John Martirano - Software Designer and Developer - JohnMartirano@gmail.com

SKILLS

I am a software designer and developer with a wealth of experience in areas ranging from real-time media and data-mining systems to enterprise web applications. I have experience in most roles and stages of software development. I am a strategic planner, an effective communicator and public speaker and a hard-nosed developer. I am a fan of documentation, code reviews and building consensus. I have enthusiasm for planning and meeting deadlines. My written, verbal, and visual communication skills are excellent.

EXPERIENCE

COMMON GROUND PUBLISHING

Senior Developer, Champaign, IL [MAY.2010 - Present]

- Designed and developed the Creator and Publisher web applications in the Scholar software suite.
- Gathered user and business requirements in the US and India, produced wireframes, designed features, wrote and managed sprint stories, planned releases, developed front and back-end code.
- Upgraded the entire suite from Rails 2.3.5 to Rails 3.1, re-architected 15 plugins into engine gems.
- Led the design and development of the custom state-based workflow subsystem, the role and permissions service and the AMQP-based event layer among many other critical components.
- Deeply customized Javascript CKEditor to produce the cross-browser Scholar semantic editor.
- Planned, setup and managed the human software testing (QA) team in Mumbai, India.
- RoR, Ruby, Sinatra, jQuery, Javascript, Postgres, services, workflow, TDD, Scrum

REVISION42

Independent Consultant [JUL.1997 - Present]

- Caterpillar, Motorola, Common Ground Publishing, Optalytic, Illinois Green Business Association.

NATIONAL CENTER FOR SUPERCOMPUTING APPLICATIONS

Senior Research Programmer, Visualization and Experimental Technologies [FEB.2001 - JAN.2005]

- Developed modular streaming media transport infrastructure used primarily to embed video avatars in networked virtual reality software for distributed data visualization. (C++, Linux, UDP, RTP/RTCP)
- Planned, setup and presented at major computing conferences throughout the USA and Europe.
- Collaborated with Stephen Hawking for IGRID 2002 and installed our software in the Hawking lab.
- Collaborated on Intellibadge RFID system to track and assist conference participants in real-time.

Software Engineer, Automated Learning Group, Champaign, IL [MAY.1997 - DEC.2000]

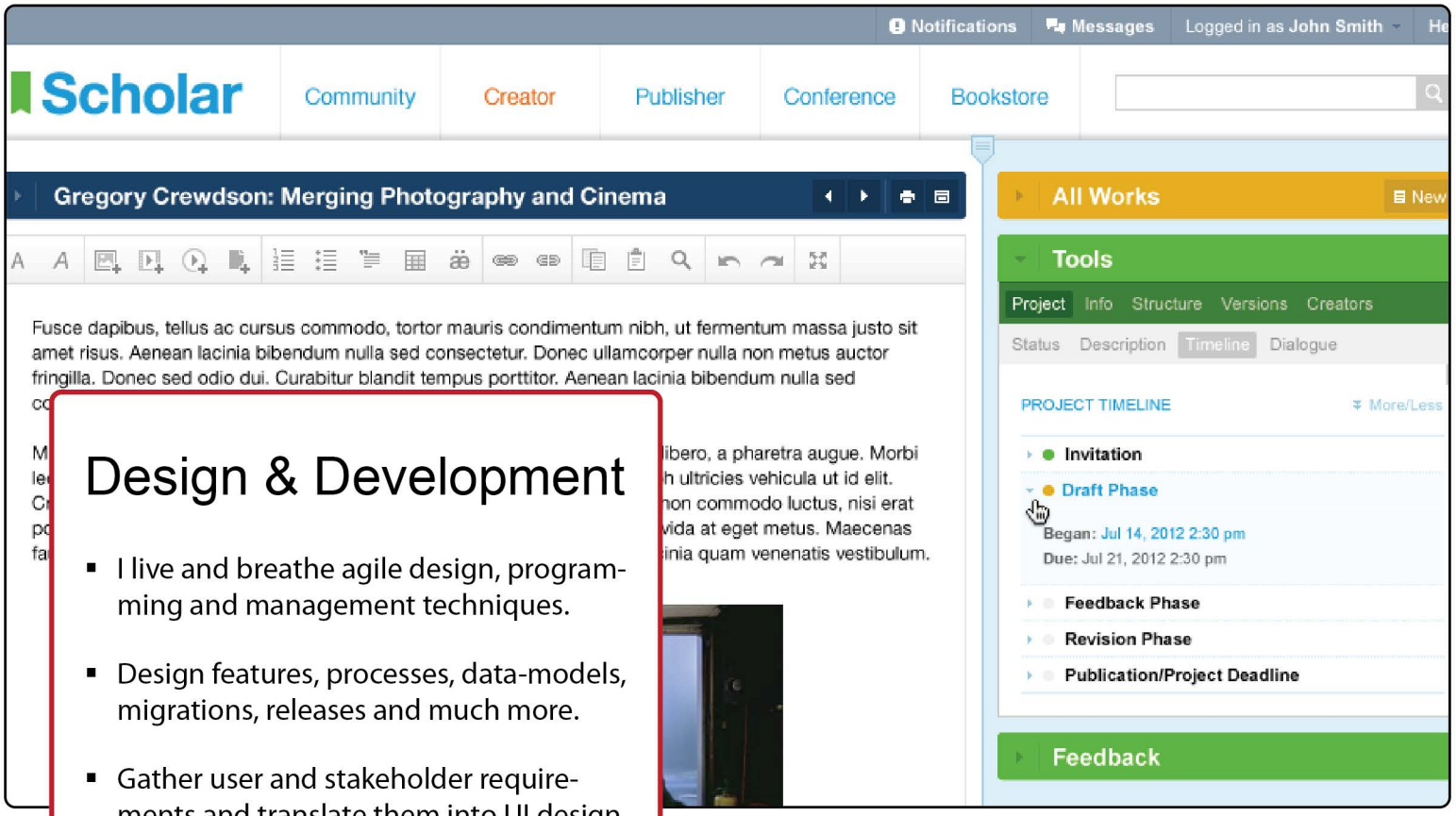
- Collaborated with a small team of machine learning experts to design data-mining solutions utilizing supercomputers for NCSA industrial partners: Sears, Allstate, Boeing, Caterpillar, Motorola, more.
- Designed the D2K (Data2Knowledge) framework, a 100% Java data-flow data-mining environment which saw 8 years of NSF funding and launched an analytics startup through the University. (Java)

EDUCATION

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

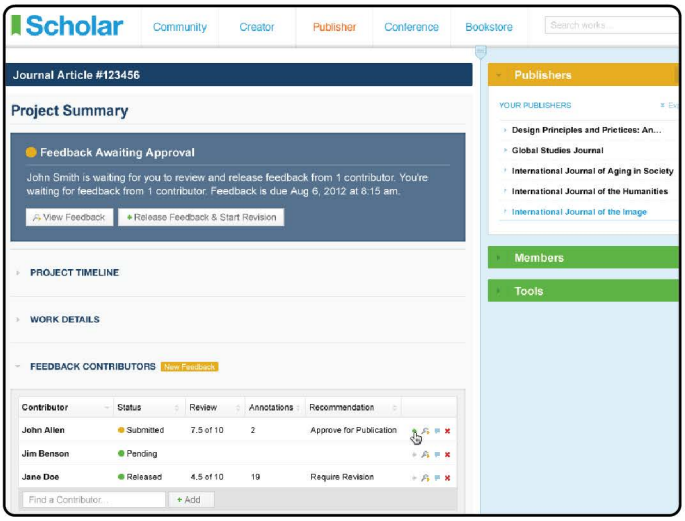
Masters of Science in the Graduate School of Library and Information Science [AUG.2003 - MAY.2005]
Bachelor of Fine Arts with Concentration in Digital and Interactive Media [AUG.1997 - DEC.2000]

John Martirano - Software Designer and Developer - JohnMartirano@gmail.com



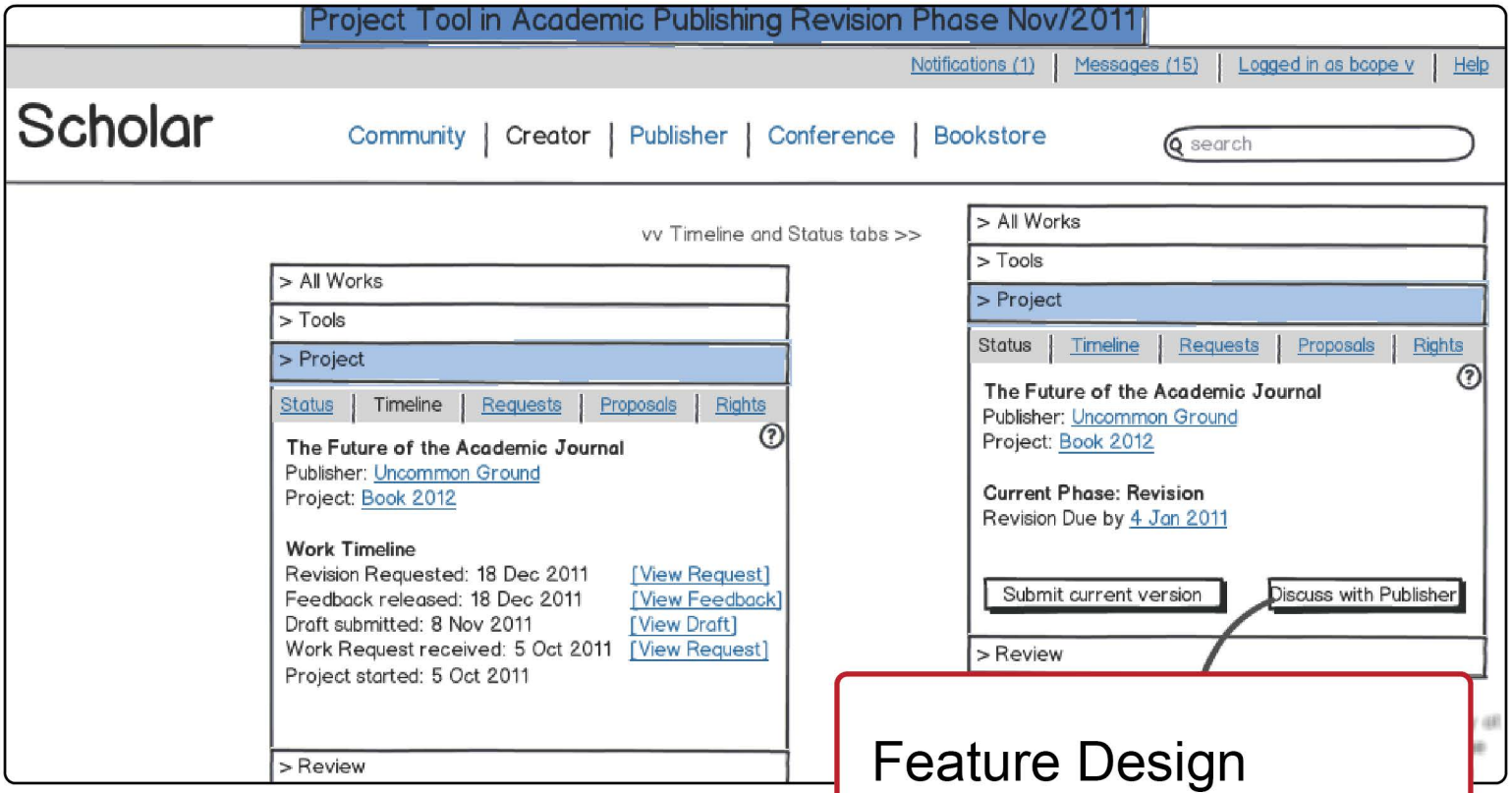
Design & Development

- I live and breathe agile design, programming and management techniques.
- Design features, processes, data-models, migrations, releases and much more.
- Gather user and stakeholder requirements and translate them into UI design.
- Communicate effectively, explore options, build consensus around solutions.
- Strategize to keep immediate goals in focus and guide toward the long term.
- Have experience in most roles and stages of software development.
- Bring best-practices learned from years of experience on a variety of teams.
- I write code too.

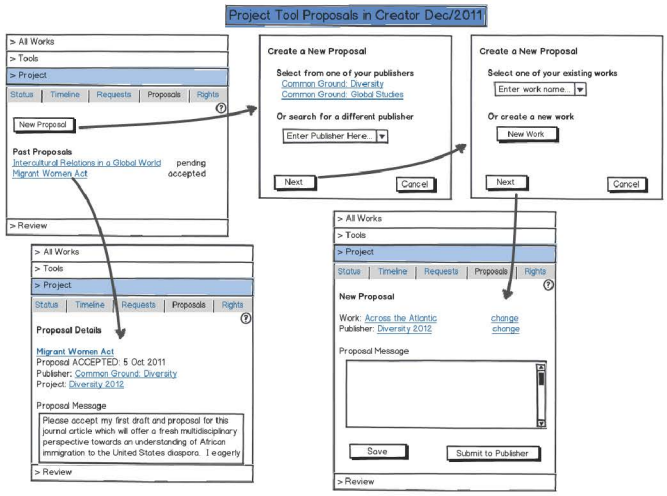


Scholar is a revolutionary publishing, authoring and elearning environment with active production deployments in US and Australian schools. I led the design and buildout of many of the critical parts of the Creator and Publisher applications while working with a small but dynamic team in Champaign, IL and with satelite members in India, Italy and elsewhere.

John Martirano - Software Designer and Developer - JohnMartirano@gmail.com



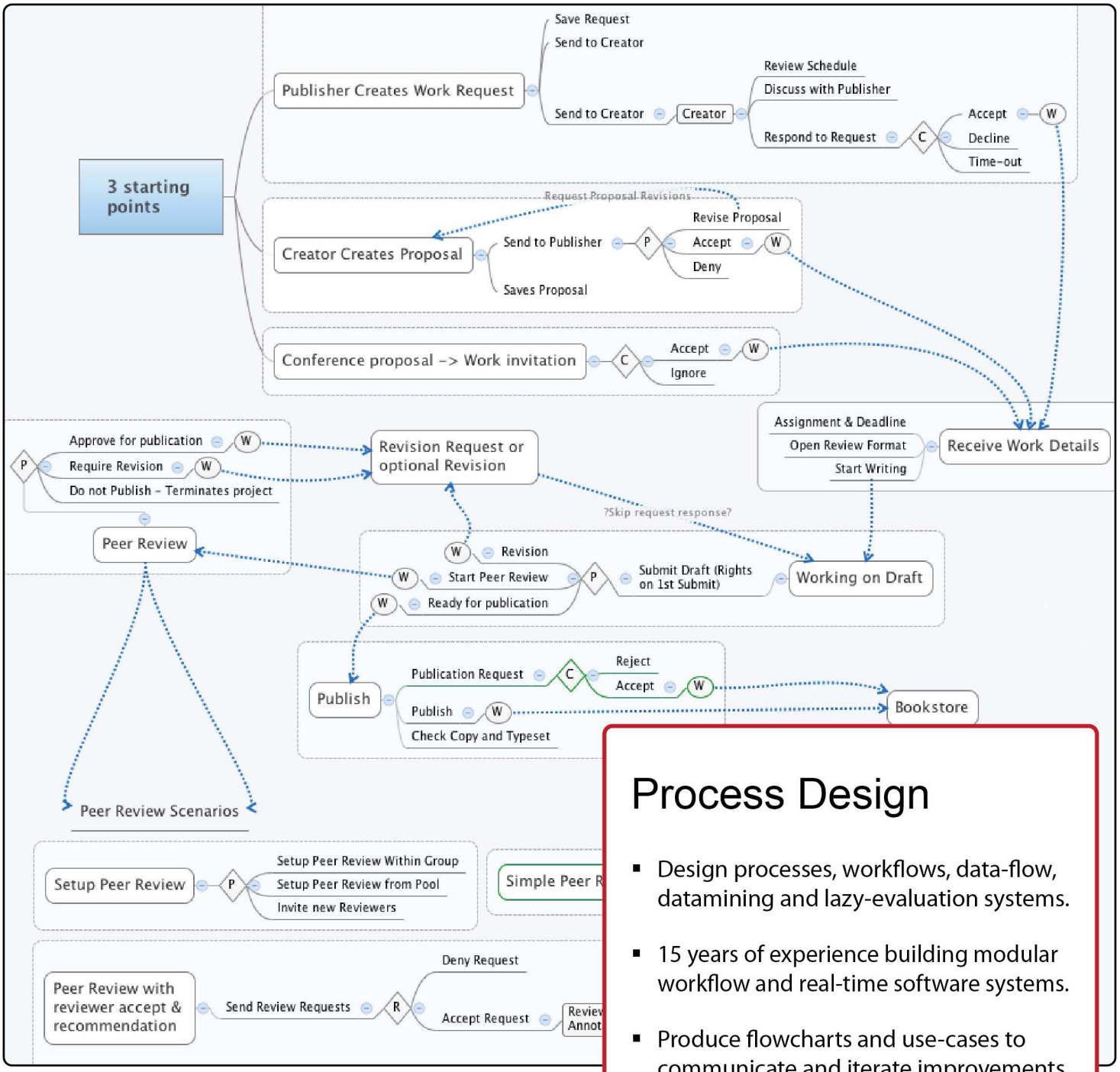
After producing a full set of wireframes for Scholar, I worked with stakeholders, our graphic designer and the development team to refine the outcome. I worked through often conflicting and diverging requirements from stakeholders, users and business realities to produce a cohesive set of software features expressed in a cogent UI.



Feature Design

- Work with stakeholders, gather requirements and understand business needs.
- Produce wireframes to create dialogue around proposed features.
- Work with graphic designers and others to refine mockups and UI.
- See the big picture and dive into the nitty gritty details.
- Imagine back-end implementation while considering front-end alternatives.
- Strive to remain detached and objective throughout the design process.

John Martirano - Software Designer and Developer - JohnMartirano@gmail.com



Process Design

- Design processes, workflows, data-flow, datamining and lazy-evaluation systems.
- 15 years of experience building modular workflow and real-time software systems.
- Produce flowcharts and use-cases to communicate and iterate improvements.

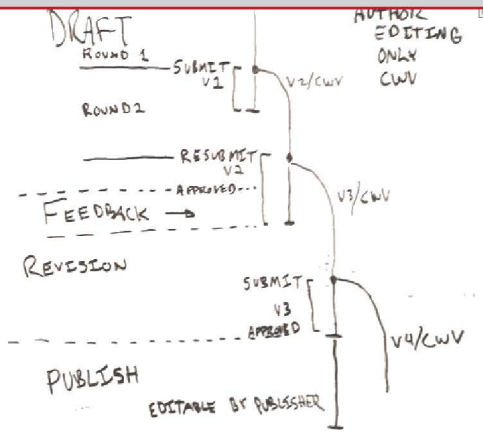
I built this flowchart while gathering the requirements for the Scholar peer-review and publishing workflow subsystem. This workflow subsystem was designed for and now has been deployed in 2 distinct markets: publishing and education. As with whiteboarding, flow-chart iterations were made quickly, often updating during a meeting based on real-time feedback.

John Martirano - Software Designer and Developer - JohnMartirano@gmail.com

Data Design

- Design data-models, well versed in ActiveRecord and Postgres.
- Write services to decouple and scale selected software components.
- Start with a whiteboard, a sketch of pseudocode or write a test and iterate.
- Use enough UML to communicate effectively but not drag on development.

The data-model sketched out above became the basis for our role and permissions service after a few quick iterations and plenty of feedback from team members. I led the design of this service and wrote the code which will be open-sourced in the near future. Discussion with team members covered examples ranging from Unix file permissions to the popular authentication and authorization ruby gem Devise.



```ruby
module CgEvent

  class Event
    include ::ActiveModel::Serializers::JSON
    include ::CgEvent::Publishable

    class << self
      def attribute_keys
        @attribute_keys ||= []
      end

      def event_attr_accessor(*syms)
        attr_accessor(*syms)
        syms.each do |sym|
          attribute_keys << sym
        end
      end
    end

    def initialize(attributes = {})
      self.attributes = attributes
    end

    def attribute_keys
      self.class.attribute_keys
    end

    def attributes
      self.class.attribute_keys.inject(
          ActiveSupport::HashWithIndifferentAccess.new) do |result, key|
        attribute_value = send(key)
        result[key] = attribute_value unless attribute_value == nil
        result
      end
    end

    def attributes=(attrs)
      # Throw an error if any attributes are missing
      attr_hash=ActiveSupport::HashWithIndifferentAc
      attribute_keys.each do |a|
        if !attr_hash.has_key?(a)
          raise EventAttributeMissingError, self.cla
        end
```

Coding

- Enjoy writing code, lots of it. Concise. Focused.
- Aggressively refactor.
- Deep experience in OOP and data-structures.
- Enthusiastic about TDD and documentation.
- Comfortable in the console, code with vim.
- Speak Rails, Ruby, jQuery, Java and much more.

I designed this event module to allow for various application and daemon processes to communicate via AMQP. These events are extremely lightweight, easy to define, create and publish, providing for an event layer which cuts across the entire codebase: simple and effective.

Define an event:

```ruby
class SubmittedReviewEvent < ProjectEvent
  event_attr_accessor :review_id
end
```

Publish an event:

```ruby
CgEvent::SubmittedReviewEvent.new(:review_id => 22).publish
```