COMET webalkalmazás fejlesztés

Tóth Ádám

Jasmin Media Group

Az előadás tartalmából

- Alapproblémák, fundamentális kérdések
- Az eseményvezérelt architektúra alapjai
- HTTP-streaming megoldások
  - AJAX Polling
  - COMET
    - AJAX COMET
    - Long/Smart Poll
    - Forever Frame

Alapproblémák

- A HTTP protokoll kérés-válasz jellegű kommunikációt valósít meg
  - A kommunikációt a kliens kezdeményezi, a szerver erre válaszol
  - A szerver által kezdeményezett kommunikáció nem támogatott
- Webalkalmazás: 3 rétegű alkalmazás
  - Megjelenítési réteg (Böngésző)
  - Üzleti logika (Webszerver + cgi)
  - Adat szint (Adatbázis)

Alapproblémák

- Gond:
  - A webalkalmazás állapota nemcsak a kliensoldalon változhat meg,hanem szerveroldalon is
  - Hogyan értesül a kliens oldal a szerver oldalon történt változásokról?
    - A szerver nem kezdeményezhet kommunikációt
    - A kliensnek le kell kérdeznie az aktuális állapotot
- Cél: szerveroldali események eljuttatása a kliensnek, hatékonyan, oldal újratöltés nélkül

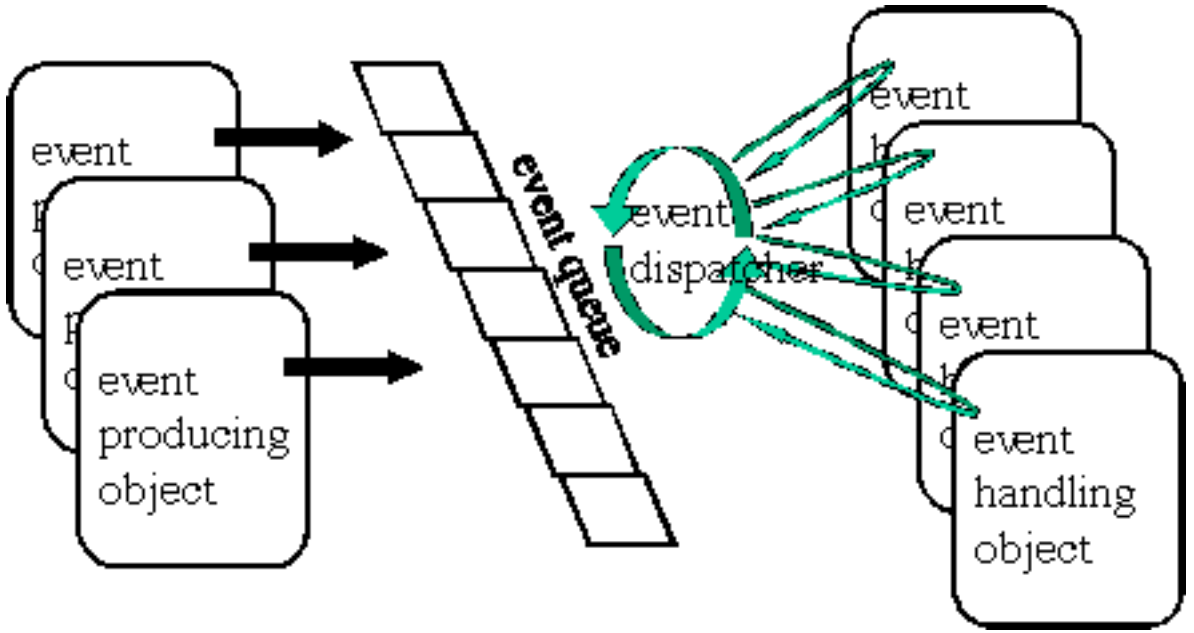
Eseményvezérelt architektúra

- Programozási paradigma; a program futásának menete nem mereven rögzített, azt különböző események (felhasználói, szoftver) befolyásolhatják
- Kliens oldalon biztosított (OS, Javascript), szerveroldalon azonban nem, így célszerű kiépíteni

Eseményvezérelt architektúra

- Az architektúra komponensei
  - Esemény (event)
  - Eseménysor (event queue)
  - Eseményciklus (event loop)
  - Eseménykezelő (event handler)

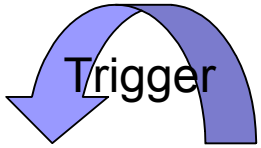
Eseményvezérelt architektúra



Eseményvezérelt architektúra

- Esemény-infrastruktúra: az események többsége az adatbázis tartalom változásával kapcsolatos => tartsuk az adatbázisban az eseménysort

Eseményvezérelt architektúra



eventID	Source	sourceID	Type	Time
1	Session	213231	I	1174 9366 19
2	Session	213231	U	1174 9366 120
3	Publicat ion	942	U	1174 9366 121

sessID	sessData	…
213231	…	…
…	…	…

Eseményvezérelt architektúra

- Eseményciklus: új események kivétele a sorból és eljuttatása az eseménykezelőhöz
  - Webalkalmazás: eseménykezelő a kliens oldalon
  - Adatátvitel: valamelyik HTTP-Streaming technika segítségével

HTTP-Streaming megoldások AJAX-Polling

- Talán a legkézenfekvőbb megoldás
- Használjuk az XMLHTTPRequest (XHR) objektumot az eseménysor periodikus lekérdezésére!
- Események átvitele pl.: XML formátumban
- DEMO

HTTP-Streaming megoldások AJAX-Polling

- Előnyök
  - Könnyen megvalósítható
  - Kis trükkökkel jól skálázható
  - Hordozható
- Hátrányok
  - Ha az események gyakoriak, csökkenteni kell a lekérdezés periódusidejét => sok kapcsolat
  - Ha az események változó gyakorisággal jelennek meg => erőforrásokat pazarlunk el felesleges lekérdezésekkel
  - Nem túl rugalmas megoldás

HTTP-Streaming megoldások COMET

- Küszöböljük ki a polling hibáit!
- Sok lekérdezés helyett nyissunk egy kapcsolatot a szerver felé, és tartsuk azt nyitva (keep-alive)!
- A nyitott kapcsolaton keresztül a szerver folyamatosan küldhet adatokat a kliensnek
- Adatforgalom csak új esemény esetén

HTTP-Streaming megoldások
COMET - Problémák

- Gondok a hosszú HTTP kapcsolatokkal
  - Webszerver
    - Rossz skálázhatóság
  - Tűzfalak
    - Hosszú kapcsolatok bontása
  - Böngészők
    - 1 szerverhez max. 2 párhuzamos kapcsolat
    - Kapcsolat bontása, ha bizonyos ideig nem jön adat

HTTP-Streaming megoldások
Ajax COMET

- Kapcsolat létesítése az XHR objektumon keresztül
- Események küldése a szerveroldalról
  - XML?
    - Nem jó, mivel a responseXML csak akkor olvasható ki, ha a teljes XML dokumentum rendelkezésre áll
  - Text
    - Érdemes azonnal végrehajtható JavaScript kódot küldeni
    - JSON praktikus választás

HTTP-Streaming megoldások Ajax COMET

- Események fogadása
  - Mikor?
    - Nyitott kapcsolat esetén az XHR readyState attribútuma a 3-as (interaktív) értéket veszi fel
      => az onReadyStateChange callback-ben a 3-as állapotot kell figyelnünk
  - Hogyan?
    - A responseText változó folyamatosan „hízik”, ahogy az új események érkeznek
      - Tárolni kell, hogy a callback utolsó meghívásakor hol fejeztük be a responseText olvasását
      - Az eseményeket valamilyen határoló karaktersorozattal kell elválasztanunk egymástól
- DEMO

HTTP-Streaming megoldások
Ajax COMET

- Problémák
  - Internet Explorer alatt nem működik
    - Az IE XHR implementációja nem engedi kiolvasni a responseText tartalmát, amíg a readyState nem 4 (Complete)

HTTP-Streaming megoldások Long/Smart Poll

- Próbáljuk meg egyesíteni az AJAX Polling és az AJAX COMET pozitív tulajdonságait!
  - AJAX Polling: hordozható, egyszerű
  - AJAX COMET: hatékonyabb kapcsolat kihasználás
- Ötlet:
  - Ha vannak új események, küldjük el őket a kliensnek, zárjuk le a kapcsolatot, majd indítsuk újra a stream-et
  - Ha nincs új esemény, nem térünk vissza, nyitva tartjuk a kapcsolatot
- DEMO

HTTP-Streaming megoldások Long/Smart Poll

- Előnyök
  - Minden böngésző alatt működik
  - Takarékosabban bánik új kapcsolatok kiépítésével, mint az AJAX Polling
- Hátrányok
  - Ha gyakran következnek be események, ugyanott vagyunk, mint az AJAX Polling esetében

HTTP-Streaming megoldások
Forever Frame

- A Gmail-ben debütált
- Alapötlet: minden böngésző automatikusan végrehajtja a frame-ekbe ágyazott inline javascript kódokat
- => Az XHR objektum helyett használjunk iframe-et a streaming lebonyolítására
- A szerveroldali script inline javascript-et küld az iframe-nek, melyet a böngésző automatikusan végrehajt
- DEMO

HTTP-Streaming megoldások
Forever Frame

- Előnyök
  - Minden böngésző alatt működik
  - Megvalósítja a COMET működést
- Hátrányok
  - A hosszú HTTP kapcsolatok során felmerülő problémák itt is érvényesek!

Összefoglalás

- HTTP-Streaming – léteznek megoldások
  - A két legrugalmasabb a Long Poll és a Forever Frame
  - A különböző időzítési paraméterek megválasztása nagyon fontos!
- Kész megoldások
  - Light Streamer
  - COMET and AJAX Request Router
  - COMETd
  - XAJAX
  - JQuery