

# The Allure of Gothic Markup

Simon St. Laurent



York Minster, by GuyHH

# DISCLAIMER

- O'Reilly Media, Inc. bears no responsibility for the content of this presentation. This presentation in no way reflects the official views of O'Reilly Media, Inc. No other O'Reilly employees were involved in its creation. It exclusively represents the individual perspective of Simon St.Laurent.
- I am extremely lucky to have an employer that allows these kinds of wanderings. You should find one too, if you don't have one already.

# Ten Years Ago





# Montreal

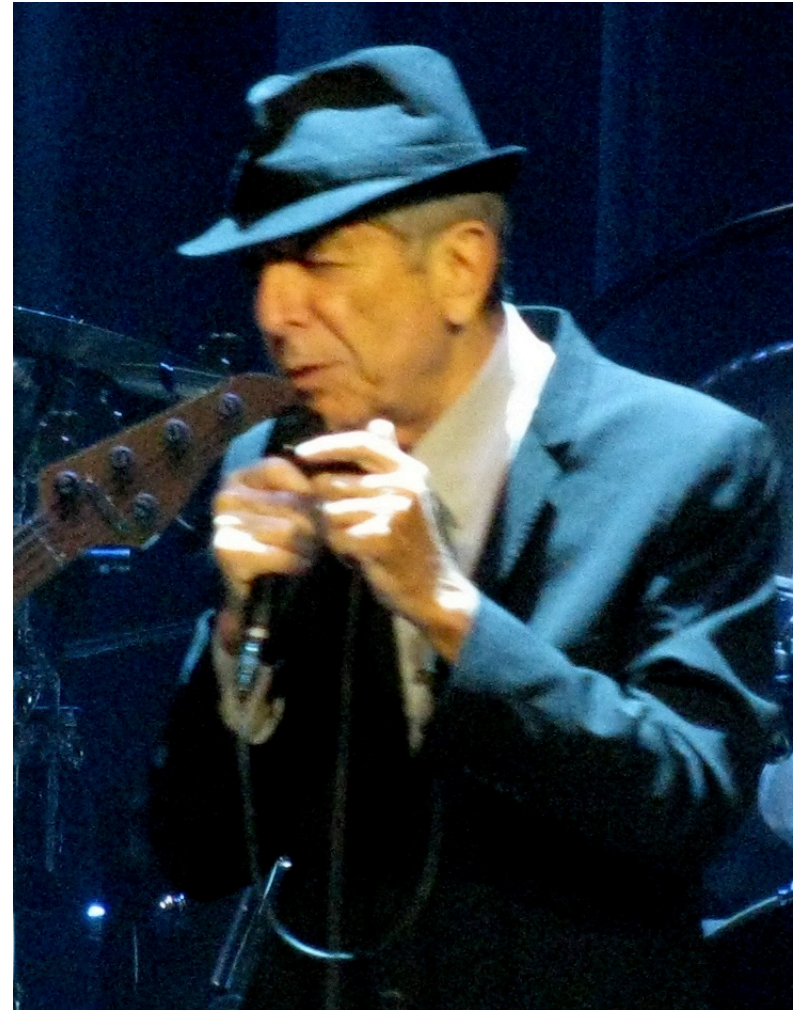


August 5, 2013.



# St. Leonard Cohen de Montreal

‘When they  
said  
“REPENT”  
I wonder  
what they  
meant.’



“The Future”, 1992.

Photo by Hyacinthe Rambault.

# We Are Fallen Creatures



Night Vigil, York Minster, photo by Catholic Church (England and Wales)



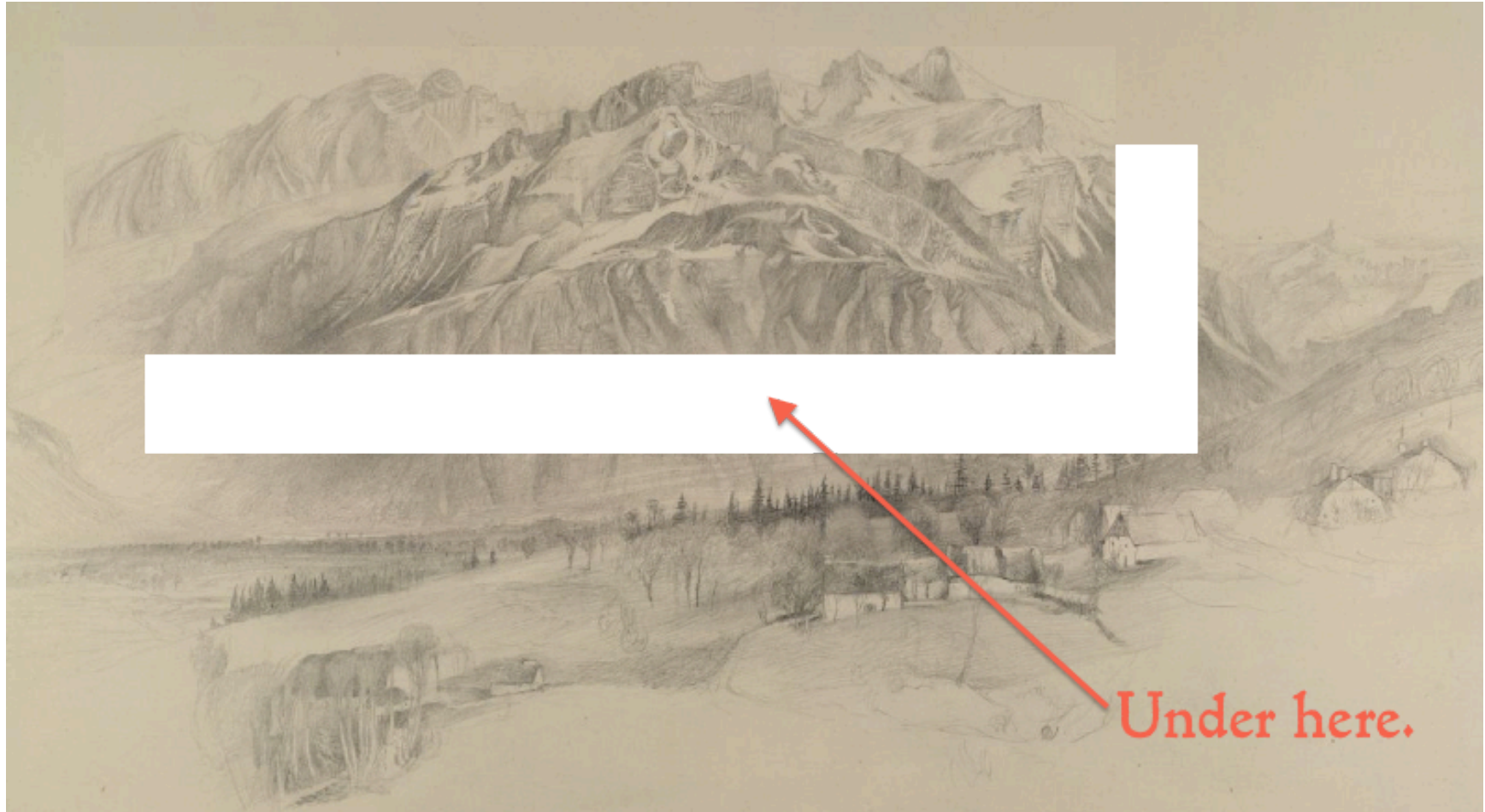
# Don't: Politics and Morality of Tech



John Ruskin, The Brezon and Alps of the Reposoir, seen from Mornex



# The Problems Lie Deep



# We Are Complicit – I Am Complicit



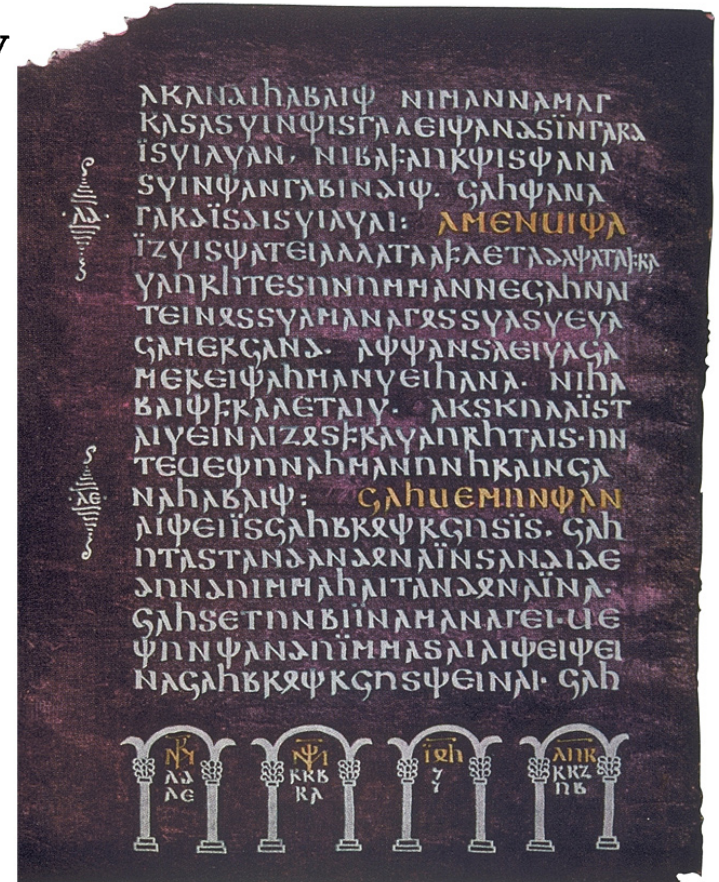
Photo by laughingsquid



Electronic Frontier Foundation Version



Codex Argenteus, the Bible in Gothic. Photo by  
Benutzer Signue, Wikimedia Commons



Nice pictures, but...

# WHAT HAS GOTHIC TO DO WITH MARKUP?



# Ruskin & The Nature of Gothic

- **The Stones of Venice** (1851/3): a book on architecture – evolves into a polemic about social organization and craft.



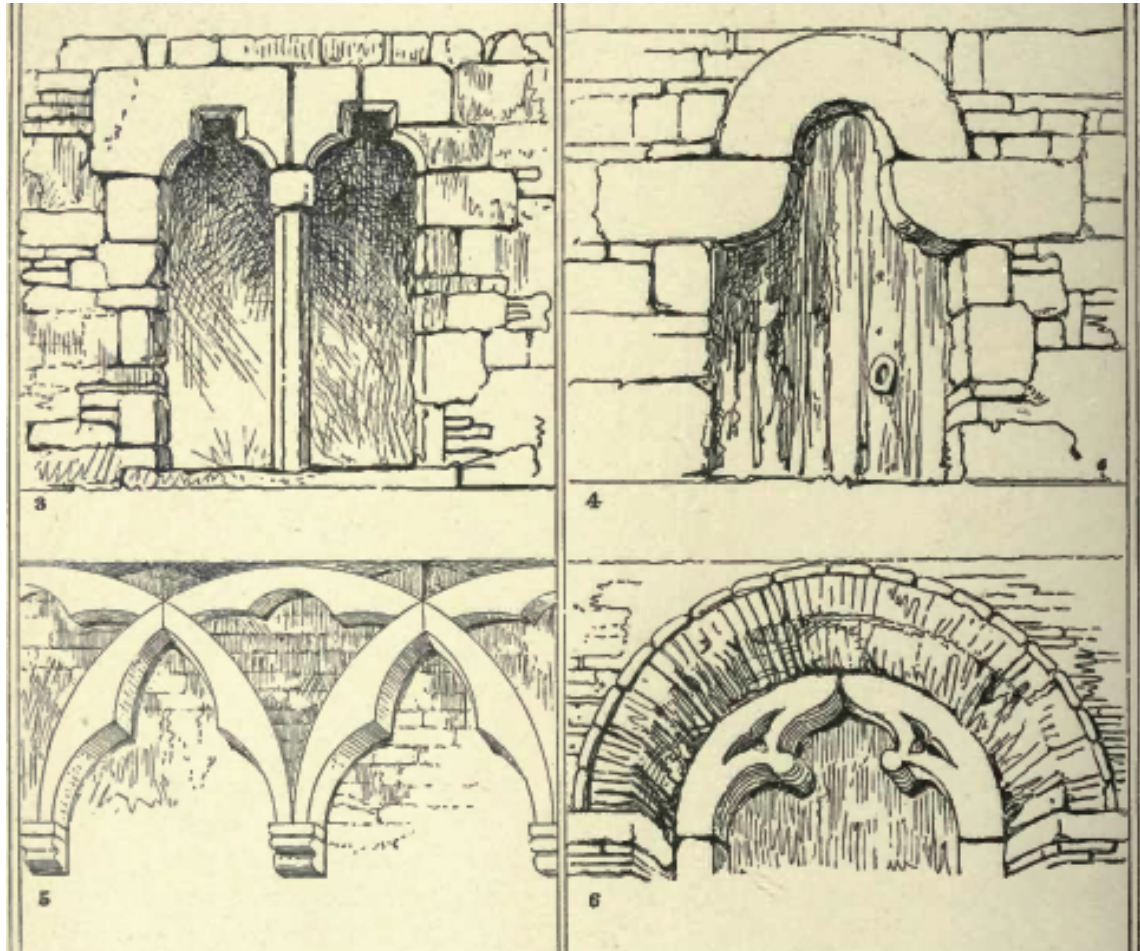
Ca D'Oro, Venice. Photo by Kenton Forshee

# Savageness





# Changefulness

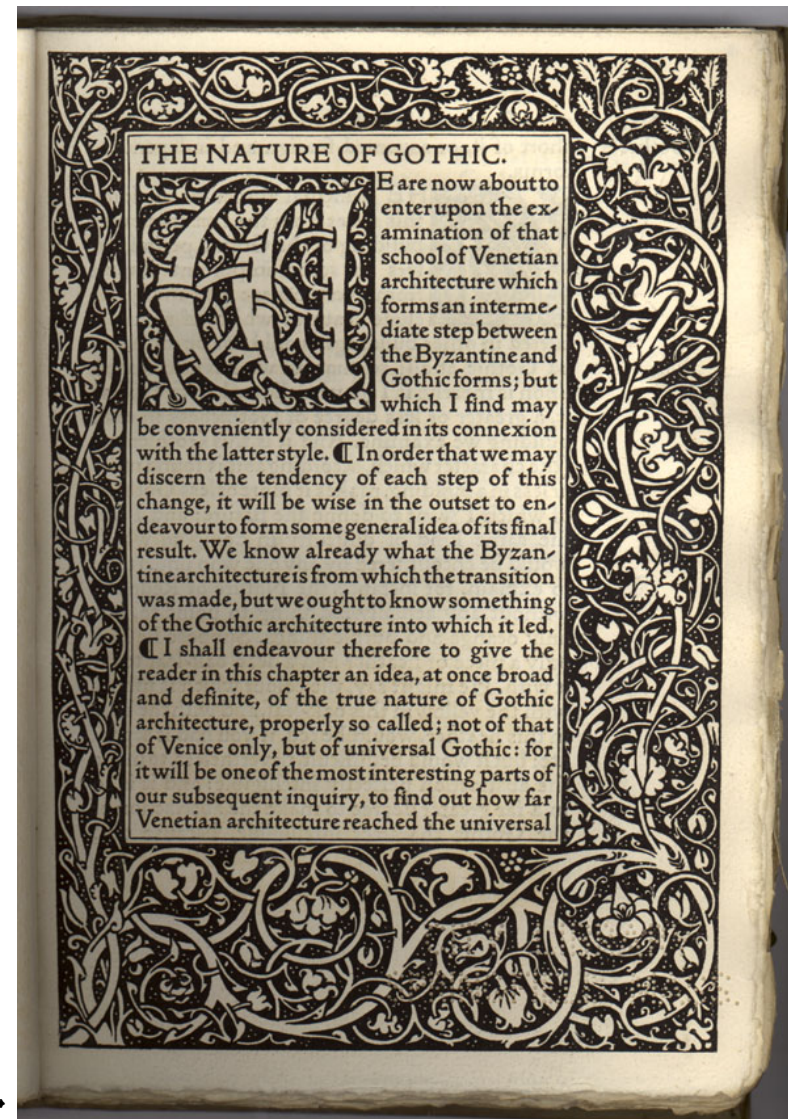


Arches with variations. John Ruskin, *The Stones of Venice*.



# William Morris

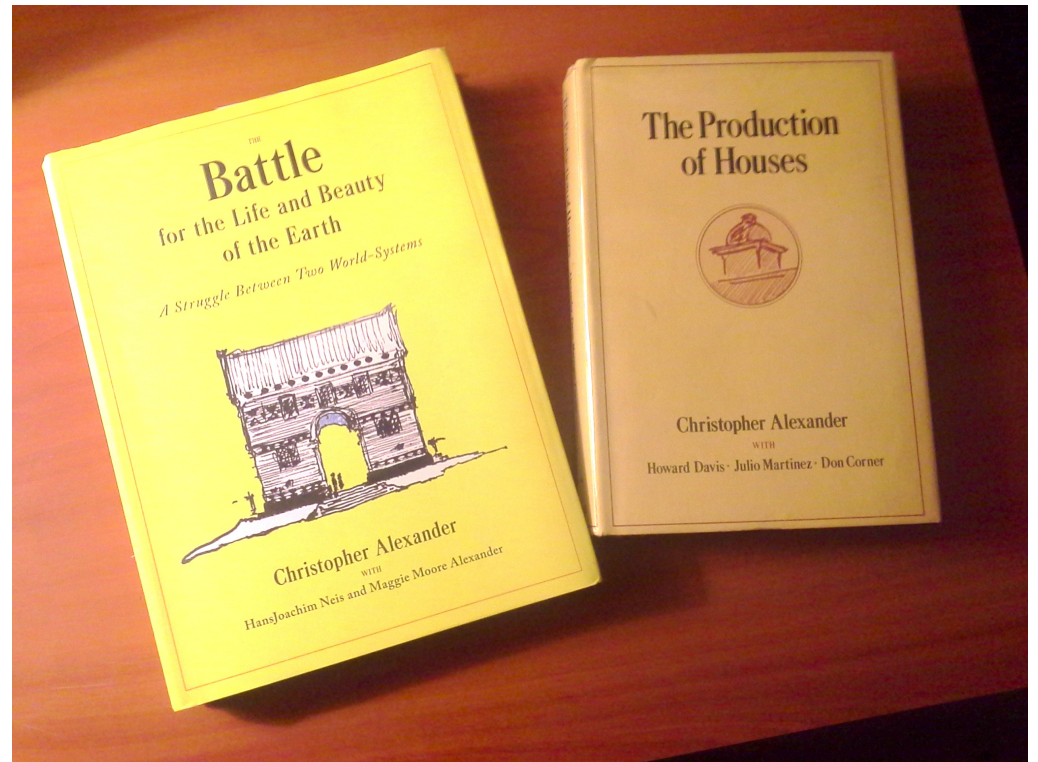
- “it is possible for man to rejoice in his work, for, strange as it may seem to us to-day, there have been times when he did rejoice in it... unless man’s work once again becomes a pleasure to him, the token of which change will be that beauty is once again a natural and necessary accompaniment of productive labour, all but the worthless must toil in pain, and therefore live in pain”



The Nature of Gothic, Kelmscott Press Edition (1892).

# Christopher Alexander

- **A Pattern Language** (1977), not **Design Patterns** (1994).
- Architect who wants to re-architect and re-orient architecture.



# System-A and System-B

- Iterative
  - Drawings after
  - Includes builders and clients at all stages
  - Continuous development from large features to small, unknown at start
  - Meets budgets by dropping features as necessary
  - Cost control through customization
- Planned (and corrections)
  - Drawings before
  - Hierarchy of designer to contractors to customers
  - All features included in original plan, possibly with options
  - Meets budgets by dropping quality or halting
  - Cost control through standardization



# Depth, Not Quantity

- “The architect-builder... has greater powers but more limited domain. Any one architect builder may control no more than twenty houses at a time, but he will take full responsibility for their design and construction, and he will work far more intensely with the individual families, and with the individual details of their houses. Thus, in this model of construction, both design and construction are decentralized....”

You didn't answer the question!

**BUT WHAT HAS THIS TO DO  
WITH MARKUP?**



# Markup as Architecture

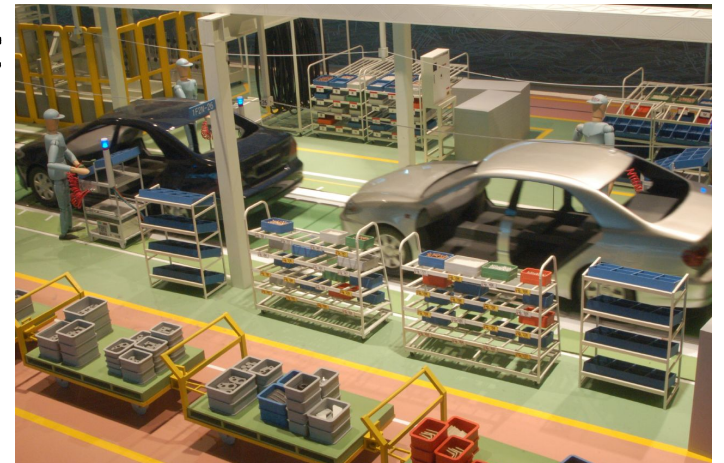
In virtual world, markup:

- Creates spaces in which we can talk
- Builds structures in which we can work
- Lays out marketplaces in which we shop
- Offers a canvas on which to decorate
- Assembles structures into larger groups

# XML as Industrialized Data

- Automate as much as possible.
- Prioritize predictability.
- Versioning means remodeling the factory.
- Standards about industrial collaboration.
- Human involvement driven to minimum.
- Contains anarchy by treating it as allowed but exceptional.

Photo by Urban Garden





# Validation



William Holman Hunt, The Scapegoat



# We Fear That Work Is This



Ford Madox Brown, Work



# Asynchronicity Works This Way



John Everett Millais, Christ in the House of His Parents (The Carpenter's Shop)

# Data Transformation Flows

- Changefulness shifts state.
- Transformations traditionally more forgiving than validators.
- XSLT as classic XML model, functional programming broader.
- Differentiate internal and external structures.
- “Data Pachinko”



Pachinko machine photo by limetom.



# Douglas Crockford: JSON and Functional Programming

- JSON extracts existing syntax from JavaScript
  - no invention.
- Crockford reorients developers toward functional JavaScript approaches.
- Practice avoids schemas, uses transformations from simple to massive.
- “Don’t be evil” license.

Photo by niallkennedy





# Human Transformers

- Reading as a transformation from “what you gave me” to “what I want to understand”, not just “taking in what is there”.
- Document creation as an especially flexible transformation.
- Even automated transformation tools should often be simpler even than “classic” XSLT 1.0, letting humans specify direction. CSS selectors a classic enabling tech, but should be more interfaces.

After the paper...

# **A FEW RECENT FINDS**

# John von Neumann

## PROBABILISTIC LOGICS AND THE SYNTHESIS OF RELIABLE ORGANISMS FROM UNRELIABLE COMPONENTS

J. von Neumann

### 1. INTRODUCTION

The paper that follows is based on notes taken by Dr. R. S. Pierce on five lectures given by the author at the California Institute of Technology in January 1952. They have been revised by the author but they reflect, apart from minor changes, the lectures as they were delivered.

The subject-matter, as the title suggests, is the role of error

(1951)



# Roger Costello: Fine-Grained Validation

- “how to validate / in an automated fashion / each item of an XML document and make validation decisions on a per-item basis rather than a per-document basis. The items to be validated may be as fine-grain or as course-grain as desired: the item may be a single element (or even just the text within a leaf element) or the item may be an element composed of many descendent elements. Validation may be done from the outside in (the root element to the leaf nodes) or the inside out (the leaf nodes to the root element).” (7/2013)

# Waterfall, Agile, Lean

- Waterfall: Know problem, figure out solution, implement.
- Agile: Know problem, iterate over solutions.
- Lean: Figure out problem, test solutions.  
Minimum Viable Product (MVP)
- “Test solutions” makes it easy to put people into a system at least at the outset, for “concierge options” that make a lot more sense for testing than complete build out.

# The Importance of Wheedling

- “When a programmer instructs a computer to reject, or disregard, all input longer than 64 characters, she effectively makes it impossible for a bureaucrat – however sympathetic – to accommodate a name that’s longer than she’s imagined names might be. With a human bureaucrat, there was always the possibility of wheedling an exception; machines don’t wheedle.”



**SO...**

# Emphasize Markup

- Markup as direct access to data.
- Programming languages shift back – transformations critical but not necessarily required.
- Declarative annotations offer tremendous power for presentation, hyperlinking, commentary, and more.

# The Promised Land



Ford Madox Brown, An English Autumn Afternoon, Hampstead – Scenery in 1853



# The Journey: Setting Out



Tapestry designed by Sir Edward Coley Burne-Jones, William Morris, and John Henry Dearle,  
The Arming and Departure of the Knights of the Round Table on the Quest for the Holy Grail



# Welcoming: Entry Points



Standen Garden. Photo by UGArdener



# Welcoming, Not Overwhelming



Kelmscott Manor, Lechlade, Gloucestershire. Photo by Tim Waters.

# Teaching: Get Them First!



Konrad St.Laurent



# Designing for Approachability



Mackintosh glass, Scotland Street School Museum. Photo by dalbera



# Maximum Welcome



Angels, Jesus College Chapel. Photo by TheRevSteve