

Ajax és Echo 2

Bokor Attila
attila.bokor@netvisor.hu



Tartalom

- Ajax áttekintés
- Echo2
- Echo2 kiegészítések



Asynchronous JavaScript and XML

- Webfejlesztési technika interaktív webalkalmazásokhoz
- Kifejezés közösen használt technológiákra
- A kliens kis mennyiségű adatot cserél a szerverrel
 - Interaktivitás, sebesség, használhatóság



Támogató technikák

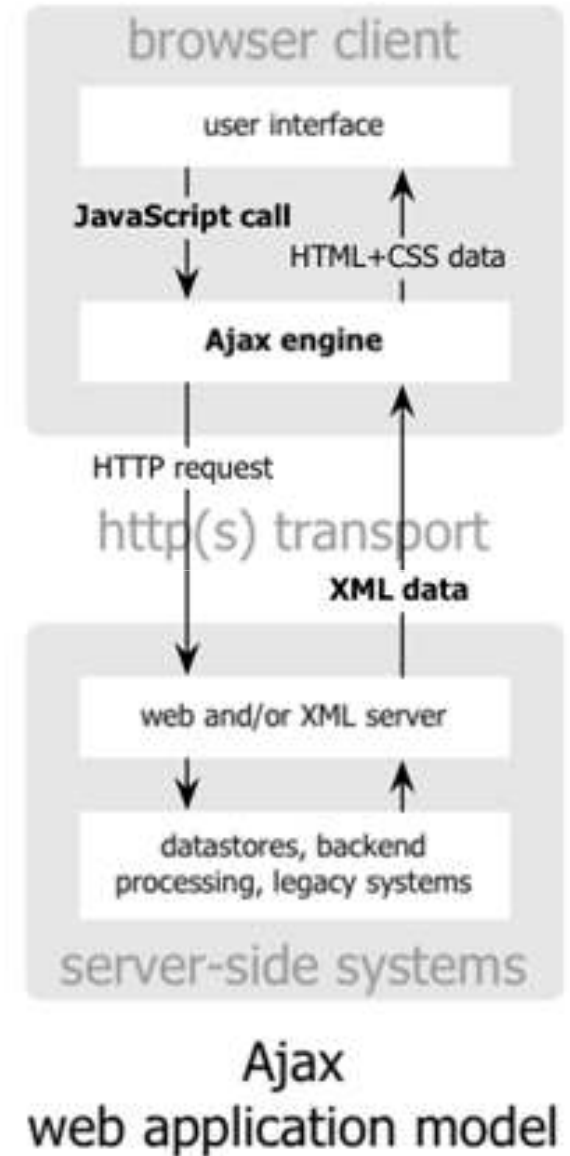
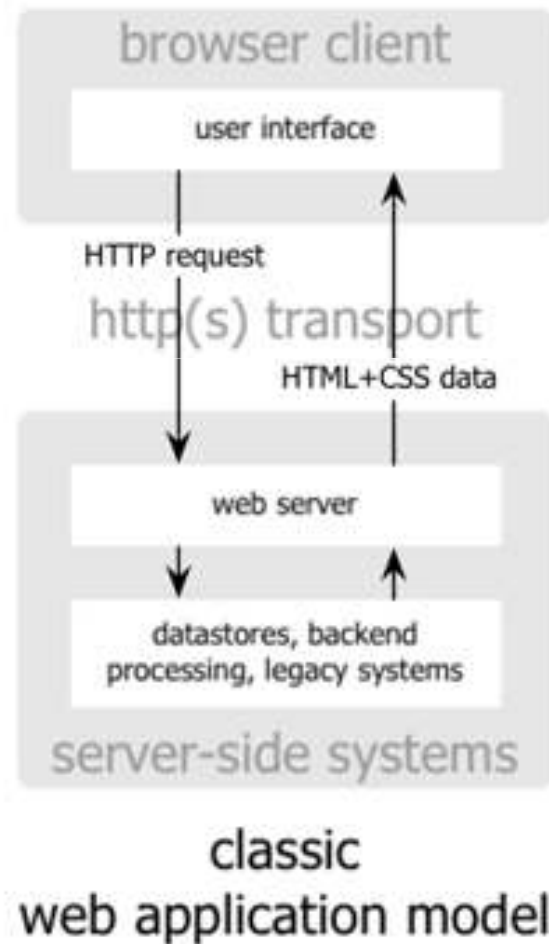
- **XHTML** (vagy **HTML**) és **CSS** a tartalom leírására és formázására.
- **DOM** kliens oldali script nyelvekkel kezelve a dinamikus megjelenítés és a már megjelenített információ együttműködésének kialakítására.
- **XMLHttpRequest** JavaScript objektum az adatok aszinkron kezelésére a kliens és a webservert között. Néhány Ajax keretrendszer esetén és bizonyos helyzetekben **IFrame** -et használnak XMLHttpRequest objektum helyett.
- **XML** formátumot használnak legtöbbször az adattovábbításra a kliens és a szerver között, bár más formátumok is megfelelnek a célnak, mint a formázott HTML vagy a sima szöveg.



Történelem

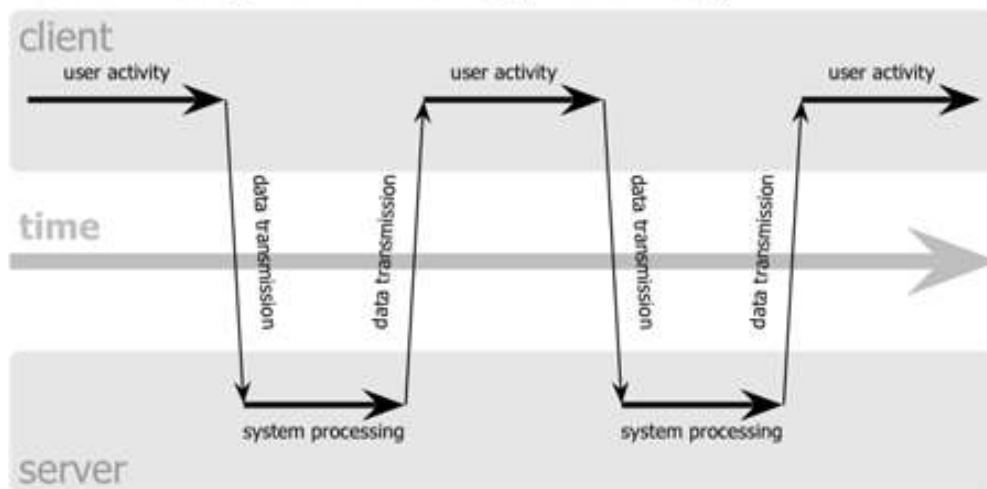
- Netscape LiveScript
- JavaScript
- XML, renderelés JavaScript –ből
- XMLHttpRequest (IE5)
- 2005. Jesse James Garret:
Ajax: A New Approach to Web Applications

Ajax- és hagyományos web alkalmazás architektúra



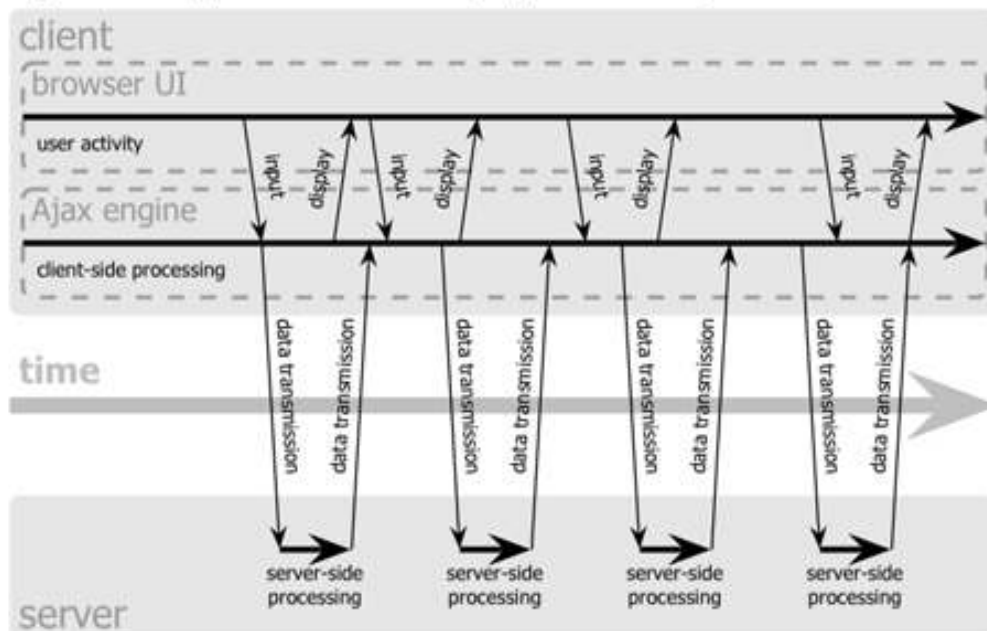
- A felhasználói tevékenység eredménye egy HTTP kérés.
- A felhasználó nem tudja használni a weblapot, míg a teljes válasz meg nem érkezik.

classic web application model (synchronous)



- A szokásos HTTP kérések helyett, többnyire JavaScript hívás.
- Aszinkron kommunikáció

Ajax web application model (asynchronous)





XMLHttpRequest

- <http://www.w3.org/TR/XMLHttpRequest/>
- Állapotok
 - UNSENT, OPEN, SENT, LOADING, DONE
 - status, statusText
- Request
 - open(..., url, ...), send(data), abort();
- Response
 - header, responseXML, responseText
- Eseménykezelő
 - onreadystatechange



DOM fa módosítása

- <http://www.slimeland.com/content/articles/jenhance/>
- példa



Ajax alkalmazások

- <http://www.orkut.com/>
- <http://groups-beta.google.com/>
- <http://www.gmail.com/>

- <http://search.live.com/images>

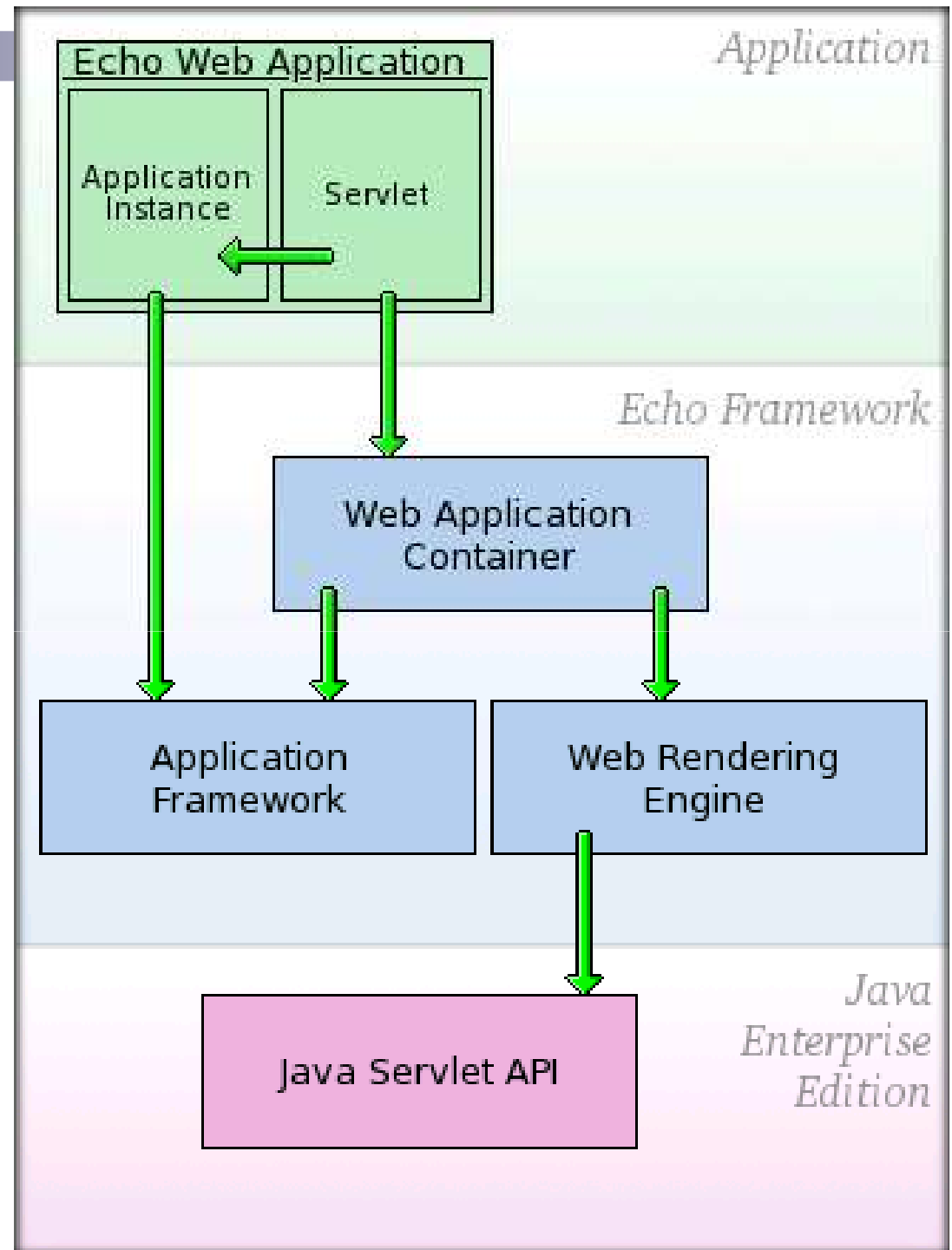


Echo 2 (Nextapp)

- Előzmény: Echo Web Framework
- Ajax alapú keretrendszer webalkalmazásokhoz
- Nem „oldal-alapú” fejlesztés
- Java nyelvű fejlesztés
- Objektorientál szemlélet
- Eseményvezérelt
- Tetszőleges servlet container –ben fut
- GNU LGPL licenz

Modulok

- Minimális függés
- AF – GUI modellezése
- WRE – megjelenítés (kliens- és szerver oldal)
- WAC – WRE kiterjesztése az AF megjelenítéséhez





Application Framework

- Alkalmazás állapotának és a GUI –nak a kezelése
- Fejlesztés során a meghatározó modul:
 - GUI komponensek
 - Események, figyelők
 - ApplicationInstance
 - Nem képes a böngészővel kommunikálni, vagy bármit megjeleníteni



Web Rendering Engine

- Kliens – szerver kommunikáció
- Szerver oldalon
 - HttpServlet (Java) szolgálja ki a klienst
- Kliens oldalon
 - JavaScript
 - XMLHttpRequest objektum használata
 - Ha szükséges, kapcsolódik a szerverhez



Web Application Container

- Az WRE –nek kiterjesztése
- Képes az AF –fel épített GUI –k megjelenítésére
- Szintén szerver- és kliens oldali



Működés

- Alkalmazás állapota a szerver oldalon
- A böngésző, mint távoli terminál
- Kliens oldal „folyamatosan” frissül
- Fontos események jelentése a szervernek
- Változások érvényesítése a Document Object Model (DOM) – en keresztül (sosem tölti újra az oldalt)



Folyamat

- Felhasználói tevékenységekről a server XML üzenetben értesül (ClientMessage)
- Szerver feldolgozza az üzenetet, megváltoztatja a szerver-oldali állapotot
- Szerver XML üzenetben válaszol a kliens felé
- A kliens az üzenetnek megfelelően módosítja a DOM –ot



Példa a kommunikációra

```
<server-message>
```

```
...
```

```
<message-part-group id="remove">
```

```
<message-part processor="EchoDomUpdate.MessageProcessor">
```

```
<dom-remove target-id="c_4"/>
```

```
..
```

```
</message-part>
```

```
</message-part-group>
```

```
...
```

```
<message-part-group id="update">
```

```
<message-part processor="EchoDomUpdate.MessageProcessor">
```

```
<dom-add parent-id="c_2_cell_c_4">
```

```
<content><span xmlns="http://www.w3.org/1999/xhtml" id="c_4">
```

```
Your guess, 75 was too high. Try again: </span></content>
```

```
</dom-add>
```

```
...
```

```
</message-part>
```

```
...
```

```
</message-part-group>
```

```
</server-message>
```



Hello, World!

```
public class HelloWorldApp extends ApplicationInstance
{
    public Window init()
    {
        Window window = new Window();
        ContentPane contentPane = new ContentPane();
        window.setContent(contentPane);
        Label label = new Label("Hello, world!");
        contentPane.add(label);
        return window;
    }
}
```



Komponenstípusok

- I/O és vezérlőelemek

- Címke, gomb, szövegmező, szövegdoz, checkbox, stb

- Konténerek

- Pane, sor, oszlop, grid,
- Layout
- Szülő-gyerek kapcsolatok helyességének run-time ellenőrzése



Komponensek

- **Component**

- AbstractButton** - minden gomb őosztálya
- AbstractListComponent** - listák őosztálya
- TextComponent** - szövegmező és szövegdoboz
- Window** - legfelsőbb szintű ablak
- Label** - címke
- Table** - táblázat
- ContentPane** - innen konténerek
- Grid**
- Row**
- Column**
- SplitPane**
- WindowPane**
- ...



Layout

- Egy elem elhelyezkedésének meghatározása a konténer celláján belül
- A különböző konténerekhez egyedi layout osztály. (közös ős: `LayoutData`)
- Igazítás, margó, háttérszín, stb.



Eseménykezelés

- Esemény objektumok

- `java.util.EventObject` leszármazottjai
- Külön eseményobjektumok az egyes esemény típusokhoz

- Figyelők

- `java.util.EventListener` leszármazottjai
- Minden esemény típushoz megfelelő figyelő típus



További képességek

- MessageBundle
 - Üzenetek tárolása message fájlban
- Stíluslapok alkalmazás a komponenseknél

EchoStudio

The screenshot displays the EchoStudio IDE interface for a Java Swing application named "LoginScreen.java". The design view shows a login form with the "NextApp" logo, "Login:" and "Password:" labels, text input fields, and a "Login" button. A blue circle with the number "3" is positioned above the form.

The component tree on the left shows the following structure:

- ContentPane
 - Grid (1 column)
 - Grid\$Cell (auto)
 - Grid (2 columns)
 - Grid\$Cell (auto) [2x1] (highlighted with a red circle "1")
 - Label
 - Grid\$Cell (auto) [2x1]
 - Grid\$Cell (auto)
 - Label "Login:"
 - Grid\$Cell (auto)
 - TextField
 - Grid\$Cell (auto)
 - Label "Password:"
 - Grid\$Cell (auto)

The Properties window at the bottom left shows the following table:

Property	Value
ID	cell1
Visibility	None
background	
backgroundI...	
columnSpan	2
enabled	
flow	2
flowPriority	
focused	
font	
foreground	
horizontalAlig...	Center
identifier	
insets	



EchoPoint NG

- Kiterjeszti az Echo2 komponensmodelljét
- Fejlettebb grafikus komponensek
 - Lapozható táblázat, tabbed pane, naptár
- Nem hivatalos kiterjesztés
- CVS –ben, forrásban elérhető!



Echo Extras

- A hivatalos Echo2 kiterjesztés
- Echopoint NG –vel hasonló funkcionalitás
 - AccordionPane, BorderPane, CalendarSelect, ColorSelect, PulldownMenu, TabPane, TransitionPane



Referenciák

- <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- <http://www.nextapp.com/platform/echo2/echo/>
- <http://nextapp.com/platform/echo1/echostudio/>