

Investigative Architecture

Making Sense of Your Enterprise

The Open Group

19th Enterprise Architecture Practitioners Conference

July 22th 2008

Ben Sommer, *Senior Consultant*, Systems Flow, Inc.

Dan Hughes, *Principal Consultant*, Systems Flow, Inc

James Hosey, *Senior Consultant*, Systems Flow, Inc



Abstract

A foundational skill for an architect is the ability to rapidly assess and document "as is" and proposed solution architectures. The challenge lies in the typical state of enterprise knowledge regarding the systems - a myriad of internal and external information sources at all levels of quality and completeness. Rapidly converting this sea of information into useable knowledge requires a repeatable, structured approach for gathering information from internal stakeholders and documents, as well as performing focused research for publicly-available product and industry information.

Presentation Overview

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

The Challenge

- Enterprise knowledge regarding systems is usually in a sorry state
 - Disparate information sources
 - All levels of quality and completeness
- Architects are charged with untangling this mess

The Solution

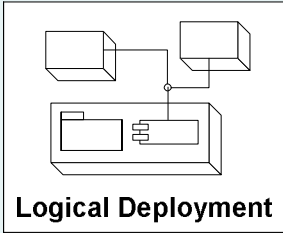
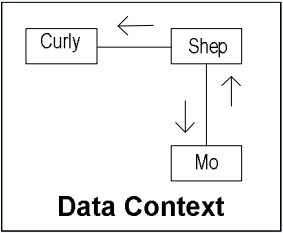
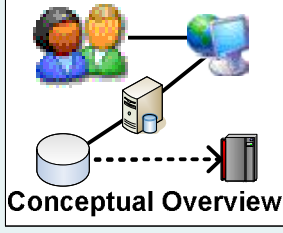
Understand and document a system's architecture

- Understand
 - Interrogate and interview
 - Read the tea leaves
 - Sift through junk (to produce a gem)
- Document
 - Choose the right diagram type
 - Go for clarity and scope

The Tool

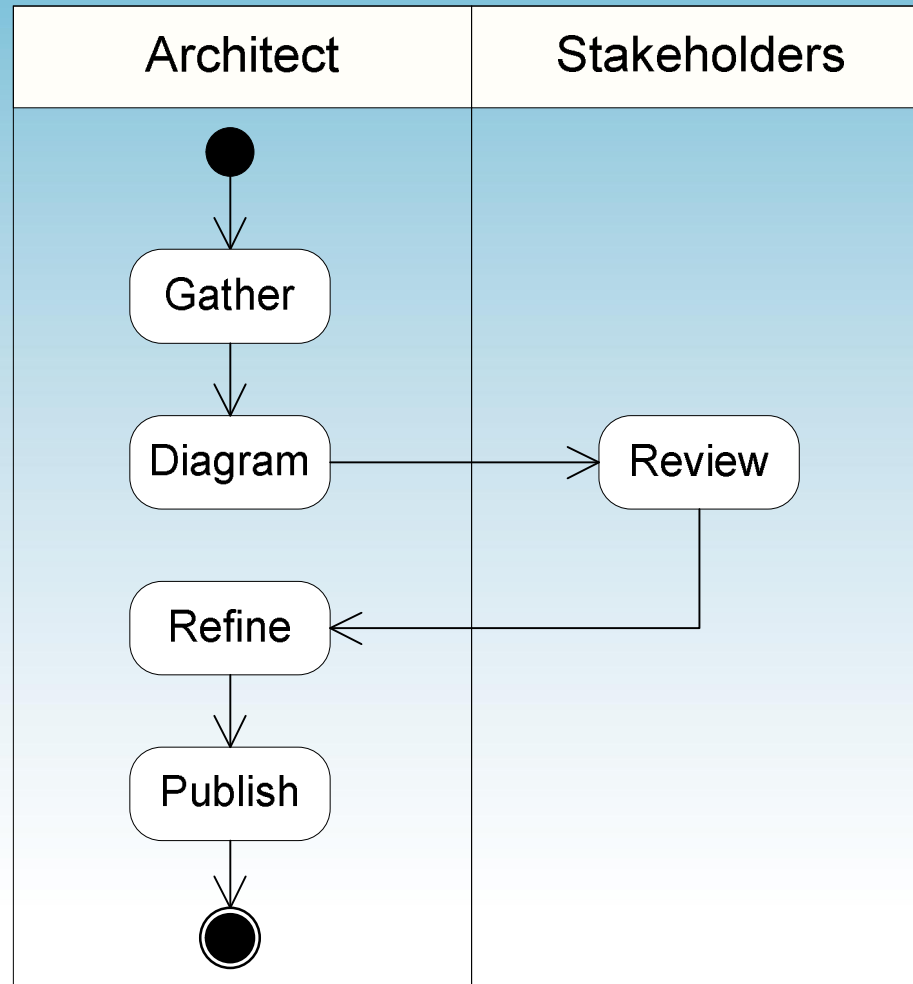
- Foundation of Systems Flow's method is a structured set of UML diagrams
- Why UML?
 - Industry standard diagramming notation
 - Controls *form*, encouraging focus on *content*
 - Precise structure encourages consistency, clarity, and scope
 - There's a UML diagram type for every situation

Investigative Architecture Diagrams

Diagram	Scope	Notation
 <p>Logical Deployment</p>	System View	UML Component Diagram
 <p>Data Context</p>	Data View	UML Collaboration Diagram
 <p>Conceptual Overview</p>	PowerPoint View	Icon-based w/guidelines

See [Leveraging UML as a Standard Notation for Enterprise Architecture](#) for additional diagram types

Investigative Architecture Approach



Investigative Architecture Inputs

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

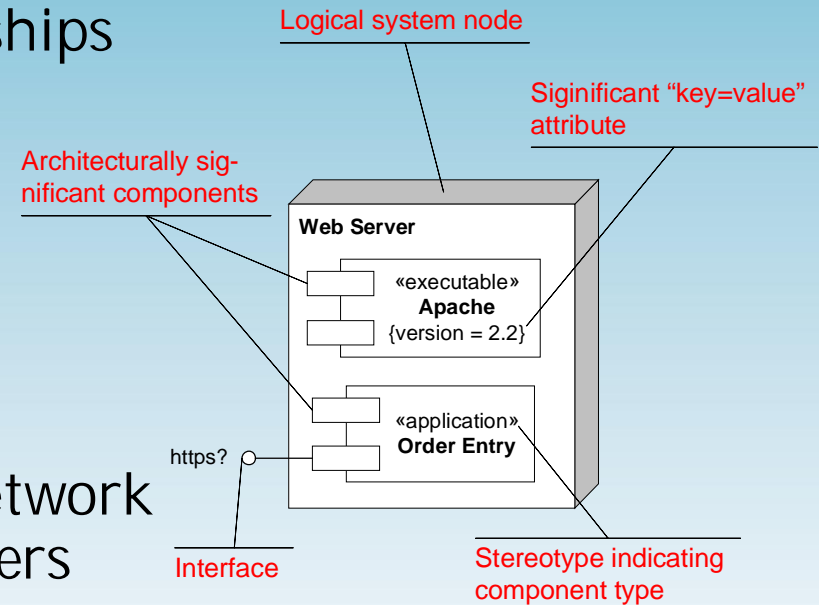
Investigative Architecture Case Study

The Company	<i>Massive Insurer, Inc.</i>
The Vendor	<i>EzeDoesIT, Inc.</i>
The Product	<i>EzeWorkflow</i>
The Project	In disarray
The Task	Produce a Logical Deployment Diagram of the target solution

Case Study

Side-bar – The Logical Deployment Diagram

- Static view of systems and relationships
- Logical nodes inventory hardware and software needs
- Nodes and components highlight support tasks and responsibilities
- Protocol identification useful for network and information security stakeholders
- Basis for many other stakeholder specific artifacts – queuing views, code deployment views, etc.
- Roadmap artifact for most stakeholders including development, infrastructure, and support teams



Input #1 – Vendor Documentation

System Requirements	
EzeWorkflow Manager 6.1	
Desktop	
Windows 2000, XP	
Microsoft Internet Explorer 6.x, 7.0	
Adobe Acrobat Plug-In 6.0	
HTTP Server	
IBM HTTP Server 2.0+	
Apache Web Server 2.0+	
Application Server	
IBM WebSphere Application Server v5.1.2, v6.1	
JBoss Application Server	
Database Server	
IBM DB2 UDB 8.1+, Oracle 10g	
Native Java Machine enabled	
EzeWorkflow Remote Admin	
Desktop	
Windows 2000, XP	
.NET Framework v1.1	

Case Study
(continued)

Stub Logical Nodes

System Requirements	
EzeWorkflow Manager 6.1	
Desktop	
Windows 2000, XP	
Microsoft Internet Explorer 6.x, 7.0	
Adobe Acrobat Plug-In 6.0	
HTTP Server	
IBM HTTP Server 2.0+	
Apache Web Server 2.0+	
Application Server	
IBM WebSphere Application Server v5.1.2, v6.1	
JBoss Application Server	
Database Server	
IBM DB2 UDB 1+, Oracle 10g	
Native Java VM line enabled	
EzeWorkflow Manager Admin	
Desktop	
Windows 2000, XP	
.NET Framework 2.0	

Desktop

HTTP Server

Application Server

Database Server

Back-office Workstation

EzeWorkflow Web Server

EzeWorkflow Database Server

EzeWorkflow Application Server

Case Study
(continued)

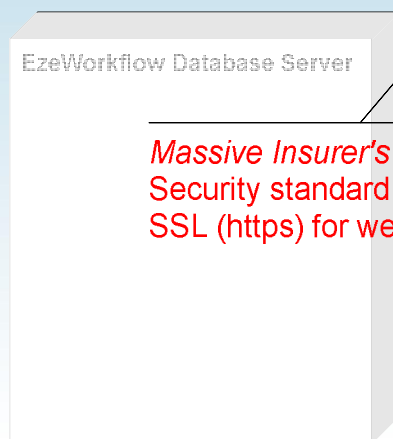
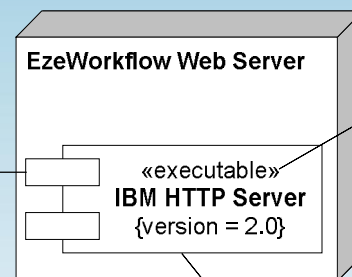
Add Web Server Node Detail

System Requirements	
EzeWorkflow Manager 6.1	
Desktop	
Windows 2000, XP	
Microsoft Internet Explorer 6.x, 7.0	
Adobe Acrobat Plug-In 6.0	
HTTP Server	
IBM HTTP Server 2.0+	
Apache Web Server 2.0+	
Application Server	
IBM WebSphere Application Server v6.1.2, v6.1	
JBoss Application Server	
Database Server	
IBM DB2 UDB 8.1+, Oracle 10g	
Native Java Machine: enabled	
EzeWorkflow Remote Admin	
Desktop	
Windows 2000, XP	
.NET Framework v1.1	

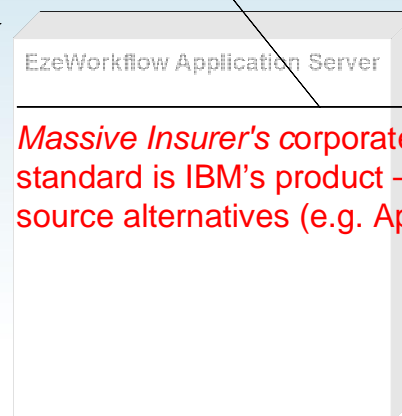
Back-office Workstation

Rule of thumb for labeling interfaces:
"Use the highest *known* protocol on
the network stack"

A "HTTP" web server is by
definition an «executable» compo-
nent - we stereotype it as such



Massive Insurer's Information
Security standard mandates
SSL (https) for web servers

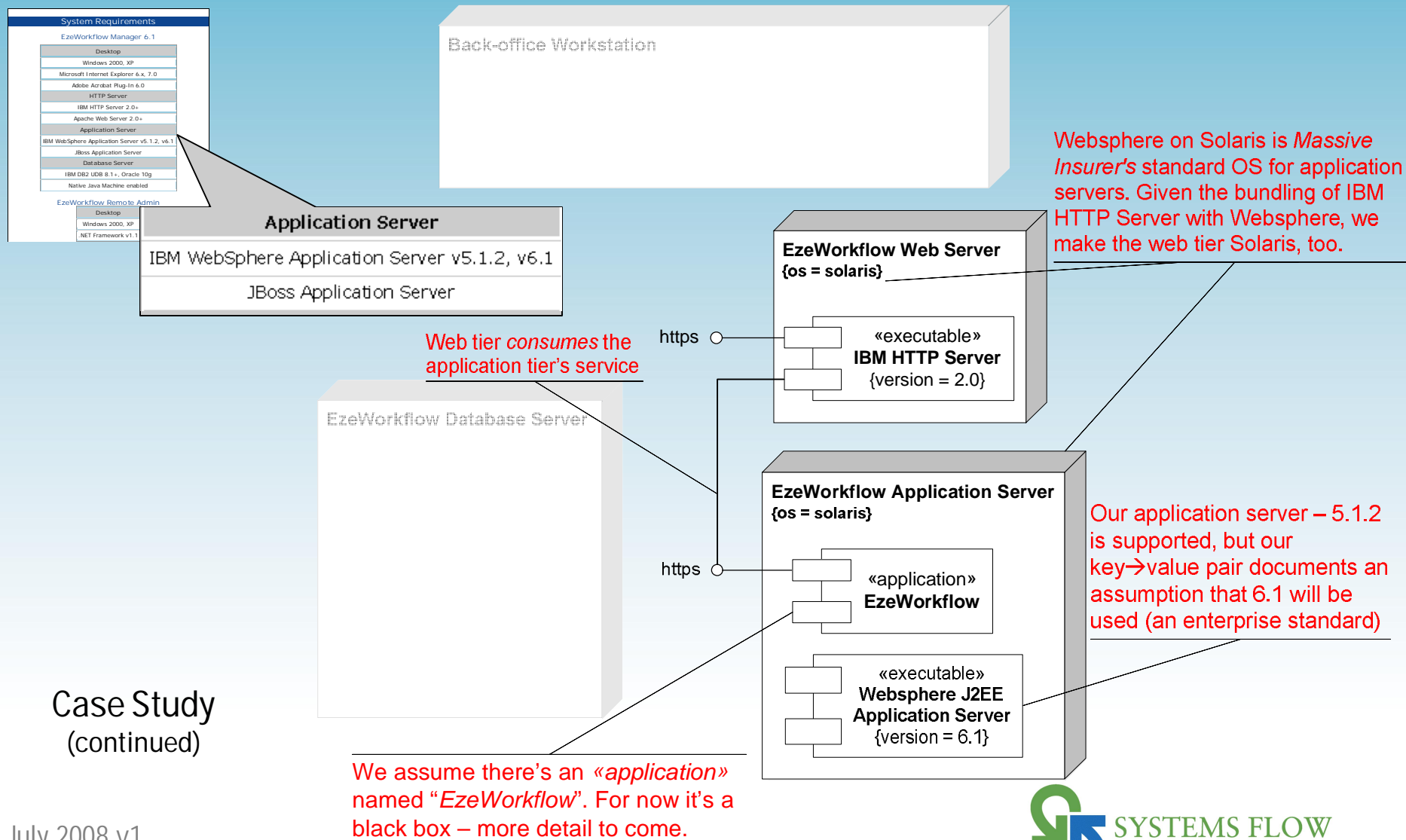


Massive Insurer's corporate web server
standard is IBM's product – no open
source alternatives (e.g. Apache)

https

Case Study
(continued)

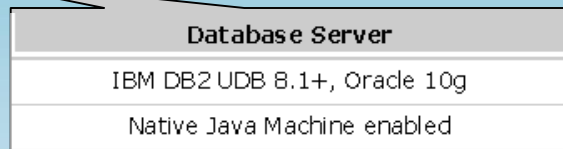
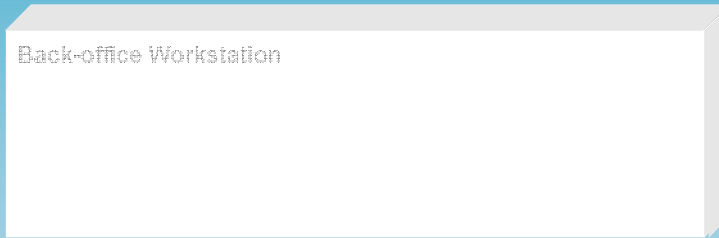
Add Application Server Node Detail



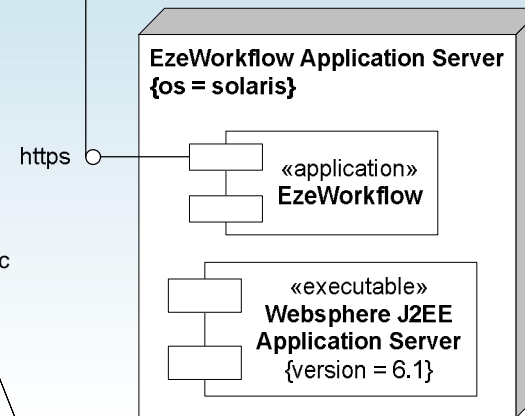
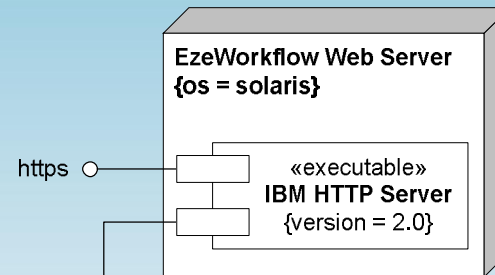
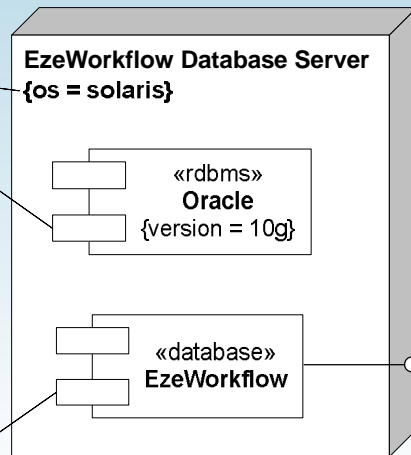
Case Study
(continued)

Add Database Node Detail

System Requirements	
EzeWorkflow Manager 6.1	
Desktop	
Windows 2000, XP	
Microsoft Internet Explorer 6.x, 7.0	
Adobe Acrobat Plug-In 6.0	
HTTP Server	
IBM HTTP Server 2.0+	
Apache Web Server 2.0+	
Application Server	
IBM WebSphere Application Server v5.1.2, v6.1	
JBoss Application Server	
Database Server	
IBM DB2 UDB 8.1+, Oracle 10g	
Native Java Machine enabled	
EzeWorkflow Remote Admin	
Desktop	
Windows 2000, XP	
.NET Framework v1.1	



Oracle on Solaris is the enterprise standard for databases



Case Study
(continued)

July 2008 v1
Slide 16

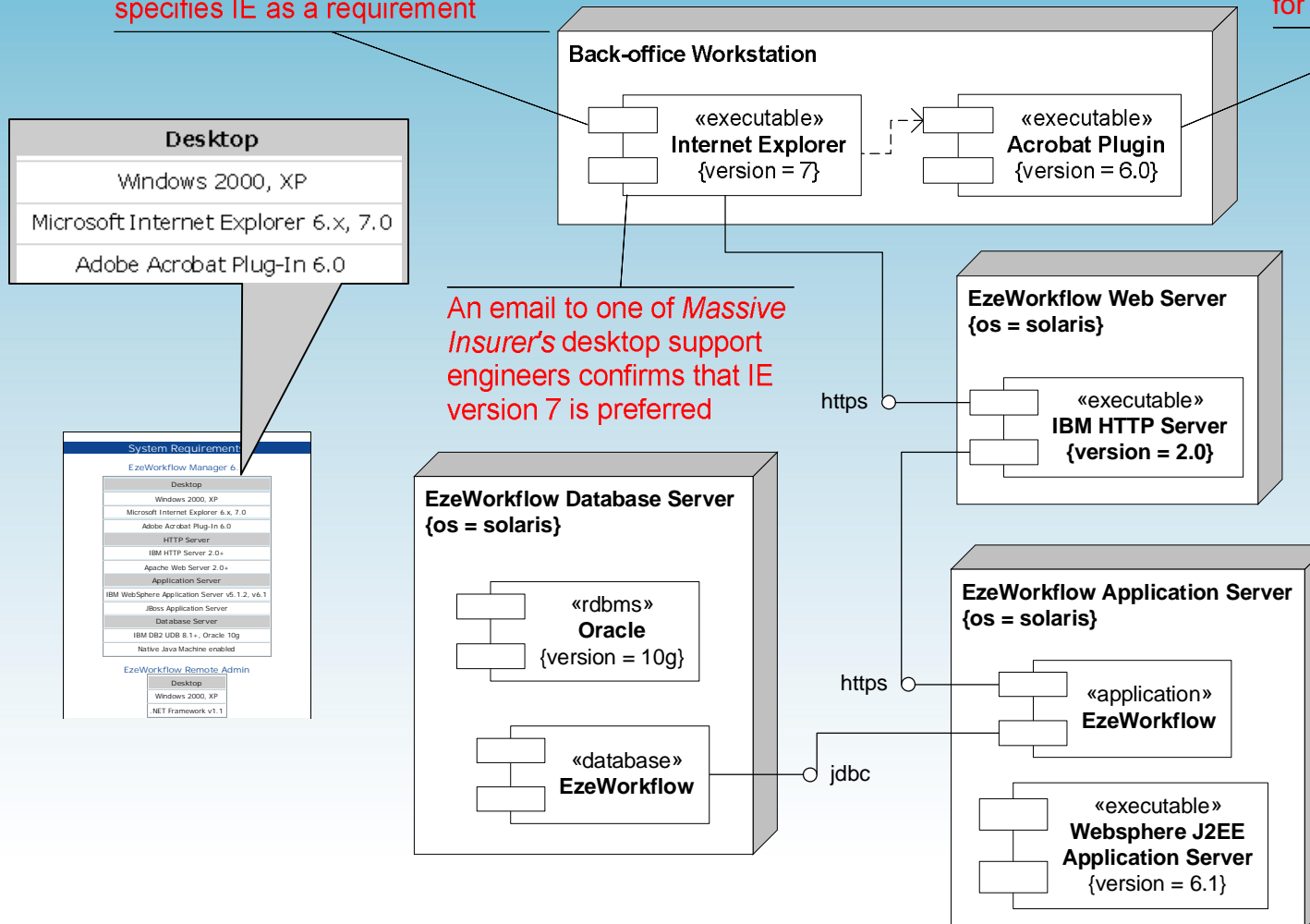
For now, we assume a single «database». Actual physical component names don't matter now – make an assumption.

- Database service offered by the «database» not «rdbms». The reason why would be apparent when more than one database is present on a node.
- JDBC is a safe assumption for a J2EE-based application – log an open design issue if confirmation is required.

Add Client Node Detail

Would be better to stay generic – “Web Browser”, but EzeWorkflow specifies IE as a requirement

Dependency duly noted. Version included for good measure.



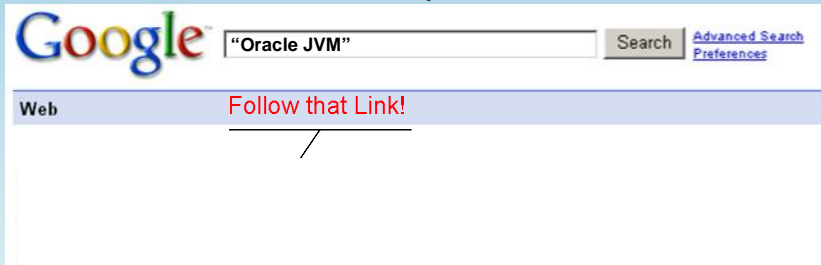
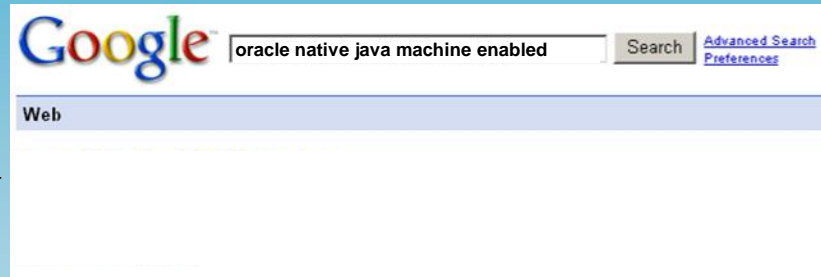
Input #2 – Google Sleuthing

“Oracle JVM”
Looks Promising

System Requirements	
EzeWorkflow Manager 6.1	
Desktop	
Windows 2000, XP	
Microsoft Internet Explorer 6.x, 7.0	
Adobe Acrobat Plug-In 6.0	
HTTP Server	
IBM HTTP Server 2.0+	
Apache Web Server 2.0+	
Application Server	
IBM WebSphere Application Server v5.1.2, v6.1	
JBoss Application Server	
Database Server	
IBM DB2 UDB 8.1+, Oracle 10g	
Native Java Machine enabled	
EzeWorkflow Remote Admin	
Desktop	
Windows 2000, XP	
.NET Framework v1.1	

Database Server

IBM DB2 UDB 8.1+, Oracle 10g
Native Java Machine enabled



Java Installation and Configuration

This chapter describes what you need to know to install and configure Oracle JVM within your database. To configure Java memory, see the "[Java Memory Usage](#)" section in [Chapter 6, "Oracle9i Java Application Performance"](#).

- [Initializing a Java-Enabled Database](#)
- [Configuring Oracle JVM](#)
- [Using The DBMS JAVA Package](#)
- [Enabling the Java Client](#)

Getting Warmer:
Looks like Oracle JVM is
a dependent library/plugin

The Smoking Gun:
Oracle JVM is
"enabled"

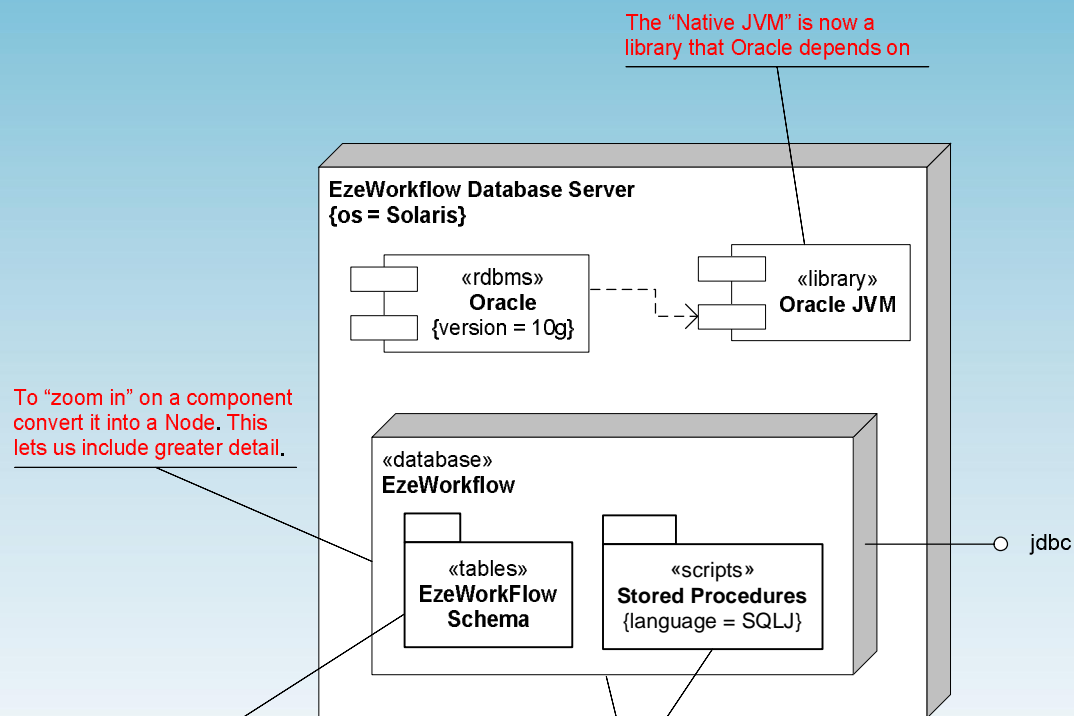
Initializing a Java-Enabled Database

If you install Oracle9i with the Oracle JVM option, the database is Java-enabled. That is, it is ready to run Java stored procedures, JDBC, and SQLJ.

Case Study (continued)

We can now infer that
Java Stored Procedures
on the database tier are a
key part of EzeWorkflow.

Zoom in on Database



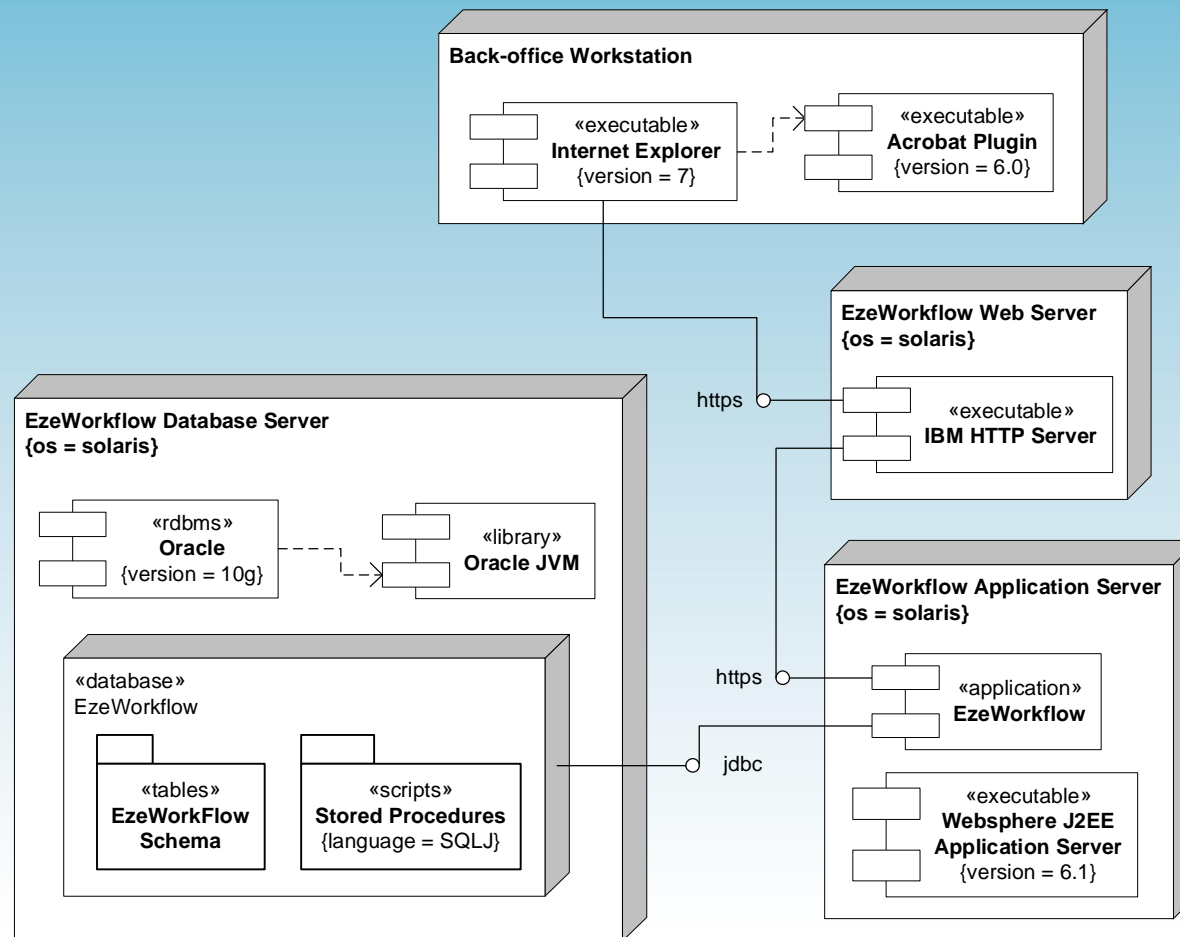
Depict the main body of schema as a package of «tables».

Our Googling uncovered that the Oracle JVM is required for Java Stored Procedures in the DB. We:

1. Assume the presence of scripts in the database
2. "Zoom in" on the database by converting it from a Component (see *Figure 1.5*) to a Node
3. Call them out as a package of «scripts», most likely in the SQLJ language

Case Study (continued)

Final Product



Case Study
(continued)

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. 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.