



AD-1380: A Beard, An App, A Blender

Eric McCormick

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Who?

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Who is talking?



- Eric McCormick
- developer
- husband
- father
- veteran
- former EMT
- blogger
- ...more than I have space for



IBMCHAMPION

edm00se.io

[@edm00se](https://twitter.com/edm00se)

Who, continued



Make Every Moment Count

- Web Developer at **BOLDT.** | BEYOND 125 YEARS
 - a fourth-generation, family-owned professional construction services firm, in operation since 1889
- my focus usually revolves around:
 - document control
 - productivity and performance metrics
 - system integration
 - process automation

Who's This Session For, and Why?



Make Every Moment Count

- Developers, who have a passion for
 - application structure, standards, and maintenance
 - documentation
 - testing
 - automation
- Non-Developers (development-adjacent roles)
 - who want a developer's take on the above

What's the Session?

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

AD-1380: A Beard, An App, A Blender...



Make Every Moment Count

- ...one developer's take on building applications with Domino/XPages
- should probably be ...one developer's take on *expanding how we can build applications with Domino/XPages*

Session Abstract



Building applications with Domino/XPages opens a number of doors. Choosing the right path is what becomes hard. This is a session on one developer's take on the way we can structure our applications to get the best of the Domino/XPages platform in addition to all the modern, front-end tooling that the rest of the industry is using to great advantage. This session will cover an approach to app dev via application segregation mechanics, providing the data via HttpServlets to provide a RESTful JSON API, and a UI-layer that automates the majority of concerns via a JS MV framework. Front-end tooling via Yeoman, Grunt, and Bower (and/or others) can aid in our development and testing, back-end mocks for making contracted work easier, and other techniques.*

Ms. America Statement



1. A harmonious development environment for myself and any development staff at my company
2. A unified ~~theory~~ practice of development, for consistent cross-platform and cross-app development behavior
3. Making all application logic and data consistently accessible across separate, but organic, systems
4. Making contracted development work more easily performed
5. Automating everything I can

Make Every Moment Count

Alternative Session Title



Make Every Moment Count



Eric McCormick
and the Quest to Develop
Amazing Applications



My Personal Goals (a.k.a.- my ‘quest’)



Make Every Moment Count

- Use the best tools available to aid my development workflow
 - Automate all tasks I can, for consistency, ease, and speed
 - Enforce coding, documentation, and testing standards at each level of the application
-
- All of which feeds into my ultimate goal of developing the best apps I can, by enabling the developer (that's me!)

I Want It All



Make Every Moment Count

Connect2016

The Premier Social Business and Digital Experience Conference



Session Overview

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Session Content



Make Every Moment Count

- Application Structure
 - Segregated back- and front-ends
 - Services layering concepts (by nature and convention)
- Back-End
 - Beans, Controllers, and M-V-C advantages
 - RESTful API benefits, via HTTPServlets
- Front-End
 - M-V-* and JS frameworks
 - Advanced tooling for better/faster/stronger development

Concerning Application Structure

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Application Structure?



Make Every Moment Count

Segregation of Application Into Service Layers

- a decoupling of the layers of your application into micro-services*
- separates the primary application logic into its own classes
- keeps the UI logic all at the UI level
- keeps styling and presentation out of the logic
- keep all your code readable by not just yourself, but others

Application Structure? (pt.2)



Make Every Moment Count

Application Layers (as I see them)

- data store / DB (NSF)
- primary application classes (server elements for data objects, controllers for behavior, server actions)
- expose application via servlets (RESTful or otherwise), XPages bindings (EL to bean, or data object)
- provides interface to UI layer (where the client-side logic lives)



Connect2016

The Premier Social Business and Digital Experience Conference

Application Structure? (pt.3)



Make Every Moment Count

- we're already starting to talk in terms of what we will require for an application's
 - data
 - primary business logic (which wrapped with the database creates the data service)
 - creating an API, for how a user or front-end can interface with our service
- all of this makes our choice of front-end, be it an XPages app with managed bean or object data source (xe:objectData) or a non-XPages front-end immaterial far less consequential

Advantages and Disadvantages



Advantages

- consistency in an app and across apps
- normalizing of development patterns
- easier to:
 - support/maintain
 - document
 - test

Disadvantages

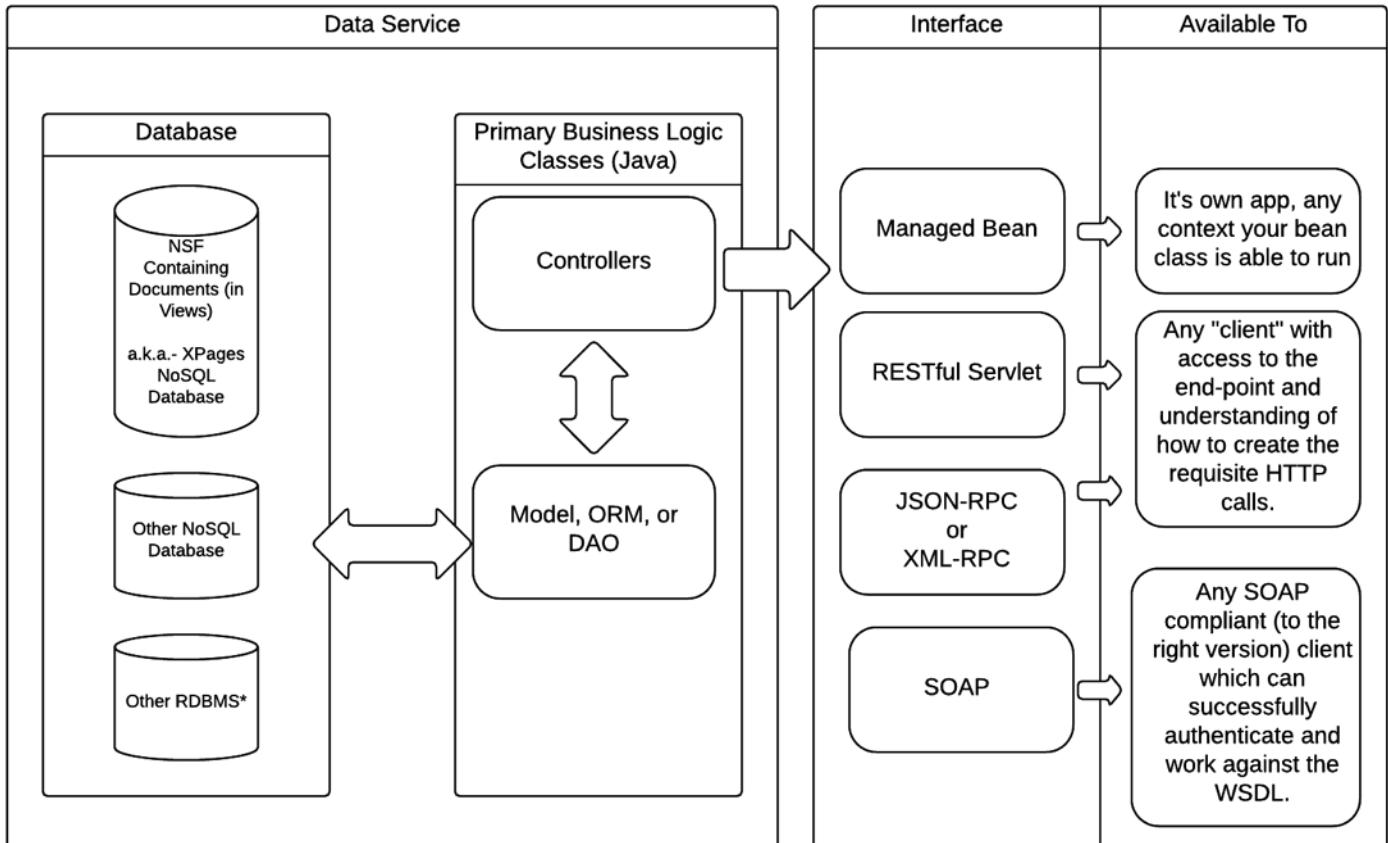
- time
- thought
- willingness to attempt
- management sign-off

Make Every Moment Count

High Level Overview



Make Every Moment Count



Service Layers



- keeps:
 - data
 - logic
 - client interface
- all nice and tidy, this benefits in the forms of:
 - organization
 - maintenance
 - documentation (let's face it, a JavaDoc is better than no doc)

Make Every Moment Count

Stay With Me Now



Make Every Moment Count



Application Structure Related References

- Jesse Gallagher
blog: <https://frostillic.us>
- Dec's Dom Blog
blog: <http://www.qtzar.com/>
- Pipalia
blog: <http://www.pipalia.co.uk/blog/>
- John Daalsgard
blog: <https://www.dalsgaard-data.eu/blog/>
- Paul S. Withers
<http://www.intec.co.uk/blog/>

Amongst many others.

Concerning the App's Back-End

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

SSJS and Java



Make Every Moment Count

SSJS

- was meant to help bring in gun shy developers
- *com.ibm.jscript* isn't SSJS like other (more popular) SSJS implementations
- DDE often misses complex context assistance as JS is not a strongly type'd language

Java

- is a strong type'd language
- “no” (normal API induced) runtime exceptions
- is (more) extensible
- more industry norm
- (DD)Eclipse “does Java well”

A Horrific Pitfall



Spaghetti Code™

- un-supportable applications that have logic strewn about through design/markup (where is the code that's breaking?)
- overly large SSJS script libraries that add runtime parse bloat
- a lack of concise organization which any larger app *needs*

Make Every Moment Count

SSJS is a crutch (at least for non-trivial / non-RAD applications)

It works, but the overarching answer is to use the Java of XPages for better performance, structure, and sanity.

Making Sense of the Back-End pt.1



Make Every Moment Count

Data as a resource...

- how should the data:
 - validate
 - at the db / data object level
 - at the ui level
 - integrate
 - methods of exposure (collections, records, actions)
 - with other systems

Making Sense of the Back-End pt.2



Make Every Moment Count

Use the best tools at your disposal!

- [OpenNTF Domino API](#)
- [Git \(or Mercurial\)](#)
- [Swiper \(to eliminate extraneous output to ODP\)](#)
- Apache Commons libraries (e.g.- [CollectionUtils](#), [validators](#))
- [Google GSON](#)
- [Barista \(for bean fans\)](#)
- [Jesse's Controllers \(or whole frostillicus framework\)](#)

Java Policy



Make Every Moment Count

*Certain Java libraries won't run from an NSF container (versus OSGi plug-in) without the java.pol(icy) having additional permissions being granted.

Setting the grant block for all permission can sound “scary”, but really only means that you’re trusting your NSF contained Java code to not bork your server via the class loader involved.



Connect2016

The Premier Social Business and Digital Experience Conference



Back-End Components Demo (#1)

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Concerning the App's Front-End

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Fasten Your Seatbelt



Make Every Moment Count



Connect2016

The Premier Social Business and Digital Experience Conference

M-V-* and JS Frameworks



- an explosion of frameworks and libraries that can help make our development easier
- frameworks adopting MVC, MVM, MVVM, MV* approaches to automate the data bindings and logic
- a veritable potpourri of options and many developers have quite different takes on which framework to use and why
- this can be highly opinionated, even heated

Make Every Moment Count



Advanced Tooling

- dependency management for the front-end
- scaffolding repeatable parts of development
- task runners to provision:
 - distributable / deployment builds
 - complete with minification/uglification
 - automatic reload during development
 - unit / e2e testing
 - publishing documentation

Make Every Moment Count

In other words...



If you can, automate it!

Make Every Moment Count

Front-End Components Demo (#2)

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Opportunity Demo (#3)

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Opportunity Demo (#4)

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Q&A

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Summary



Make Every Moment Count

- we've covered the components of application structure options and practices to build better applications through segregated service layers
- ...practices and techniques to automate and expedite our data modeling and delivery of our data service with business logic, exposing it via RESTful API (or others)
- ...front-end frameworks for better UI-layers in applications
- ...advanced tooling which can help aid in creating those advanced front-ends and provide a completed picture of testing, documentation, and deployment beyond the back-end
- we did not cover how to grow a beard

Let's Build Happy Little Apps



Make Every Moment Count





Thank You

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count



Acknowledgements and Disclaimers

Availability. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.



Acknowledgements and Disclaimers cont.

© Copyright IBM Corporation 2015. All rights reserved.

- **U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.**
- IBM, the IBM logo, ibm.com, IBM Bluemix, IBM Domino, and IBM XPages are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Monty Python and the Holy Grail and its images and likenesses are intellectual property (trademarked or otherwise) of the Monty Python comedy group (a.k.a.- Monty Python's Flying Circus).

LEGO Deadpool and its character's likeness is trademarked and owned by Marvel and LEGO.

Range is property of Paramount Pictures and Nickelodeon Movies.

The IT Crowd is property of Talkback Thames and Retort, as aired on BBC.

The Joy of Painting, with artist Bob Ross, is property of Bob Ross and BRI Productions.

Other company, product, or service names may be trademarks or service marks of others.

