

# Architecting System Performance; Defining Performance

by *Gerrit Muller* TNO-ESI, University College of South East Norway

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

## Abstract

Performance is a broad term. Each domain has its own key performance parameters. Performance can be used to indicate time-oriented performance, such as response time, throughput, or productivity. However, more broadly, it may be used for aspects like image quality, spatial performance (f.i. positioning accuracy), energy or power properties, sensitivity and specificity of algorithms, or reliability and availability.

### Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

June 5, 2018

status: preliminary

draft

version: 0.1

<i>time-oriented</i> response time latency throughput productivity	<i>spatial</i> positioning accuracy working envelope range turning cycle	<i>reliability</i> MTBF MTTR uptime unscheduled breaks
<i>energy/power</i> energy consumption range standby time maximum power heat release cooling capacity	<i>algorithmic</i> sensitivity specificity accuracy coverage	<i>image quality</i> sharpness contrast color consistency color rendition streakiness uniformity

# Performance Attributes

## *time-oriented*

response time  
latency  
throughput  
productivity

## *spatial*

positioning accuracy  
working envelope  
range  
turning cycle

## *reliability*

MTBF  
MTTR  
uptime  
unscheduled breaks

## *energy/power*

energy consumption  
range  
standby time  
maximum power  
heat release  
cooling capacity

## *algorithmic*

sensitivity  
specificity  
accuracy  
coverage

## *image quality*

sharpness  
contrast  
color consistency  
color rendition  
streakiness  
uniformity

# Defining Performance

performance is a function of:

context

perception depends on individual human characteristics

circumstances

operation of interest

} scenario  
use case<sup>1</sup>

system of interest

specification

design

configuration

version

history

} generic, valid for the class of systems  
normal and special cases  
(worst case, degraded, exceptions, ...)

} instance specific

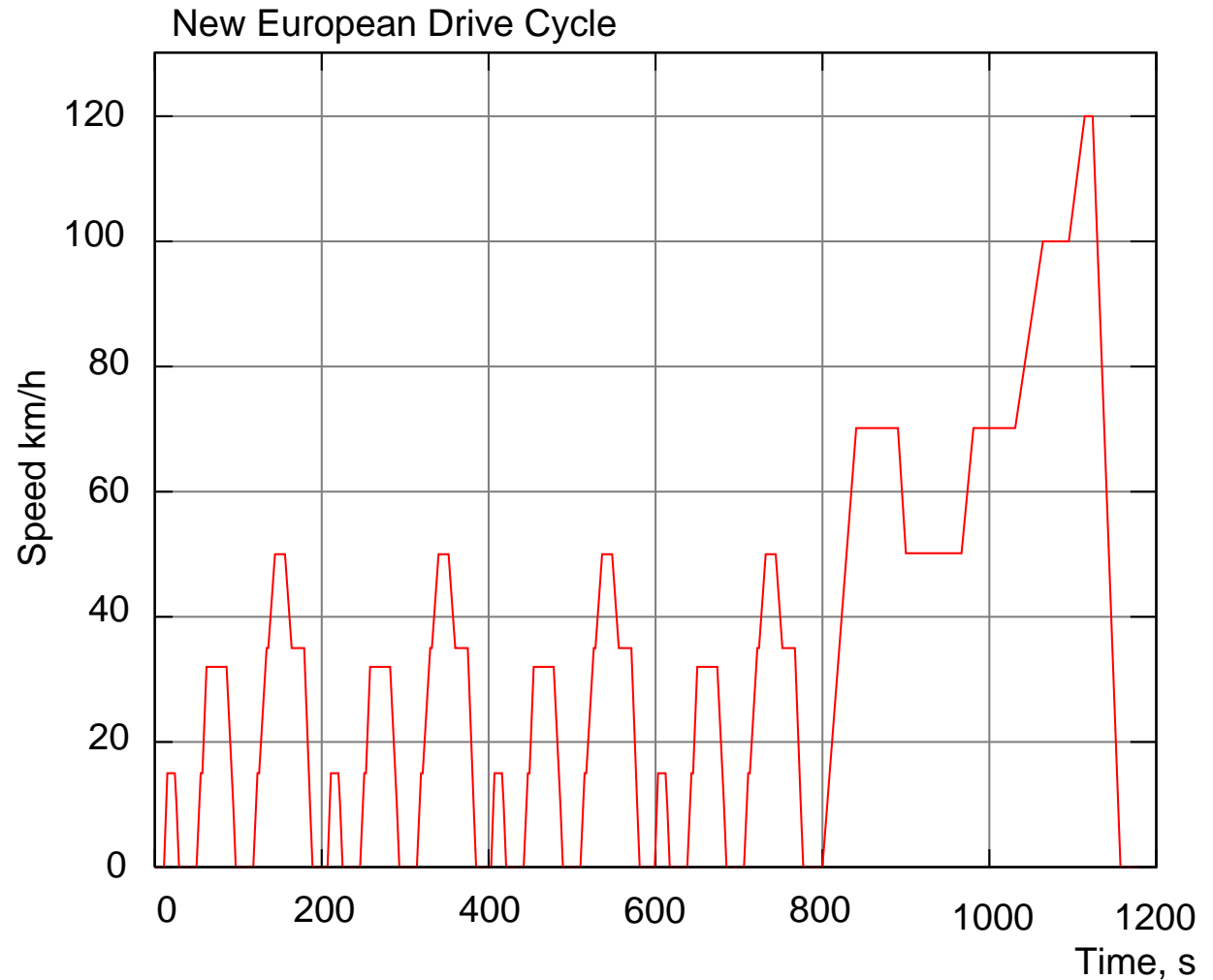
<sup>1</sup>a use case in this context is rich (includes quantifications) and broad (covers the operation of interest, not a single function)

# Example EV Range Definition

## Electric Vehicle Driving Range

Range = f(  
v(t),  
Circumstances,  
Driving style,  
Car load,  
Charging state,  
Battery age)

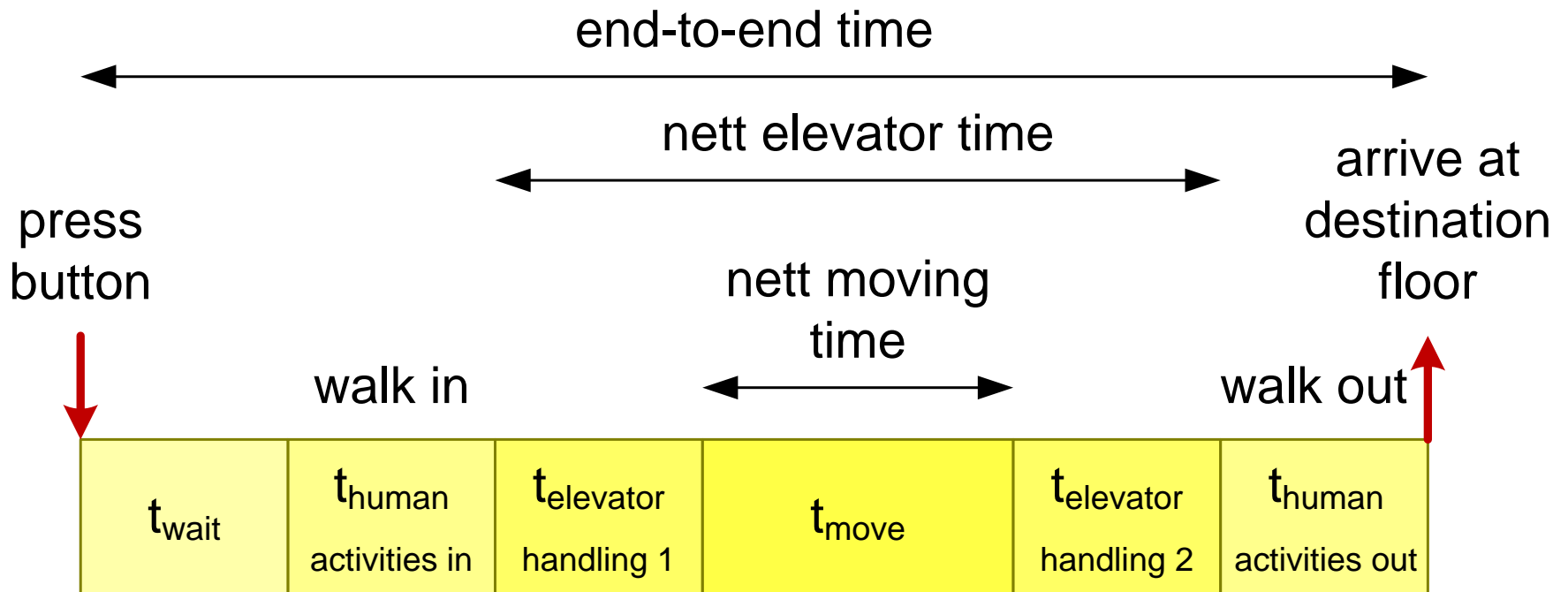
A quantified Use Case  
defines under what  
circumstances the EV will  
achieve the specified  
range.



[http://en.wikipedia.org/wiki/New\\_European\\_Driving\\_Cycle#/media/File:New\\_European\\_Driving\\_Cycle.svg](http://en.wikipedia.org/wiki/New_European_Driving_Cycle#/media/File:New_European_Driving_Cycle.svg)  
Published under GFDL, thanks to Orzetto

# End-to-End Performance

The **end-to-end** performance is the relevant performance as the **stakeholder** experiences it: from **initial trigger** to **final result**.



$$t_{end-to-end} = t_{human \text{ activities}} + t_{wait} + t_{elevator \text{ handling}} + t_{move}$$