Architecting Public Facing Website Software For High Concurrent User Load

Derrick Lau

My experiences

- In 2011 I first experienced degraded performance of public facing websites.
- Multiple online outages with a RESTful architected application.
- Ongoing recurrences even up to Feb 2015.
- Contracted load testing vendor: identified bandwidth as bottleneck.

Some famous website performance issues

- http://www.yottaa.com/blog/bid/265815/Coke-SodaStream-the-13-Websites-That-Crashed-During-Super-Bowl-2013
- http://www.thestar.com/news/gta/2015/02/03/go-transit-pledges-better-service-following-mondays-snow-snarled-commute.html

Some famous website performance issues

- Some Twitter complaints:
- https://canadianoutages.com/status/go-transit
- **@LorrieACooke** @gotransit website not working. "server too busy"
- 2015-02-02 07:02:41
- **@Chuck_02** @gotransit how long is the delay?! your site is down
- 2015-02-02 06:58:41
- **@Hockey_dokey** @gotransit service updates are not working now. a message "server is busy..." is being displayed. please check!
- 2015-02-02 06:41:22
- **@kukklapr** @gotransit whats going on with the service updates?? lots of delays and no emails?
- 2015-02-03 07:54:27
- **@KimWosnick** @gotransit guessing your service updates are still not working.. just "updated" yet been like this all am. #freezing http://t.co/hdzsydaatc
- 2015-02-03 07:22:17
- **@NymAbid** @gotransit your service update page is not working since yesterday http://t.co/iuyurrizcl
- 2015-02-03 09:12:09

Dan Kegel’s Problem: C10K

- In 1999 the problem was to handle 10000 simultaneous clients with a single web server.
- In the 21st century, we are now calling the problem the C1million problem.
- We will focus on architecture of custom code for public websites.

What are the key things to consider in website architecture?

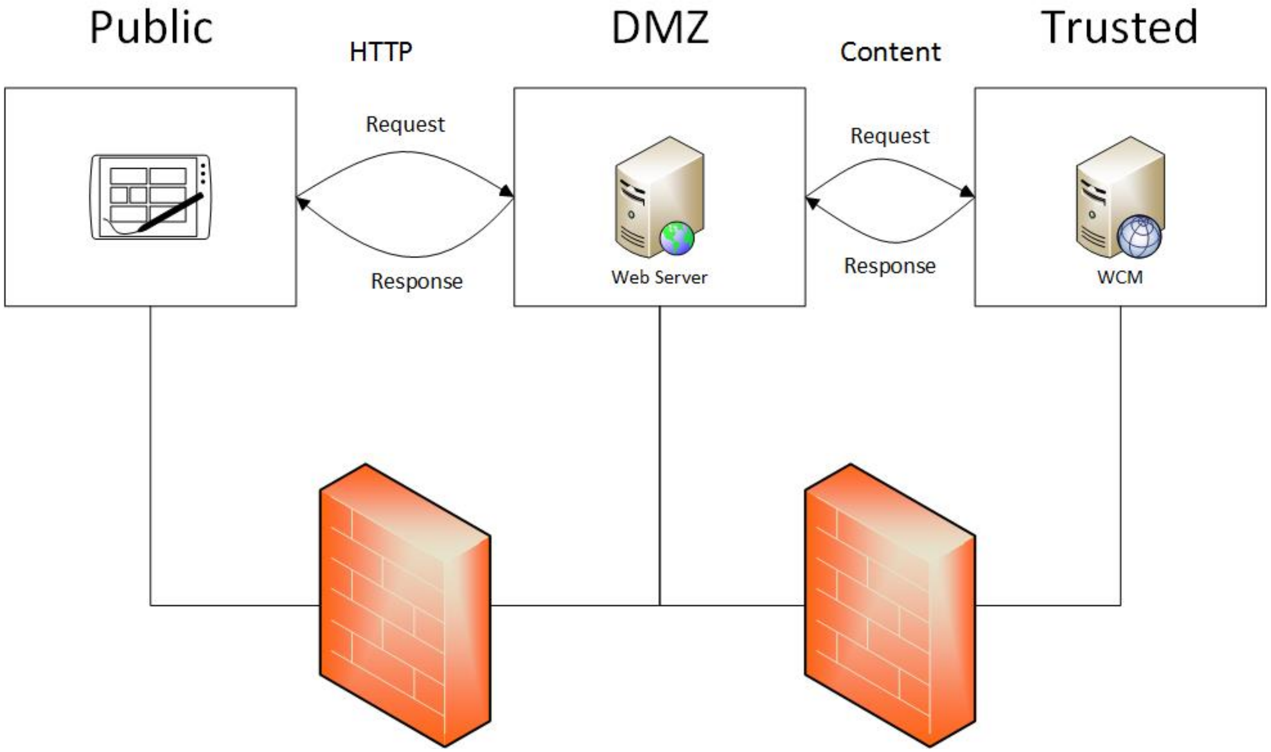
- Web content management (WCM)
- Application architecture (software architecture of online features)
- Security
- Infrastructure/Hosting

Web Content Management (WCM)

- Separation of website content from functionality.
- Better as part of a larger ECM (Enterprise Content Management) solution.
- Focus on content delivery architecture: page-based vs **decoupled**.

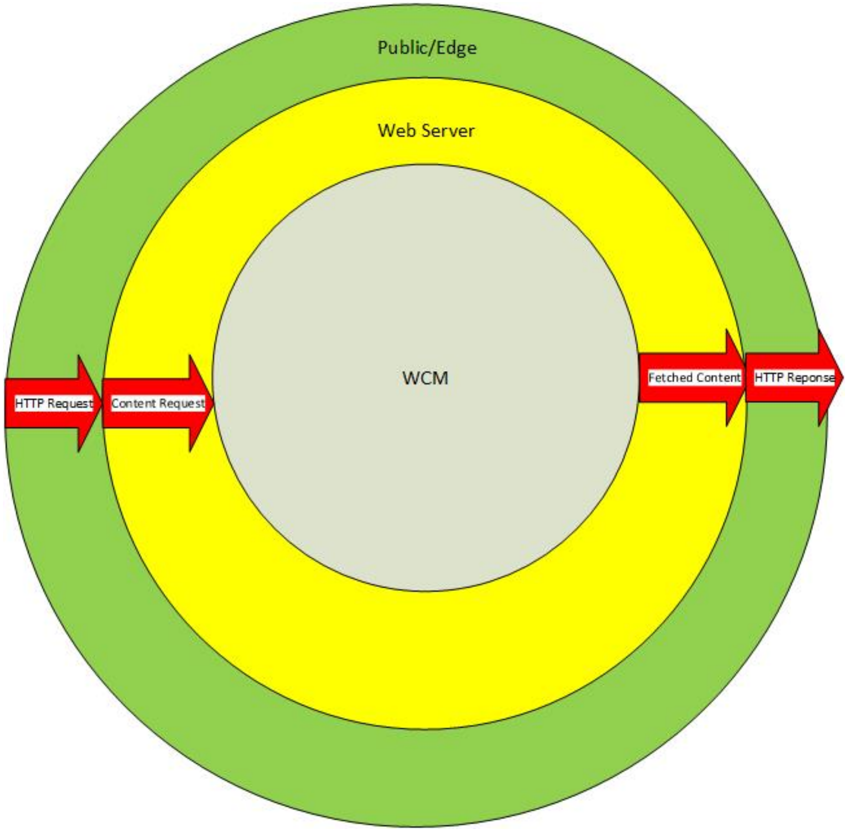
Web Content Management (WCM)

- Example of page based content delivery architecture:



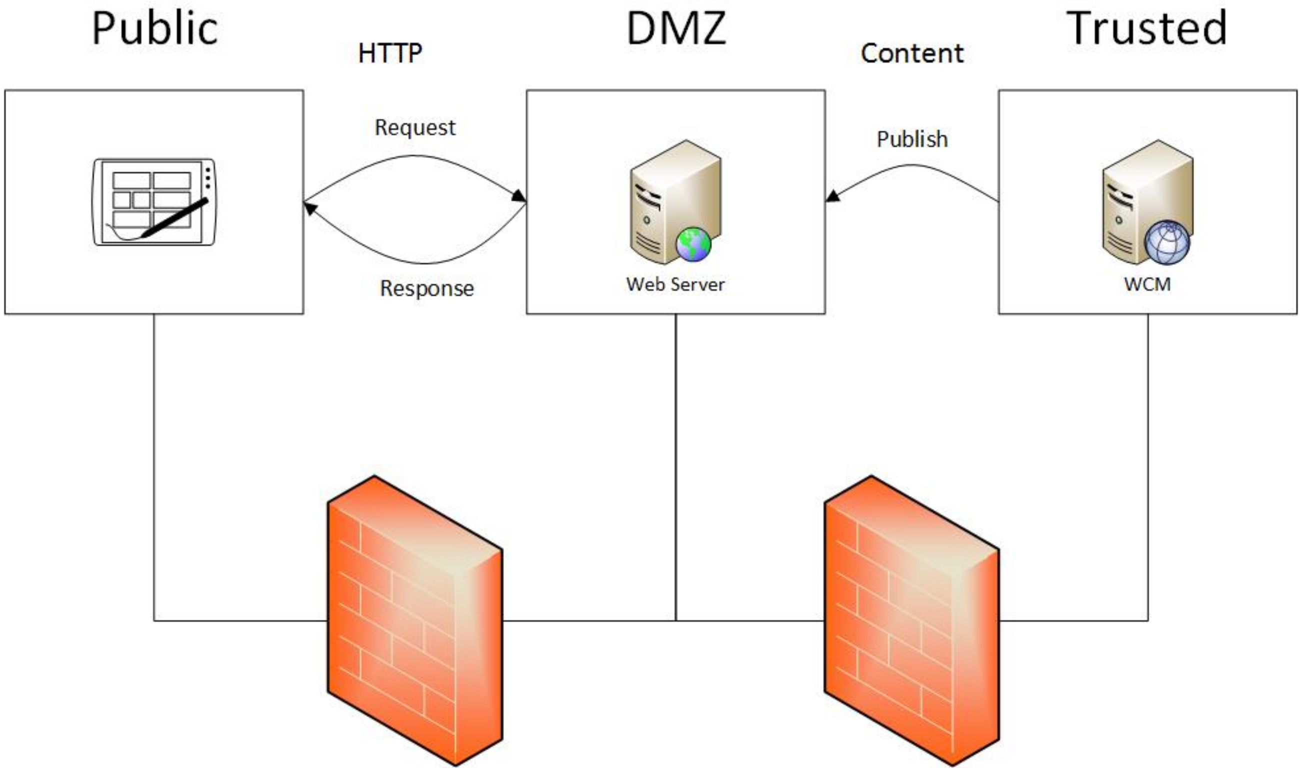
Web Content Management (WCM)

- Example of page based content delivery architecture:



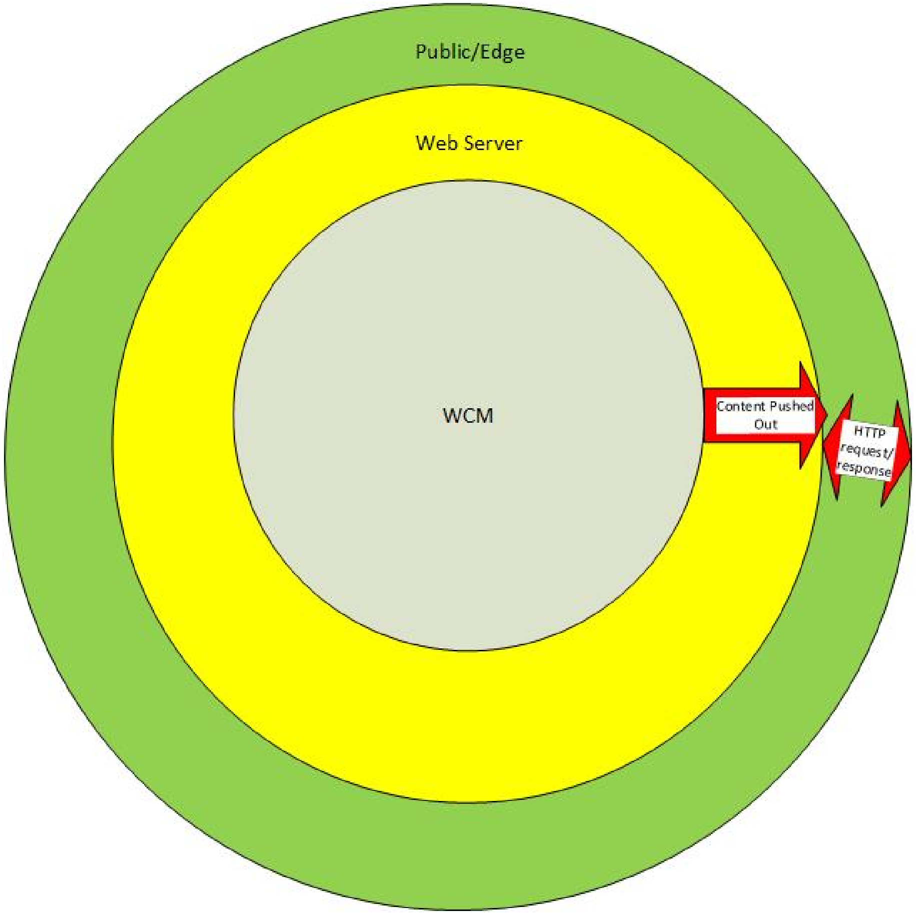
Web Content Management (WCM)

- Example of decoupled content delivery architecture:



Web Content Management (WCM)

- Example of decoupled content delivery architecture:



Web Content Management (WCM)

- Currently I know of 3 decoupled WCM solutions:
  - Ingeniux
  - SDL Tridion (EMC partner)
  - EMC Interactive Delivery Services
- Always try to buy WCM as a part of your ECM suite

Architecting online features

- Most popular is REST
- It’s stateless, so each HTTP request/response is a transaction on its own
- But what happens during high concurrent user load?

Architecting online features

- http://rant3000.blogspot.ca/2010/11/why-using-rest-will-kill-your-project.html
- Some limitations of REST are clearly explained.

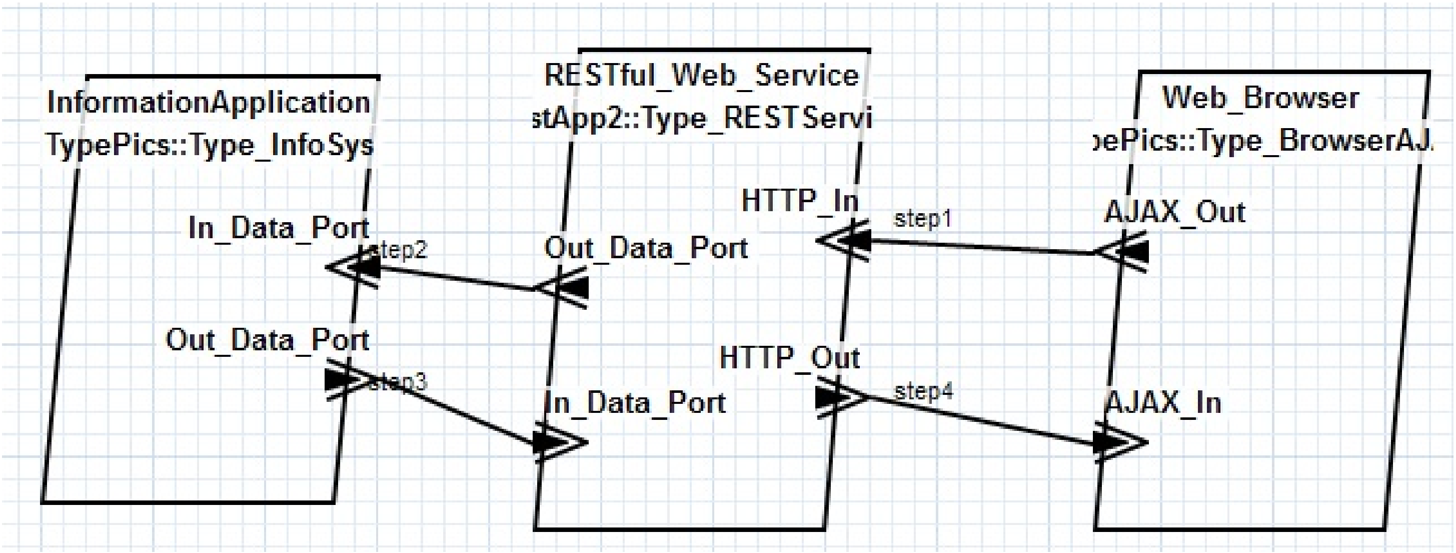
Architecting online features

"You see the root of the problem, I believe, is that REST is a heavyweight protocol. At first when you are designing the APIs in waterfall you try to alleviate this by chunking requests, and having fat methods that return lots of data in one go. In agile you don't notice it at first but when you start to scale you notice how slow everything is and how your traditional optimization strategies don't seem to work so well. Basing an application on REST is like basing an assembly program on interrupts, as the early Macintosh I/O was, like putting every memory access over a relatively glacially paced network bus. The simple act of an HTTP request/response, especially for anything complicated, is quite slow compared to on-box requests. When you start multiplying components, and have REST requests which to complete must call other REST requests in a cascading tree, then the problem becomes quite troublesome."

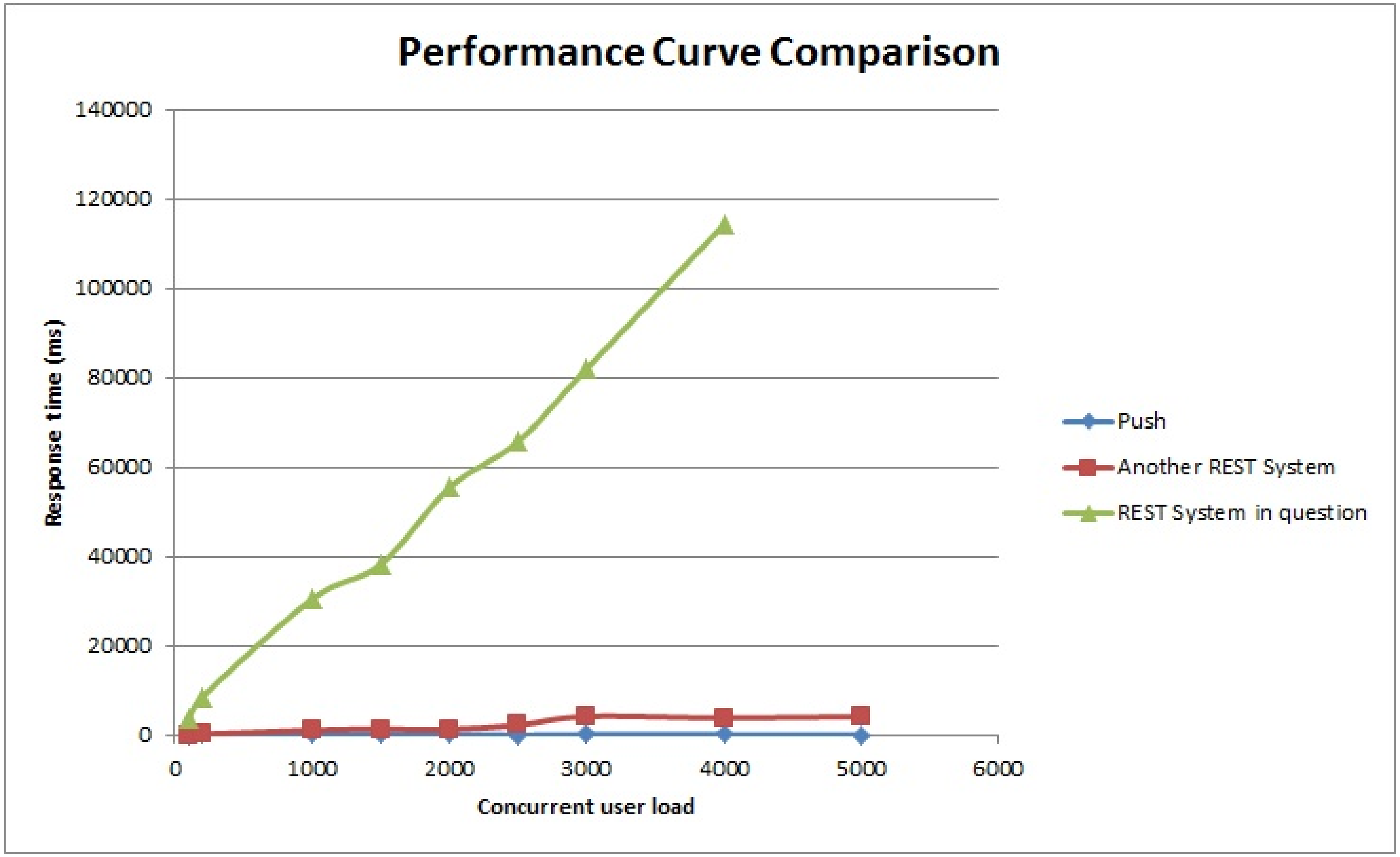
Architecting online features

- Also, the REST architectural style allows each outside connection to be felt by the internal components.
- What happens if you are using JavaScript that sends AJAX calls to a RESTful service frequently to provide up to date info to the public?

Architecting online features



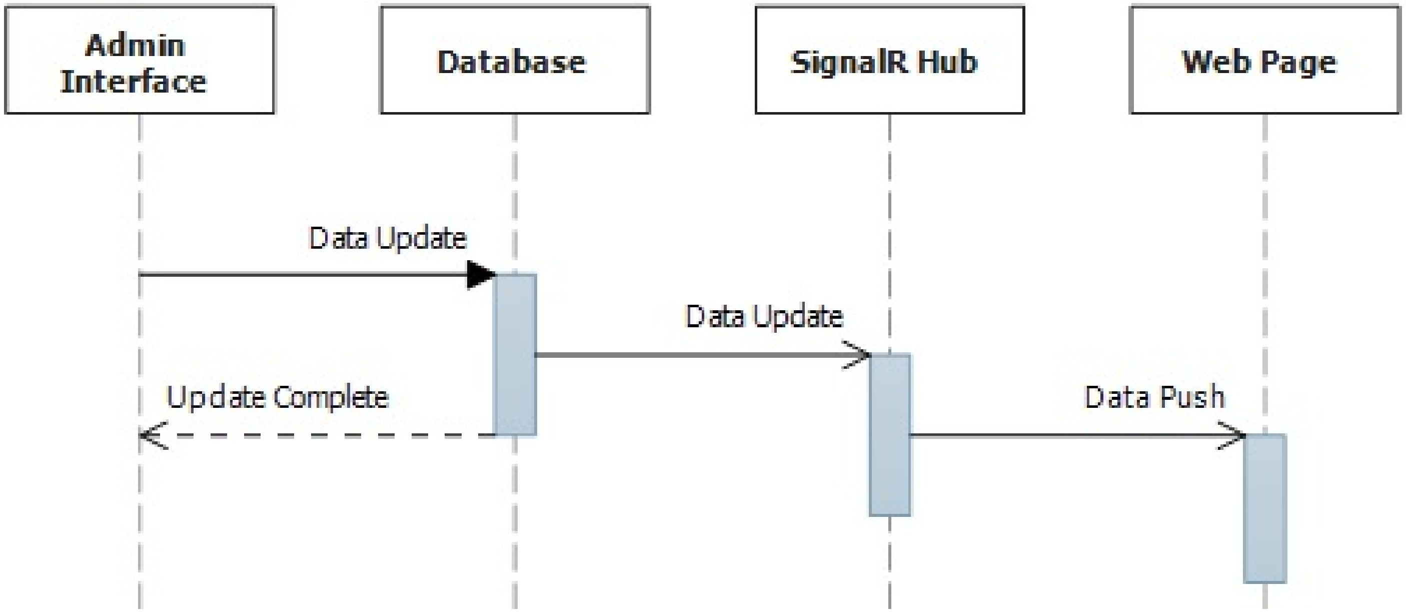
Architecting online features



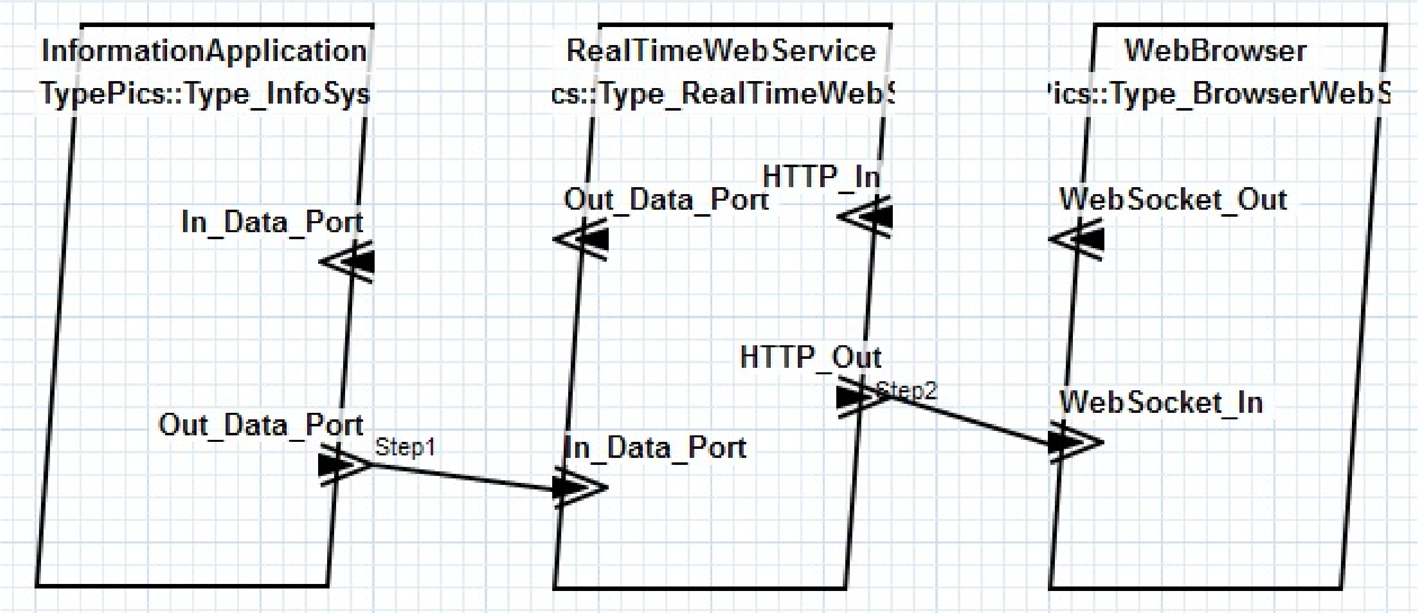
Architecting online features

- As opposed to REST, another approach for such dynamic information features would be real-time web.

Architecting online features



Architecting online features



Architecting online features

- Examples of real-time web in use:
  - Twitter
  - Facebook
  - Google

Architecting online features

- Examples of real-time web frameworks:
  - ASP.NET SignalR
  - CometD – Dojo
  - https://msdn.microsoft.com/en-us/magazine/jj891053.aspx (My MSDN article)
  - Ajax Push Engine (APE)
  - HTML5 Server-Sent Events

Architecting online features

- Or download a web socket server:
  - http://doc.wakanda.org/WebSocket-Server/WebSocket-Server.100-1049244.en.html
  - http://www.tornadoweb.org/en/branch2.1/websocket.html
- Or code your own:
  - https://developer.mozilla.org/en-US/docs/WebSockets/Writing_WebSocket_servers

Security

- Most penetrations happen from within the corporate network.
- May need to place sensitive information in a separate network zone (another trusted zone).
- Ensure trusted zone doesn’t have too many firewalls to pass through if it must send information to the Internet, otherwise performance will be impacted.

Cloud vs on-premise hosting

- Based on previous load tests, bandwidth from the DMZ to the internet can be a bottleneck, especially when pages are graphics intensive.
- CDN vs Cloud?

My recommendations

- Strategy: Edge computing
- Tactics: Use push techniques and keep other systems decoupled from the websites
  - Decoupled content delivery pattern for WCM.
  - Real-time Web communication pattern for info updates/breaking news/etc.
  - Cloud hosting.

Patterns

Decoupled content delivery pattern for WCM.

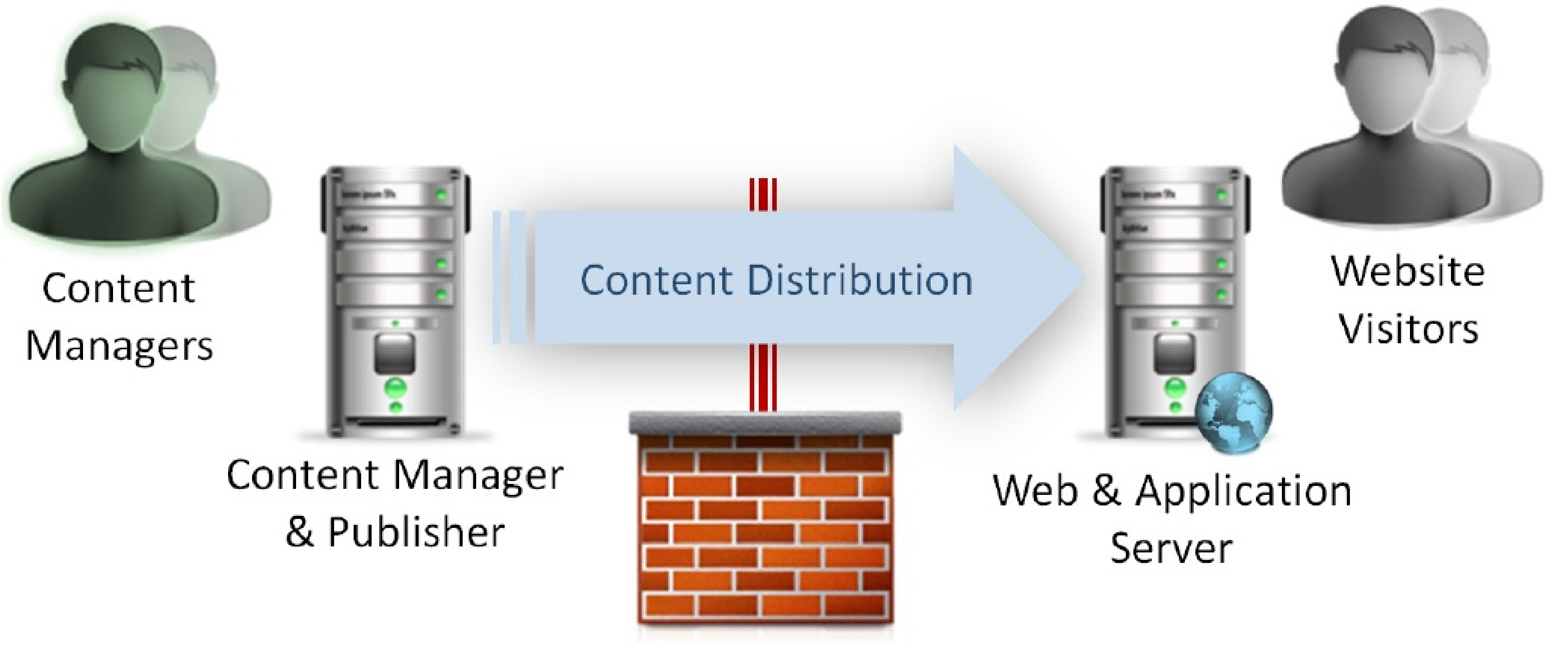
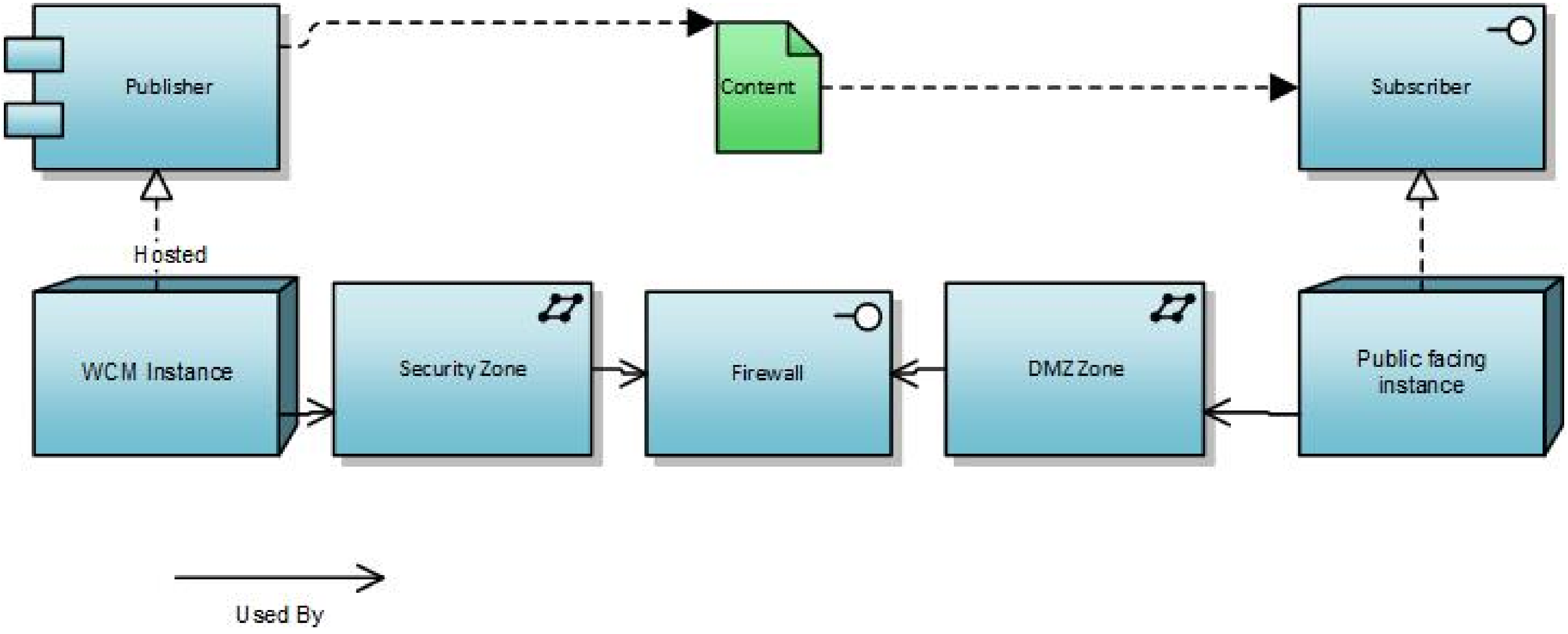


Figure 1: Minimal decoupled infrastructure

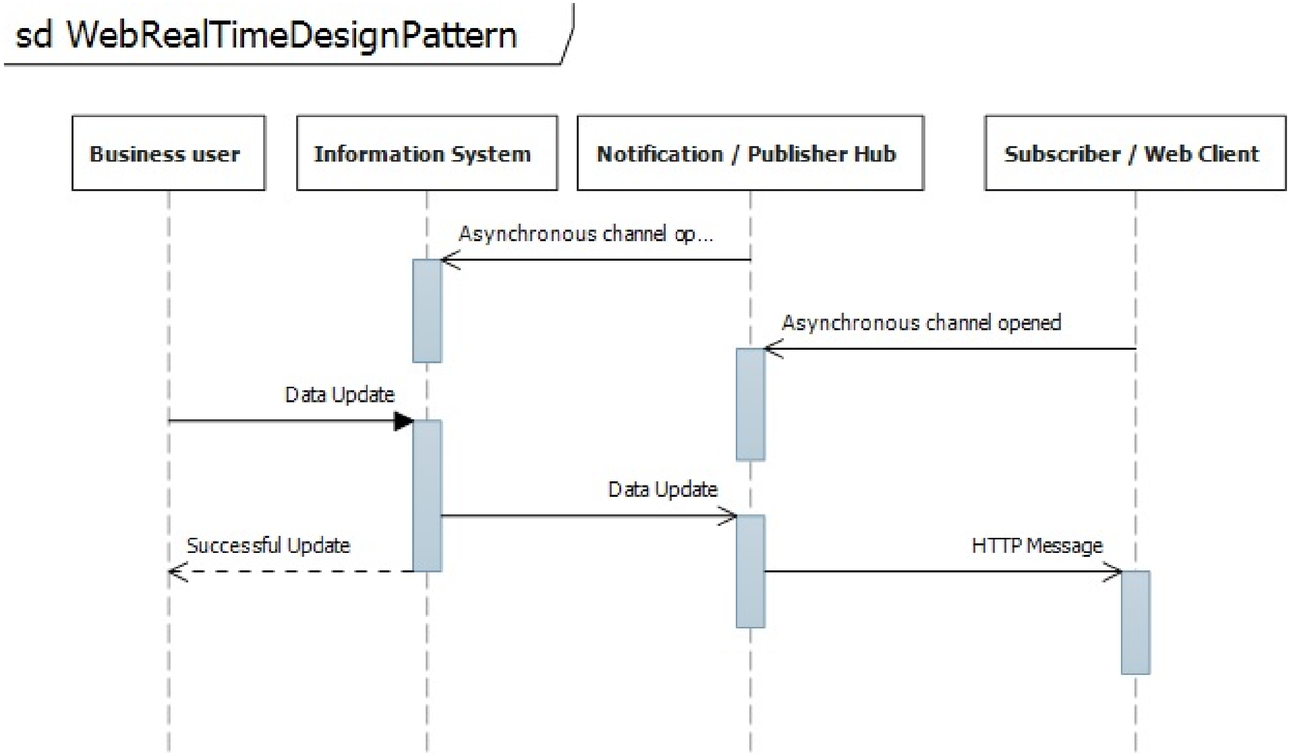
Patterns

Decoupled content delivery pattern for WCM.



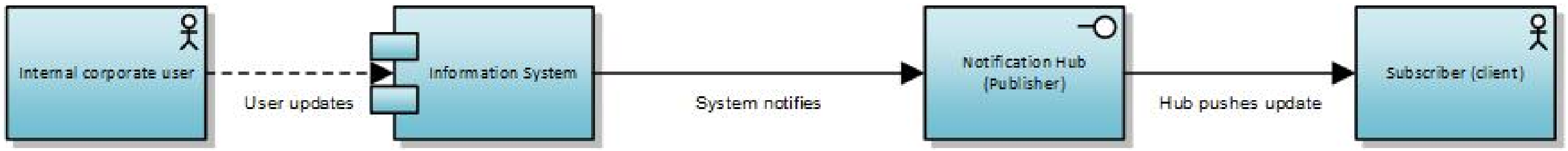
Patterns

Real-time web communication pattern.


sd WebRealTimeDesignPattern
Business user
Information System
Notification / Publisher Hub
Subscriber / Web Client
Asynchronous channel op...
Asynchronous channel opened
Data Update
Data Update
Successful Update
HTTP Message

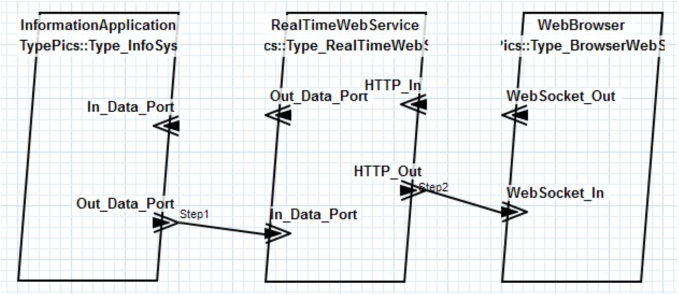
Patterns

Real-time Web communication pattern.



Patterns

Web real-time communication pattern.



Cloud Hosting

- PaaS (much easier to support and maintain).
- Examples:
  - Azure
  - EMC
  - OpenText
  - Amazon

Questions

Acknowledgements

- Andrew Douglas, Ingeniux, SDL Tridion for providing a decoupled content delivery diagram.

My contact info

- Twitter: @Derrick_Lau
- Email: derrick.lau@live.ca