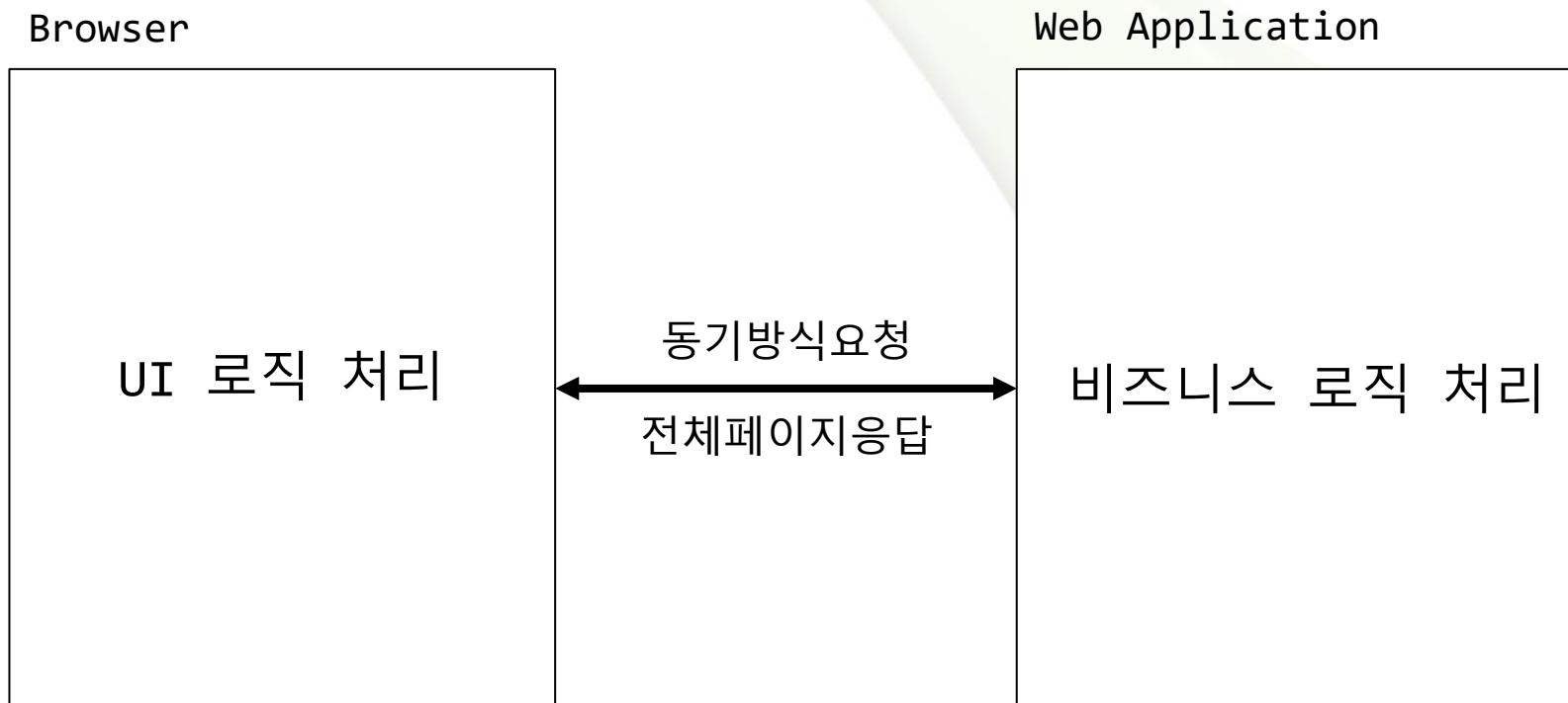


The background features a series of overlapping, wavy, ribbon-like shapes in various shades of green and white, creating a dynamic, flowing effect. The colors range from a deep forest green at the bottom to a bright lime green and pale yellow-green at the top.

# **Introduction to Ajax**

# 전통적인 웹 응용프로그램의 문제

- 동기 방식 요청 / 응답 구조
  - » 네트워크 지연 등의 요인으로 반응성 저하
- 페이지 단위 요청 / 응답 구조
  - » 불필요한 데이터도 같이 전송되어 효율성 저하



# AJAX 정의

- 전통적인 웹 응용프로그램의 문제점과 한계를 개선하기 위한 비동기 통신 기술
- 브라우저에서 UI 데이터 이외에 비즈니스 로직 데이터를 처리 가능
  - » 브라우저를 비즈니스 로직을 처리할 수 있는 Rich Client로 구현하는 것이 가능해짐
- 비동기 통신 기술, 동적 클라이언트 스크립팅 기법, 표준 데이터 포맷을 이용
  - » SOAP, XML과 같은 통신 기술을 사용하여 비동기 요청/응답 처리
  - » 자바스크립트, DOM, HTML 및 CSS와 같은 프리젠테이션 기술을 사용해서 응답 처리

Asynchronous

비동기 방식 처리

JavaScript

브라우저에서 작동하는 스크립트 언어

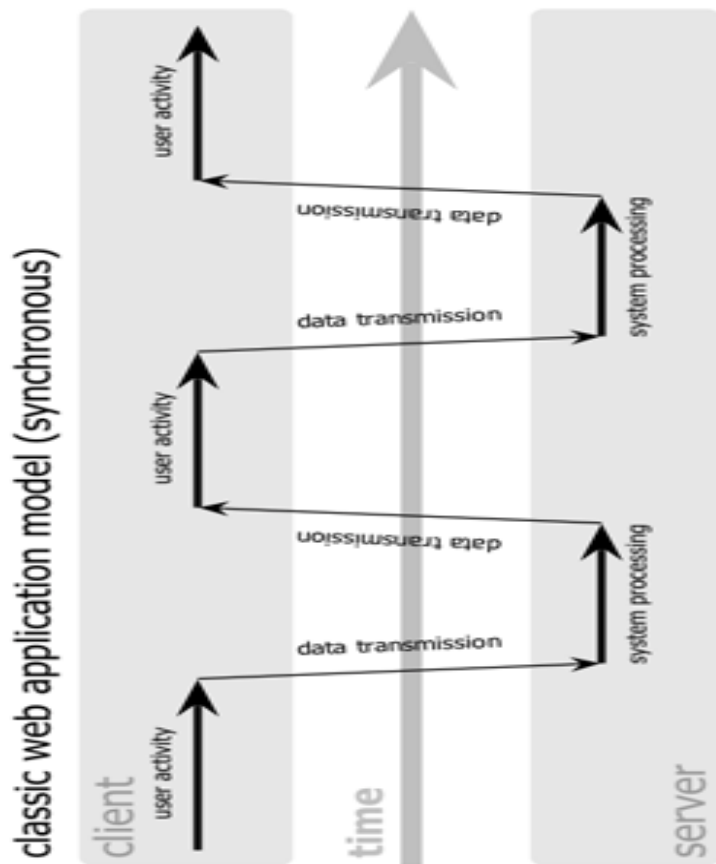
And

XML

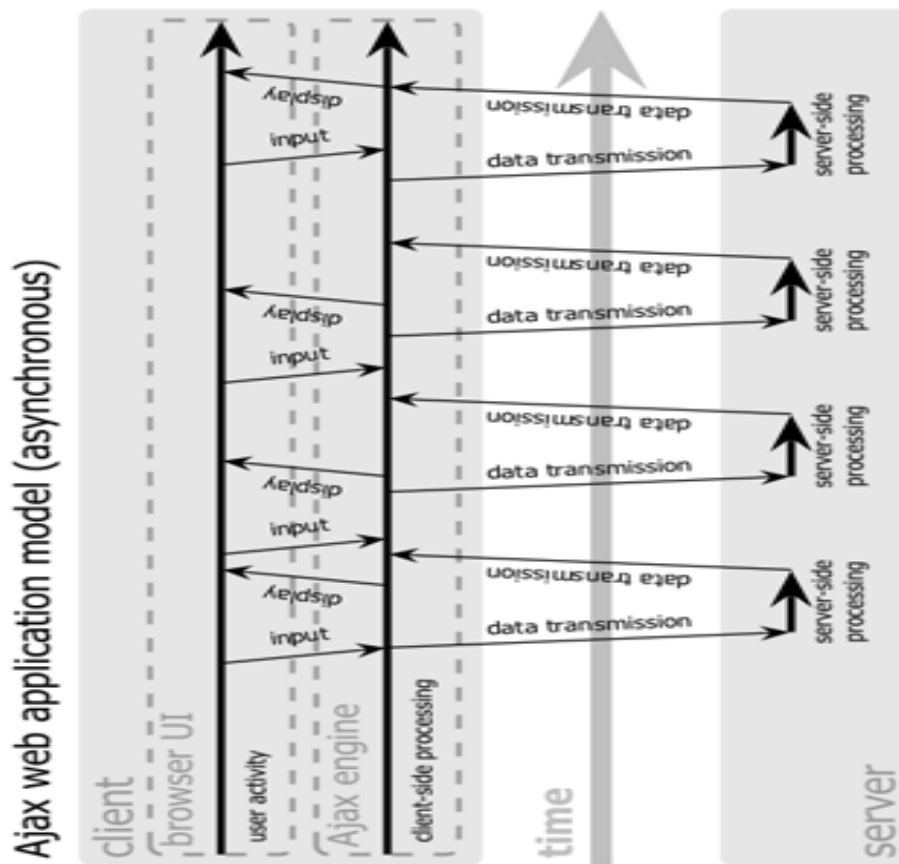
텍스트 기반의 구조화된 데이터

# AJAX 동작원리

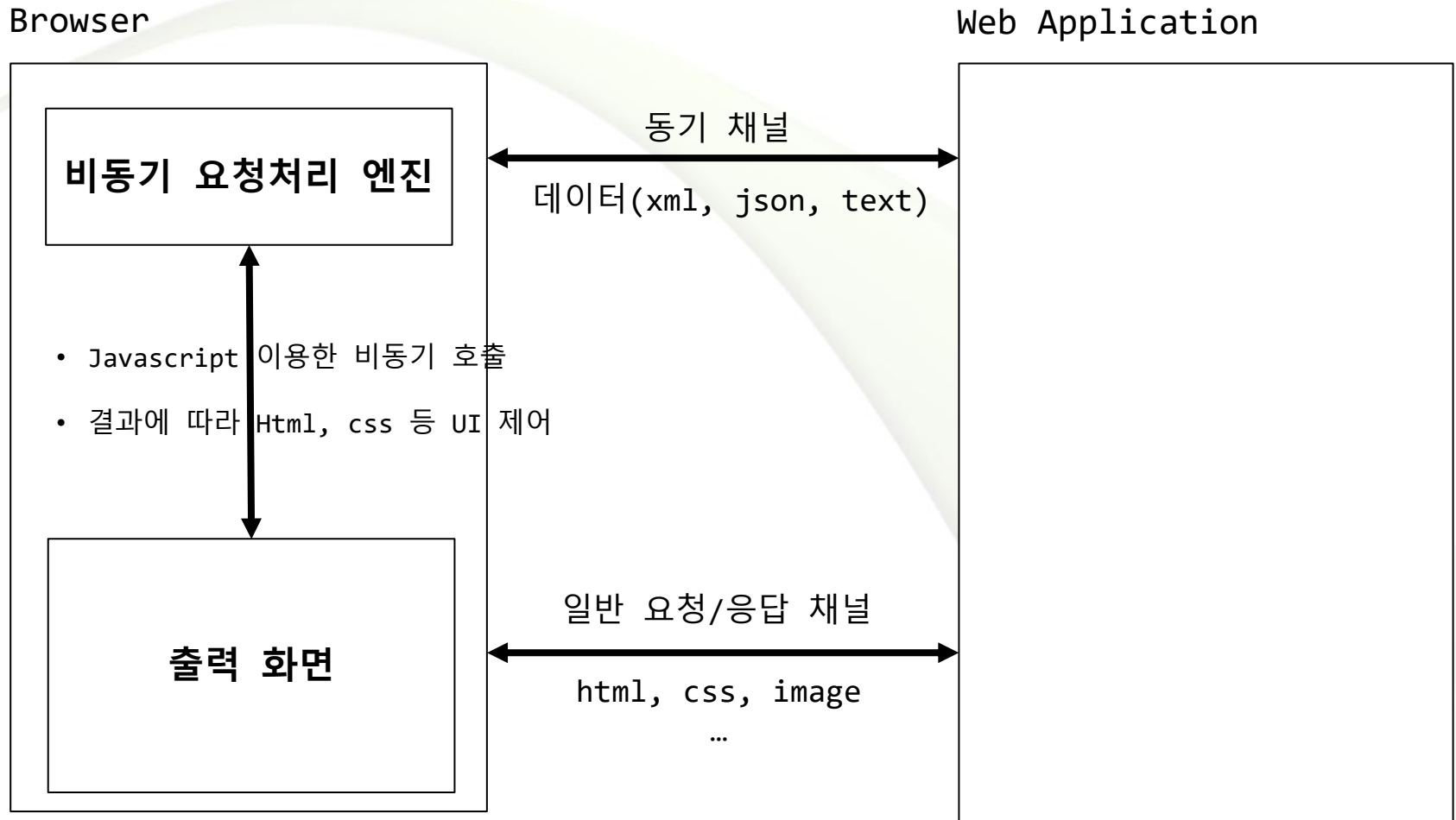
## 동기 방식



## 비동기 방식

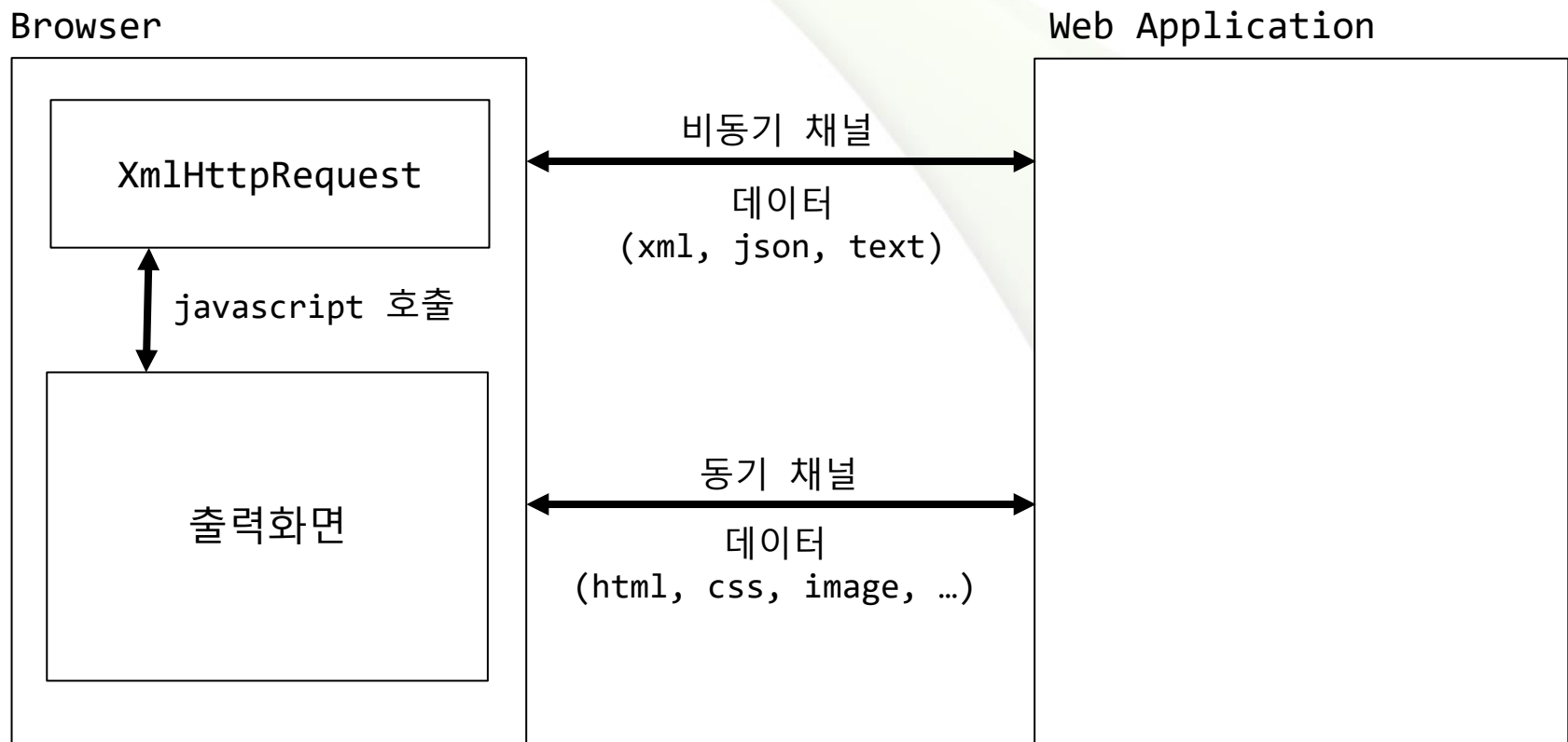


# Ajax 동작 원리 2



# 비동기 통신 기법 (XmlHttpRequest 객체 이용)

- 브라우저 내장 객체 또는 ActiveX 형식으로 제공되는 비동기 요청 객체 활용
- XML, JSON, Text 응답을 스크립트로 처리



# 문제점 및 대안

- 저수준 스크립트 기반 처리
  - » 구현의 복잡성
  - » jQuery와 같은 다양한 스크립트 라이브러리가 개발됨
- HTML의 한계
  - » 고급 그래픽 처리 및 사용자 편의성 한계

# Ajax 구현

## ▪ 저수준 javascript 사용

```
<div id="container">
  <div id="message"><%= new Date().toString() %></div>
</div>
</body>

<script src="https://code.jquery.com/jquery-3.4.1.js"></script>
<script type="text/javascript">
$(function() {

  //비동기 요청 처리 엔진 만들기
  var proxy = new XMLHttpRequest();
  $('#async-request').on('click', function(event) {

    proxy.open("GET", "get-time", true); // (method, request-path, async)
    proxy.onreadystatechange = handleResult; // 비동기 요청 수행 중에 호출할 함수
    proxy.send(null); //비동기 요청 시작

  });

  function handleResult() {
    if (proxy.readyState == 4) { //비동기 요청에 대한 응답이 도착했을 때
      if (proxy.status == 200) { //정상 응답인 경우
        var time = proxy.responseText;
        $('#message').text(time);
      }
    }
  }

});
</script>

</html>
```



# Ajax 구현

## ▪ jQuery 사용

```
<button id="async-request">비동기 방식 요청</button>
<hr>
<div><%= new Date().toString() %></div>
<div id="container">
  <div id="message"><%= new Date().toString() %></div>
</div>
</body>

<script src="https://code.jquery.com/jquery-3.6.0.js"></script>
<script type="text/javascript">
$(function() {

  $('#async-request').on('click', function(event) {

    $.ajax({
      url: "get-time",
      method: "GET",
      async: true,
      success: function(data, status, xhr) {
        $('#message').text(data);
      },
      error: function(xhr, status, err) {
        $('#message').text(err);
      }
    });

  });

});
</script>
```