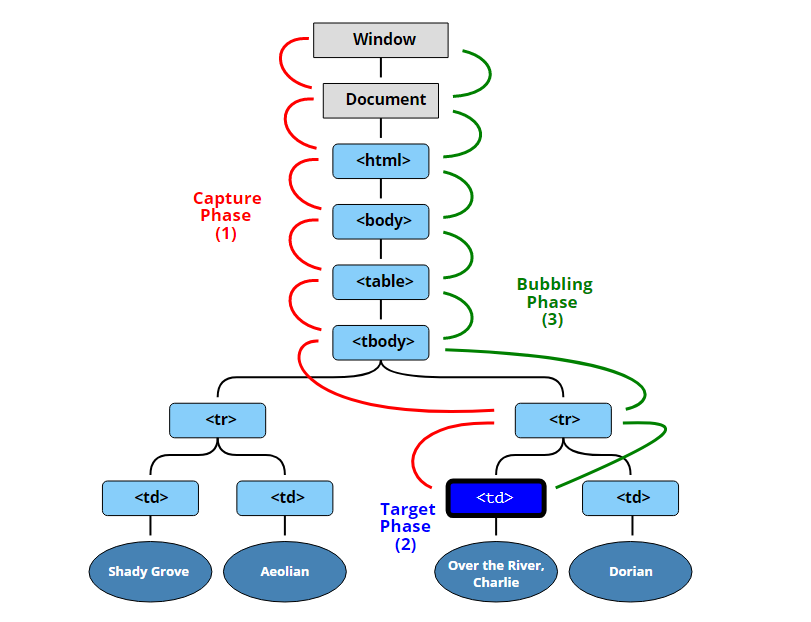
1)   
Bubbling  
**When an event happens on an element, it first runs the handlers on it, then on its parent, then all the way up on other ancestors.**

Event Capturing  
There’s another phase of event processing called “capturing”. It is rarely used in real code, but sometimes can be useful.

The standard [DOM Events](http://www.w3.org/TR/DOM-Level-3-Events/) describes 3 phases of event propagation:

1. Capturing phase – the event goes down to the element.
2. Target phase – the event reached the target element.
3. Bubbling phase – the event bubbles up from the element.



Stop Propagonation

with **event.stopPropagation()**

# Difference this/currentTarget/target:

* target is the element that triggered the event (e.g., the user clicked on)
* currentTarget is the element that the event listener is attached to.
* This refers to currentTarget if Listener direct defined (not in closure => undefinded)

# What happens if parent elements have event handlers attached?

They are triggered after reverse DOM hierarchy.

# What happens if parent and child elements have event handlers attached?

First child and then parents in reverse order (starting from deepest nested)