

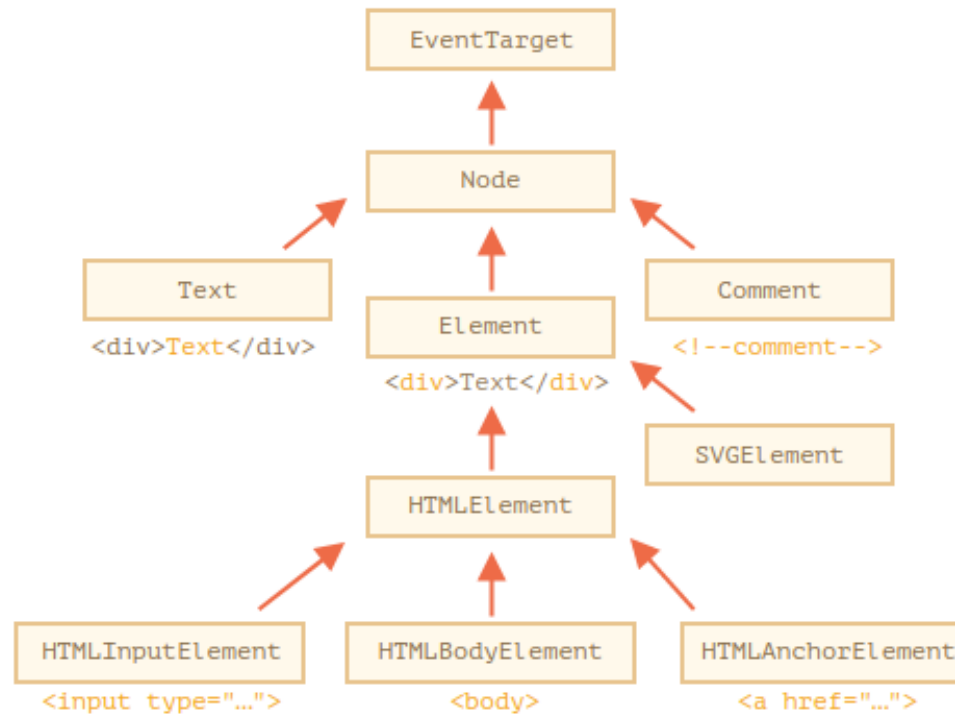
Источники событий

- Мышь (`click`, `mouseover`)
- Клавиатура (`keydown`, `keyup`)
- Элементы управления (`focus`, `submit`, `change`)
- Документ в браузере (`DOMContentLoaded`)
- Окно браузера (`scroll`)

Откуда берутся названия тегов, объектов и событий?

- **W3C** (World Wide Web Consortium). С 1994 года
Спецификация CSSOM <https://www.w3.org/TR/cssom-1/>
- **WHATWG** (Web Hypertext Application Technology Working Group). С 2004 года
Apple, Google, Microsoft, Mozilla
Стандарты HTML + BOM, DOM, ... <https://spec.whatwg.org/>

На какие события может реагировать DOM-элемент?



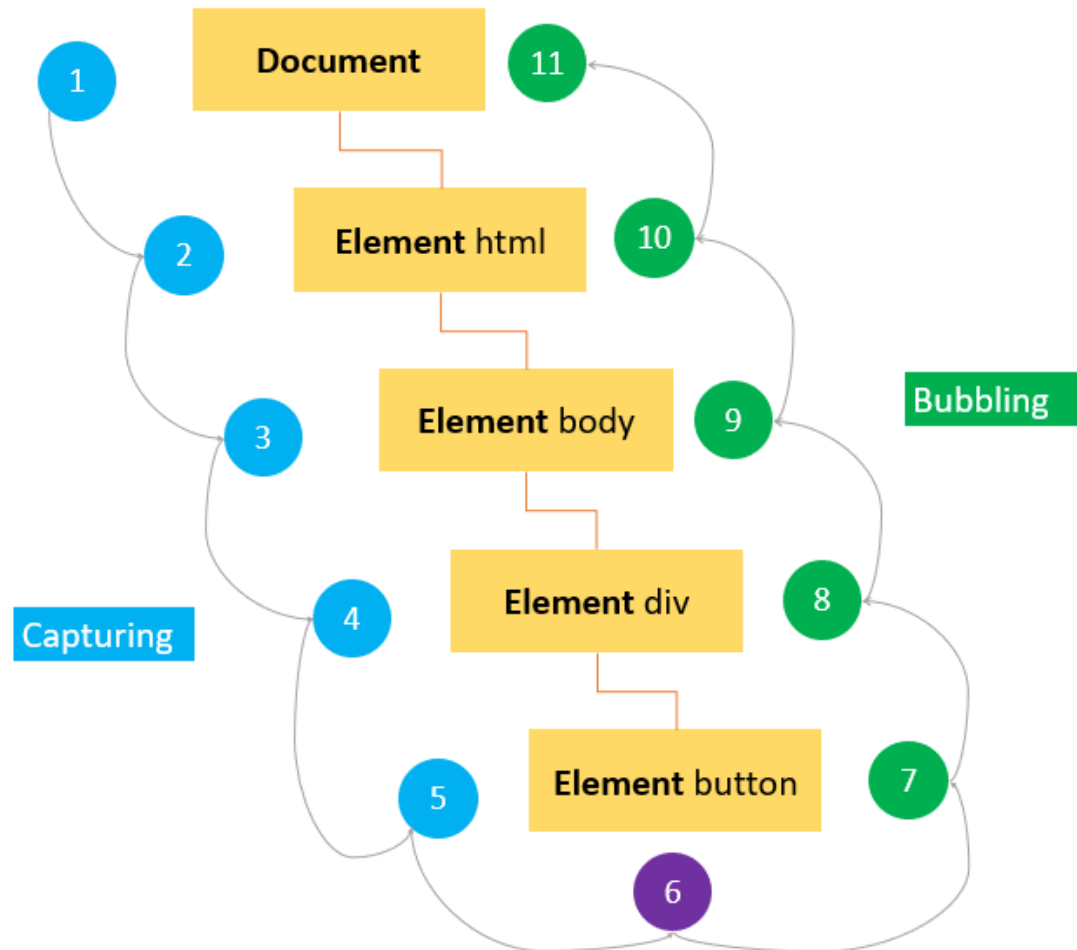
Свойства `on<событие>` у DOM-элементов

Назначение обработчика события на DOM-элементе

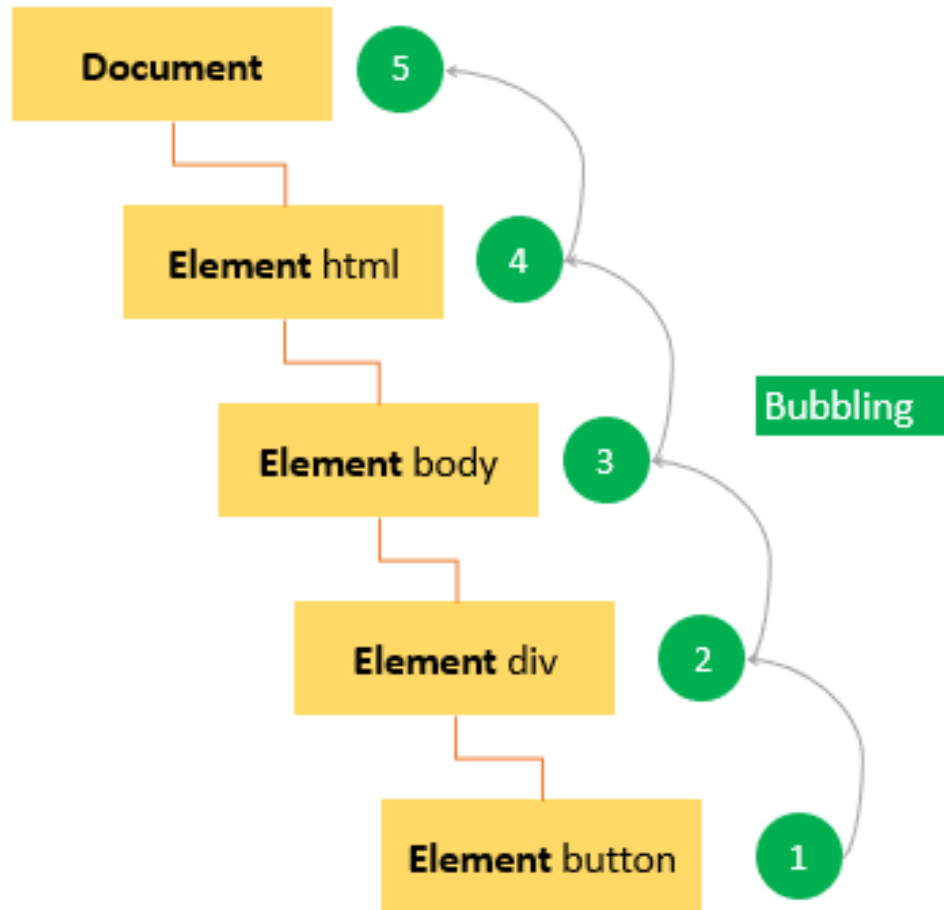
- **Атрибут HTML-тега.** `onclick="..."`
- **DOM-свойство.** `elem.onclick = function`
- **Метод** `elem.addEventListener('click', function)`

В функцию-обработчик передаётся объект с информацией о событии

Погружение и всплытие событий



Всплытие событий



Делегирование событий

1. Вешаем обработчик на контейнер.
2. В обработчике проверяем исходный элемент `event.target`.
3. Если событие произошло внутри нужного нам элемента, то обрабатываем его.

Преимущества

- Упрощает инициализацию и экономит память: не нужно вешать много обработчиков.
- Меньше кода - при добавлении и удалении элементов не нужно ставить или снимать обработчики.
- Можно декларативно добавлять элементам поведение, устанавливая HTML-атрибуты или классы.