Investigative Architecture
The Conceptual Diagram

The Open Group
23rd Enterprise Architecture Practitioners Conference
July 22th 2009

Ben Sommer, Senior Consultant, Systems Flow, Inc.
Dan Hughes, Principal Consultant, Systems Flow, Inc
James Hosey, Senior Consultant, Systems Flow, Inc.
Abstract

Formal UML-based notation provides a powerful tool for architecture design and communication. There is also a compelling need for crisp, clear, and marketing-friendly architecture diagrams for use in executive presentations. The creation of polished, clear and well-scoped renditions of an architecture can be more art than science, so we propose a diagram that balances the appropriate level of accuracy with enough flexibility for targeted marketing. This same view also provides a clear "at a glance" representation of an architecture, setting the scope for more detailed, formal views.
Presentation Overview

- The Challenge
- The Solution
- The Tool
- Investigative Architecture Review
  - Diagrams
  - Approach
  - Inputs
- Rubrics
- Case Study
- References
The Challenge

- Mission of enterprise architecture mission is to align technology design with business interests

<table>
<thead>
<tr>
<th>Intended Audience</th>
<th>Key Modeling Challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical Stakeholder</td>
<td>Proper scope and accuracy</td>
</tr>
<tr>
<td>Business Executive</td>
<td>Marketing &amp; selling solution</td>
</tr>
</tbody>
</table>
The Solution

- A formal conceptual diagram approach that results in a work product that is:
  - A *marketing view* flexible (and pretty) enough to catch attention in the board room or the sales pitch
  - Formal enough to guide an architect to the right design
  - Lightweight enough to quickly draft as a way to introduce stakeholders to the design
  - Refined enough to serve as a foundation for moving forward (can map to UML diagrams)
### Investigative Architecture Core Diagrams

<table>
<thead>
<tr>
<th>Diagram</th>
<th>Scope</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="Image" alt="Conceptual Overview" /></td>
<td>PowerPoint View</td>
<td>Icon-based w/guidelines</td>
</tr>
<tr>
<td><img src="Image" alt="Logical Deployment" /></td>
<td>System View</td>
<td>UML Component Diagram</td>
</tr>
<tr>
<td><img src="Image" alt="Data Context" /></td>
<td>Data View</td>
<td>UML Collaboration Diagram</td>
</tr>
</tbody>
</table>

See [Leveraging UML as a Standard Notation for Enterprise Architecture](#) and [Investigative Architecture – Making Sense of your Enterprise](#) for additional information.
Investigative Architecture Process

Architect

Stakeholders

Until desired level of refinement is reached

Gather

Diagram

Review

Refine

Publish
# Investigative Architecture Inputs

<table>
<thead>
<tr>
<th>Information Source</th>
<th>What to Expect</th>
<th>Target Diagram</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor</td>
<td>Product Documentation</td>
<td>Logical Deployment</td>
</tr>
<tr>
<td><strong>Support Engineer</strong></td>
<td><strong>Operations Manuals</strong></td>
<td><strong>All</strong></td>
</tr>
<tr>
<td>Server Engineer</td>
<td>Server Documentation</td>
<td>Logical Deployment</td>
</tr>
<tr>
<td>DB Administrator</td>
<td>Database Information</td>
<td>Logical Deployment, Data Context</td>
</tr>
<tr>
<td><strong>Business Line</strong></td>
<td><strong>Requirements Artifacts</strong></td>
<td><strong>Conceptual Overview, Data Context</strong></td>
</tr>
<tr>
<td>Network Engineer</td>
<td>Network Topography</td>
<td>Logical Deployment</td>
</tr>
<tr>
<td><strong>Company Intranet</strong></td>
<td><strong>Links to above information</strong></td>
<td><strong>All</strong></td>
</tr>
<tr>
<td><strong>EA Repository</strong></td>
<td><strong>As-is visual models</strong></td>
<td><strong>All</strong></td>
</tr>
<tr>
<td><strong>Google, Wikipedia, etc.</strong></td>
<td><strong>Stray pieces of the puzzle</strong></td>
<td><strong>All</strong></td>
</tr>
<tr>
<td>CMD.EXE (or csh)</td>
<td>Network information</td>
<td>Logical Deployment</td>
</tr>
</tbody>
</table>
Conceptual Diagram Quick Start

- Left to right depiction of user driven architecture
- Uses standard set of icons
- Designed to fit on one presentation slide
Conceptual Diagram Guidelines

• Assume the audience does not any familiarity with the architecture
• Keep the focus high level and conceptual
• Do not get distracted by inaccurate details
  – Target “big picture” accuracy
• Target the diagram to your need
<table>
<thead>
<tr>
<th>The Company</th>
<th>Massive Insurer, Inc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Vendor</td>
<td>EzeDoesIT, Inc.</td>
</tr>
<tr>
<td>The Product</td>
<td>EzeWorkflow</td>
</tr>
<tr>
<td>The Project</td>
<td>In disarray</td>
</tr>
<tr>
<td>The Task</td>
<td>Produce a Conceptual Diagram of the target solution</td>
</tr>
</tbody>
</table>
To: 'ben.sommer@sysflow.com'
Cc: 
Subject: Starter info - Ezeworkflow implementation

Attachments: EzeWorkflow_Overview.ppt (80 KB)

Ben – see attached overview from the vendor. We'll be rolling this out for claims, documentation middle office and possibly contract adjusters.

Best,
--
Marty K.
Ops Director, Claims
Massive Insurer, Inc.
Case Study (continued)

Stub Users

Users

Delivery Mechanism

Application

Services

Resources

“...possibly contract adjusters”

Adjusters

“..rolling this out for claims”

Claims Agents

Documentation Specialists

cumulation middle office”

Case Study (continued)
Input #2 – Vendor “Marketecture”

**EzeWorkflow – Overall Approach**

The proposed solution provides the platform for *Massive Insurer, Inc.* to transform its claims business, providing straight through processing from submission to closure.

- **Legacy/external systems,**
- **EzeWorkflow Manager (EWM),**
- **The “killer app” for claims,**
- **Fax claim submission,**
- **Archive integration (CMOD, filenet etc),**
- **EzeWorkflow Integrator (EWI),**
- **EzeWorkflow DoX (EWD),**

**ONE platform**

Built on open-source LAMP stack!
Application & Legacy Systems

EzeWorkflow Integrator...
Legacy...systems

ONE platform

Users

Delivery Mechanism

Application

Services

Resources

Adjusters

Claims Agents

Documentation Specialists

“ONE platform...” — for now black-box EWM, EWI & EWD

EzeWorkflow

Customer/policy systems

Systems of record on Mainframe

Case Study
(continued)
Input #3 – Google Sleuthing

Good old Wikipedia...

LAMP (software bundle)

From Wikipedia, the free encyclopedia

The acronym LAMP refers to a solution stack of software, usually free and open source software, used to run dynamic Web sites or servers. The original expansion is as follows:

- Linux, referring to the operating system;
- Apache, the Web server;
- MySQL, the database management system (or database server);
- one of several scripting languages: Perl, PHP or Python.[1]

“Apache Web Server”...smells like a web app

And a database
App – Front-end & Back-end

Users
- Adjusters
- Claims Agents
- Documentation Specialists
- Web Browser

Delivery Mechanism

Application

Services

Resources

EzeWorkflow

Customer/policy systems

EzeWorkflow DB

Database is a “resource”

Web browser = web application

Case Study (continued)
Case Study (continued)
Fax & Archive?
Input #4 – Intranet Sleuthing

Internal documents insinuate that RightFax is standard fax integration app. Filenet + RightFax – Bingo!

Case Study (continued)

Faxes coming in...

...capture & archive going out
Case Study (continued)

“EzeWorkflow Dox™” (cont’d)

1. **Position RightFax as an “application”**
   - Users: Customers, Claims Agents, Documentation Specialists
   - Delivery Mechanism: Fax Machine
   - Application: RightFax
   - Services: FileNet (Image Ingestion/Retrieval Services)
   - Resources: Claims Archive, Customer/policy systems, EzeWorkflow DB

   - Showing direction here makes sense (remember key word “fax submission”)
   - “Zoom in” on FileNet by translating the server icon into a bordered pair of “Services” and “Resources” icons

   - Name the archive descriptively, for the application
   - Name the services descriptively – make an educated guess
Input #5 – Follow-up Email to Business

"...possibly contract adjusters"

RE: Starter info - Ezeworkflow implementation - Message (HTML)
To:  'martyh@massive-inc.com'
Subject: RE: Starter info - Ezeworkflow implementation

Thanks Marty. Can you please clarify the following about the “contract adjusters”?

1. Are they staff-augmentation type contractors (i.e. individuals) – or who field teams (i.e. companies)?
2. Do they use any of Massive’s internal apps for adjustment workflow or would we just “throw data over the wall”?
   ○ On that note – is it envisioned we’d send them an extract of claims to work periodically – i.e. once/day? Or would it be more complicated (i.e. real-time)?

Thanks!
Ben

RE: Starter info - Ezeworkflow implementation - Message (HTML)
To:  'ben.sommer@sysflow.com'
Subject: RE: Starter info - Ezeworkflow implementation

They’re companies and free-lancers. No internal apps, though the companies do have their own workflows. Data – we fax back and forth intraday now so a daily file exchange would great if it’s hands-off/automated.

Marty K.
“Misc. Adjustment Workflow Systems”

Model “Contract Adjusters”

Adding our first batch connection—distinguish dashed from solid lines in a legend.

Make both the systems and their interfaces a “black box”

Cases Study (continued)

“daily file exchange...

...have their own workflows...”
Tidy Up

Case Study (continued)

Diagram is getting noisy – is the DB adding anything?
References

- Leveraging UML as a Standard Notation for Enterprise Architecture
- Other Systems Flow Whitepapers
QUESTIONS?
Ben Sommer (ben.sommer@sysflow.com) is a senior consultant with Systems Flow, Inc, www.sysflow.com, where he helps organizations dramatically improve their competitive advantage through the practical, effective application of best practices in enterprise architecture and software development. Ben is currently consulting at Citizens Bank, providing architectural leadership for strategic IT projects. His career has spanned network engineering, systems administration, and software development - running the gamut from tools to automate network and systems tasks, to web-based CRM applications, to Identity Management and Provisioning systems, to real-time music synthesis applications. His industry experience includes education, education finance, interactive marketing and banking. Ben is a trained composer and musician.

Dan Hughes (daniel.hughes@sysflow.com) is a principal consultant with Systems Flow, Inc. He is currently engagement lead at Citizens Bank where he guided the launch of the enterprise architecture practice and is now the lead architect for Citizens Bank's Basel II implementation. Dan has 16 years of software engineering experience spanning a broad range of technologies and techniques. Startup to enterprise, he has launched, managed, and executed all aspects of both product and enterprise life cycle for clients in industries ranging from industrial automation to banking and insurance. He maintains a blog on software engineering at xengineering.com. He holds a Bachelor of Science in Computer and Systems Engineering from Rensselaer Polytechnic Institute.

James Hosey (james.hosey@sysflow.com) is a senior consultant with Systems Flow, Inc. He is currently engaged at Citizens Bank as an enterprise architect providing strategic architectural guidance and project-specific support across the bank's technology portfolio. Over the course of his 16-year career, Jim has managed and executed all phases of the software life cycle and has delivered a wide variety of technology solutions for both commercial resale and internal use in domains that include banking, insurance, warehousing & distribution, marketing, communications, and management training & development. Having worked with organizations of all sizes, Jim can tailor his approach to the specific driving forces within each type of environment. His experience managing his own consulting practice for ten years has provided him with the entrepreneurial experience necessary to work with stakeholders at all levels to achieve results.