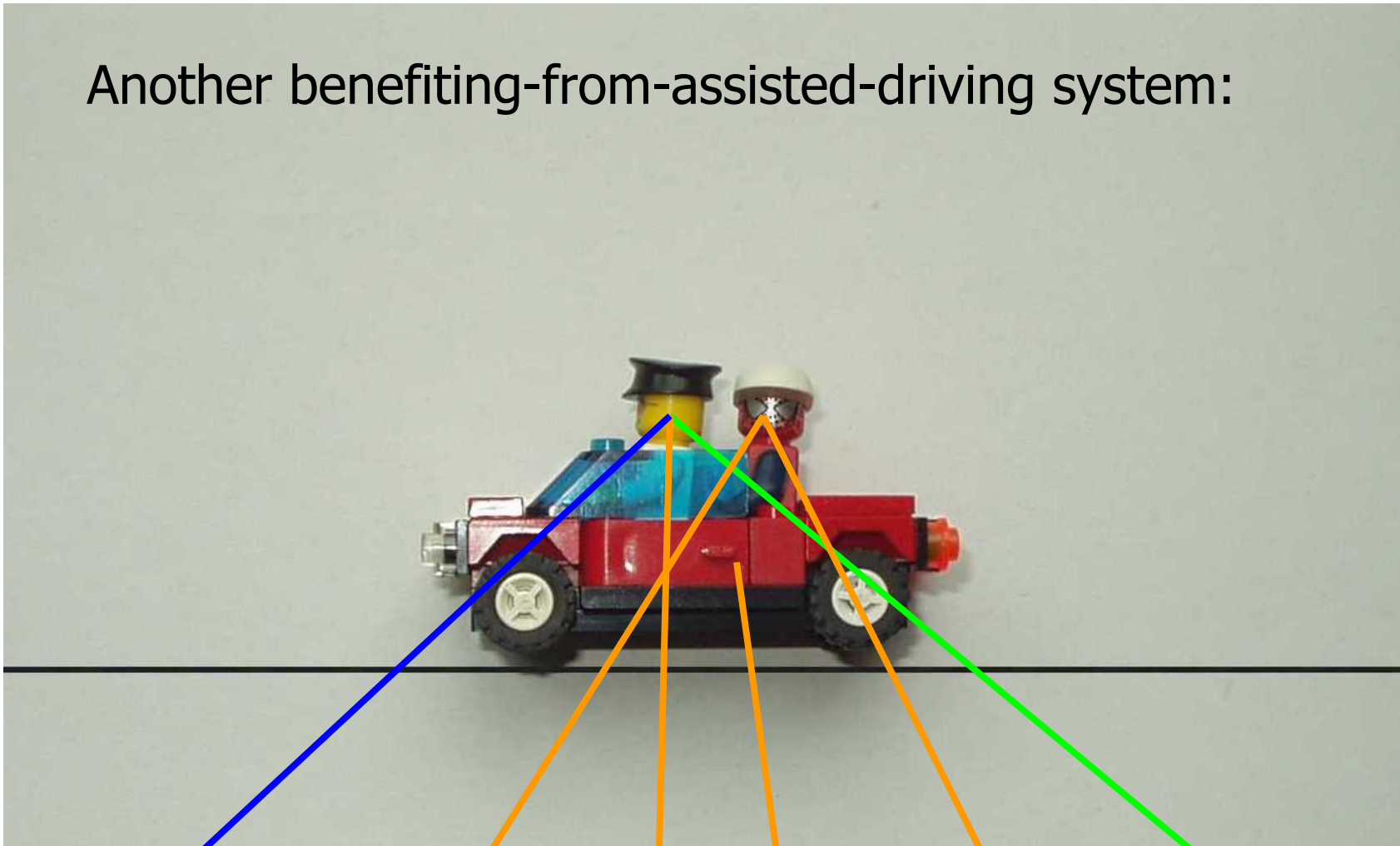
IT⟨syst. owner, CC⟨IT⟨client, CC⟨IC⟨driver, vehicle⟩, passenger(*n*)⟩⟩, beneficiary⟩⟩

Another benefiting-from-assisted-driving system:



IT⟨syst. owner, CC⟨IT⟨client, CC⟨IC⟨driver, vehicle⟩, passenger(*n*)⟩⟩, beneficiary⟩⟩

Another benefiting-from-assisted-driving system:



IT⟨syst. owner, CC⟨IT⟨client, CC⟨IC⟨driver, vehicle⟩, passenger(*n*)⟩⟩, beneficiary⟩⟩

Four ways for a slot to be filled by non-instrumental systems $IC\langle U, I \rangle$ or $CT\langle I, O \rangle$:

$IT\langle \text{user}, CC\langle \text{instrument}, IC\langle \text{user}', \text{instrument}' \rangle \rangle \rangle$

$IT\langle \text{user}, CC\langle IC\langle \text{user}', \text{instrument}' \rangle, \text{object} \rangle \rangle$

$IT\langle IC\langle \text{user}, \text{instrument}' \rangle, CC\langle \text{instrument}, \text{object} \rangle \rangle$

$IT\langle \text{user}, CC\langle \text{instrument}, CT\langle \text{instrument}', \text{object}' \rangle \rangle \rangle$

$IT\langle \text{user}, CC\langle CT\langle \text{instrument}', \text{object}' \rangle, \text{object} \rangle \rangle$

an assisted-driving system

Three possible ways for a slot to be filled by the *product* of an instrumental system:

IT⟨user, CC⟨instrument, APP(IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩)⟩⟩

IT⟨user, CC⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), object⟩⟩

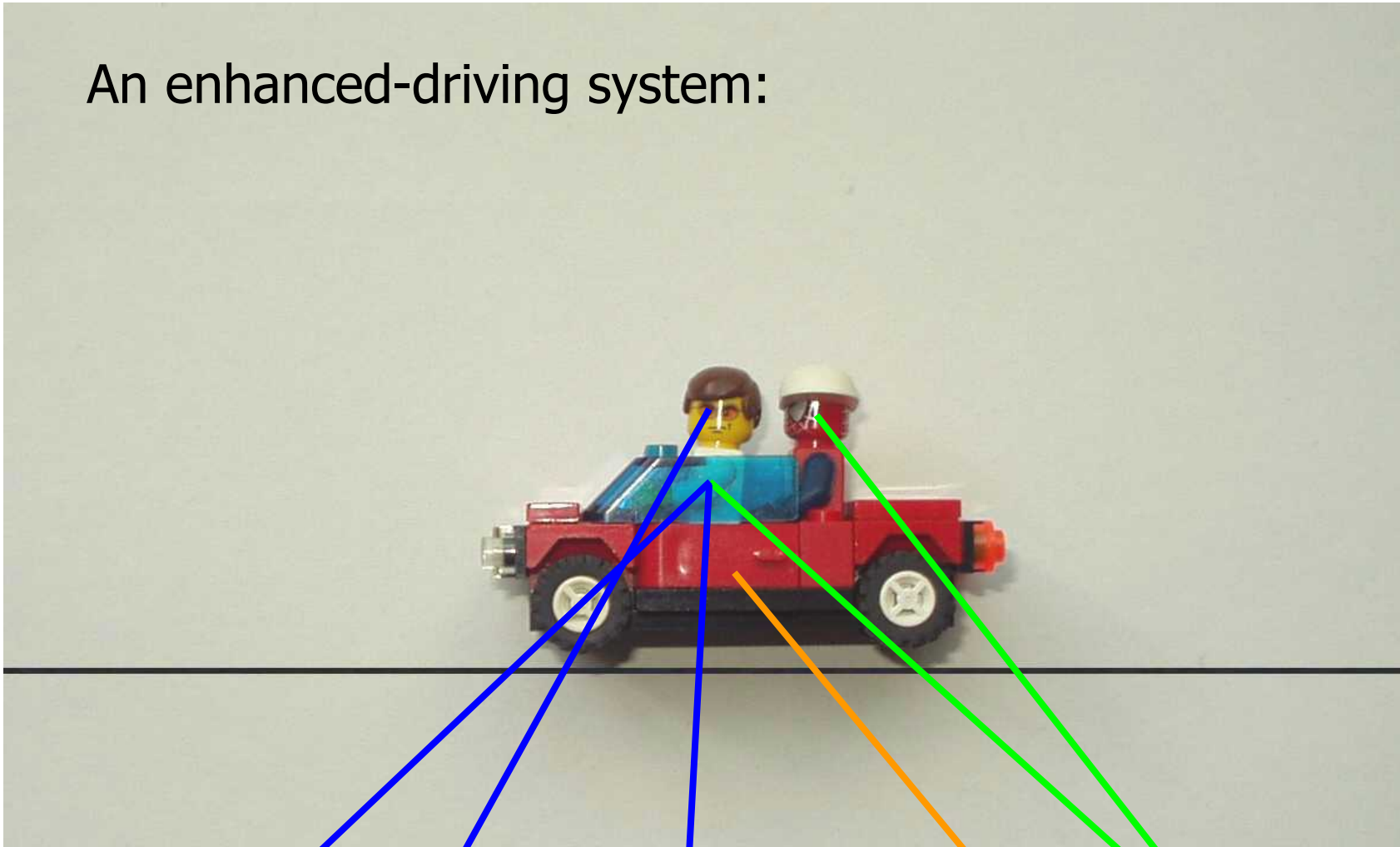
IT⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), CC⟨instrument, object⟩⟩

An enhanced-driving system:



IT⟨APP(IT⟨user, CC⟨glasses, shortsighted person⟩⟩), CC⟨vehicle, passenger(*n*)⟩⟩

An enhanced-driving system:



IT⟨APP(IT⟨user, CC⟨glasses, shortsighted person⟩⟩), CC⟨vehicle, passenger(*n*)⟩⟩

Three possible ways for a slot to be filled by the *product* of an instrumental system:

IT⟨user, CC⟨instrument, APP(IT⟨*user'*, CC⟨*instrument'*, *object*⟩⟩)⟩⟩

IT⟨user, CC⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), object⟩⟩

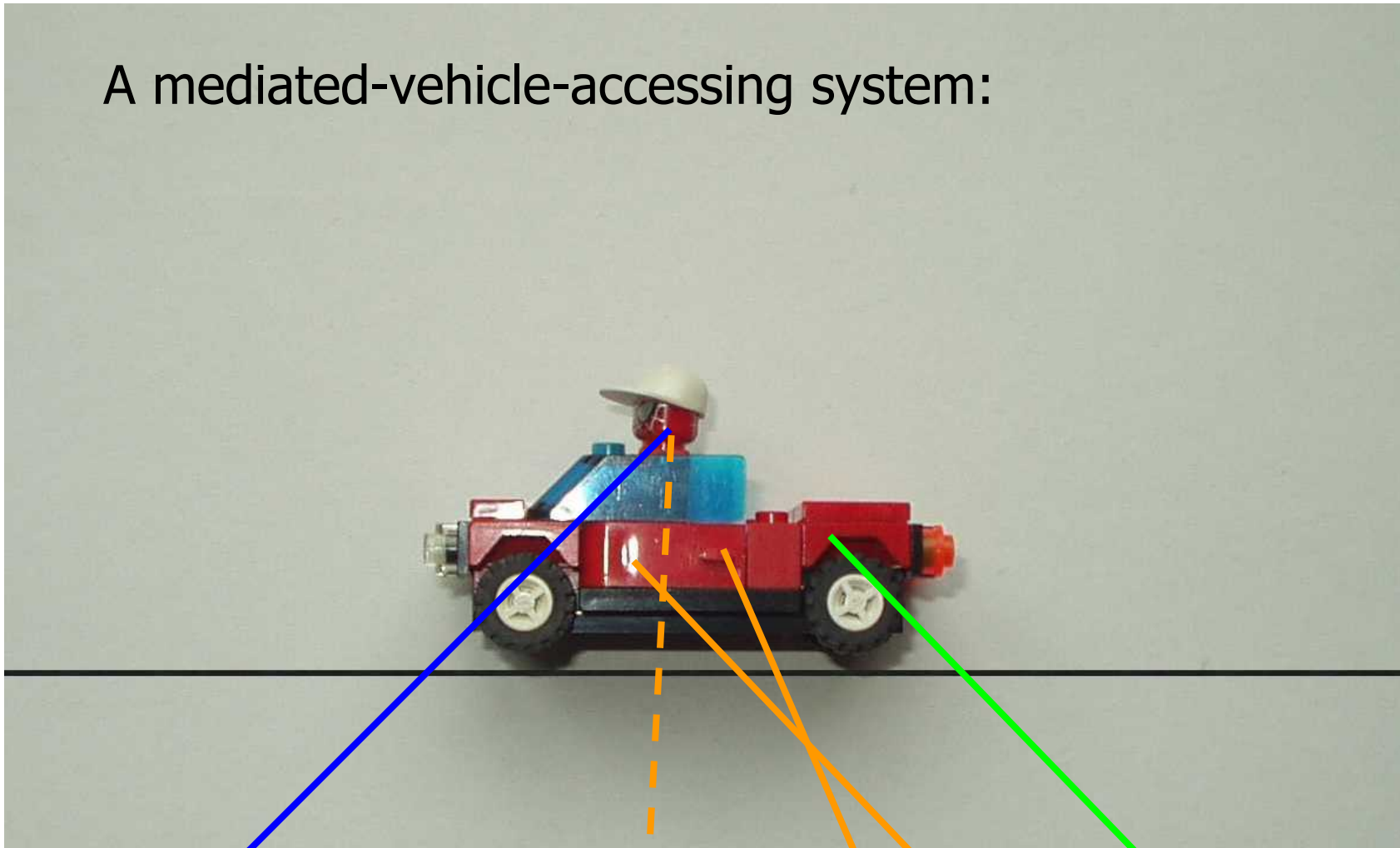
IT⟨APP(IT⟨*user'*, CC⟨*instrument'*, *object'*⟩⟩), CC⟨instrument, object⟩⟩

A mediated-vehicle-accessing system:



IT⟨user, CC⟨APP(IT⟨unlocker, CC⟨key, door⟩⟩), vehicle⟩⟩

A mediated-vehicle-accessing system:



IT⟨user, CC⟨APP(IT⟨unlocker, CC⟨key, door⟩⟩), vehicle⟩⟩

Three possible ways for a slot to be filled by the *product* of an instrumental system:

IT⟨user, CC⟨instrument, APP(IT⟨*user*', CC⟨*instrument*', *object*'⟩⟩)⟩⟩

IT⟨user, CC⟨APP(IT⟨*user*', CC⟨*instrument*', *object*'⟩⟩), *object*⟩⟩

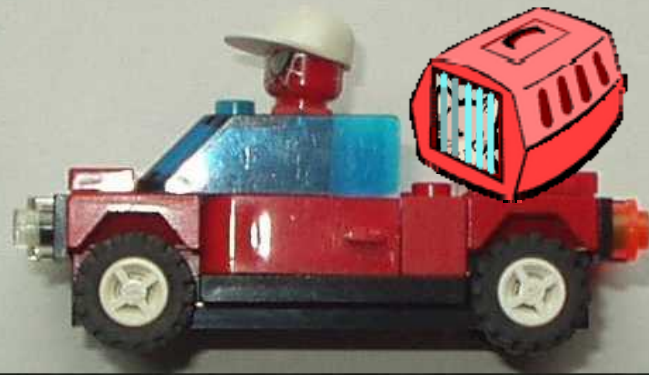
IT⟨APP(IT⟨*user*', CC⟨*instrument*', *object*'⟩⟩), CC⟨instrument, *object*⟩⟩

A (transporting-goods-by-)driving system:



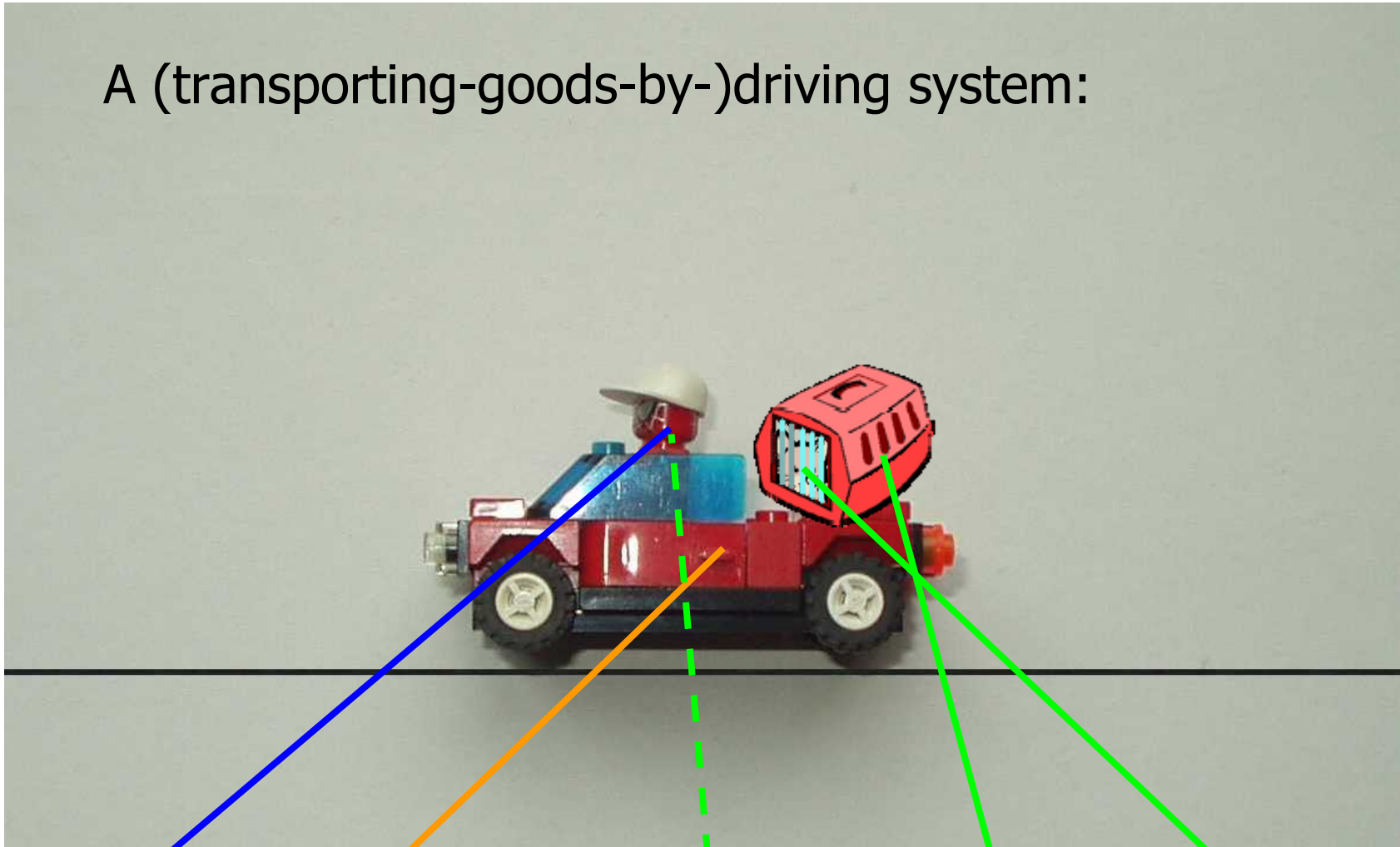
IT⟨driver, CC⟨vehicle, cargo(*n*)⟩⟩

A (transporting-goods-by-)driving system:



IT⟨driver, CC⟨vehicle, APP(IT⟨packer, CC⟨container, cargo(*n*)⟩)⟩⟩

A (transporting-goods-by-)driving system:



IT⟨driver, CC⟨vehicle, APP(IT⟨packer, CC⟨container, cargo(*n*)⟩)⟩⟩