X

Ready, Set, Architect!

300* seconds to free architecture

DISCLAIMER

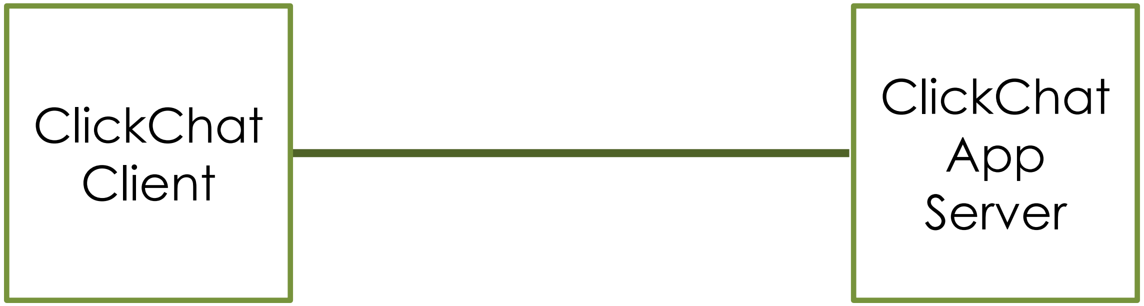
I, Ronald L. Parker, aka @scmunk, make no claims, promises, or guarantees about the accuracy, completeness, or adequacy of the contents of this presentation or the words that are used during this presentation and expressly disclaims liability for errors and omissions in any content while I also claim that this presentation isn't in any way related to my employer, or employers no matter how similar the content may seem, sound, or otherwise how it may have any molecular resemblance to anything related to any other entity with that being said everything related to this session is completely based on my opinion and right about now is when you insert some random Latin filler phrases to make it look like there is more verbiage than there actually is although no one really cares at the end of the day.

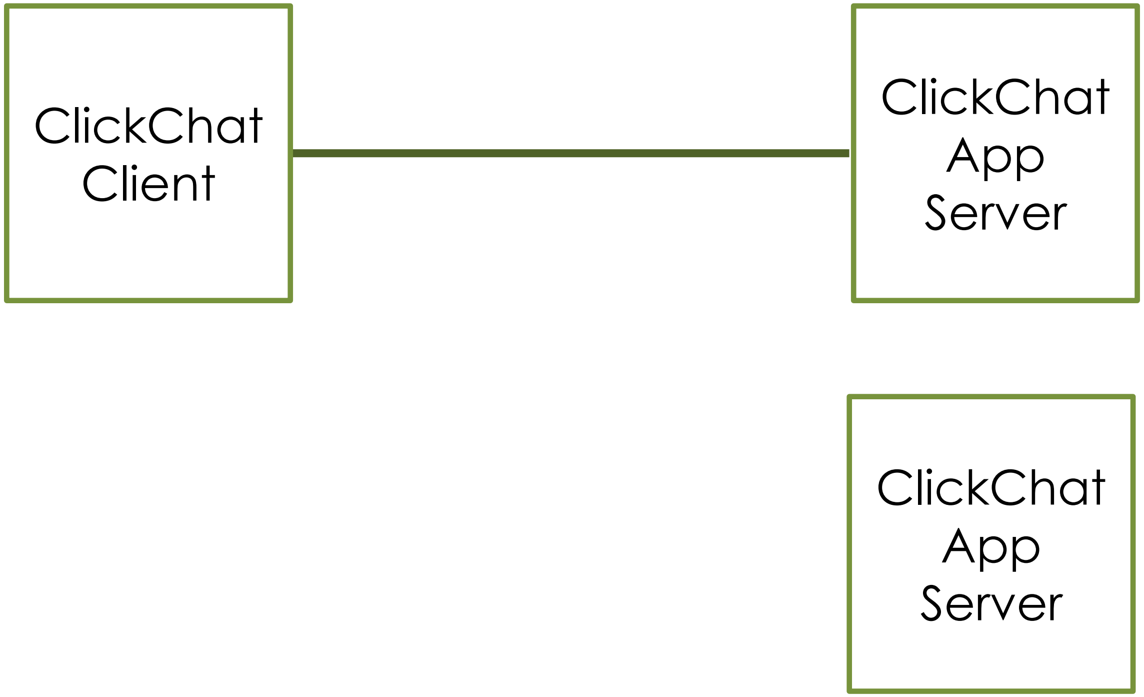
Agenda

Why do we have architecture and what does it look like?

What are some approaches or frameworks for architecture?

What tools can be used?

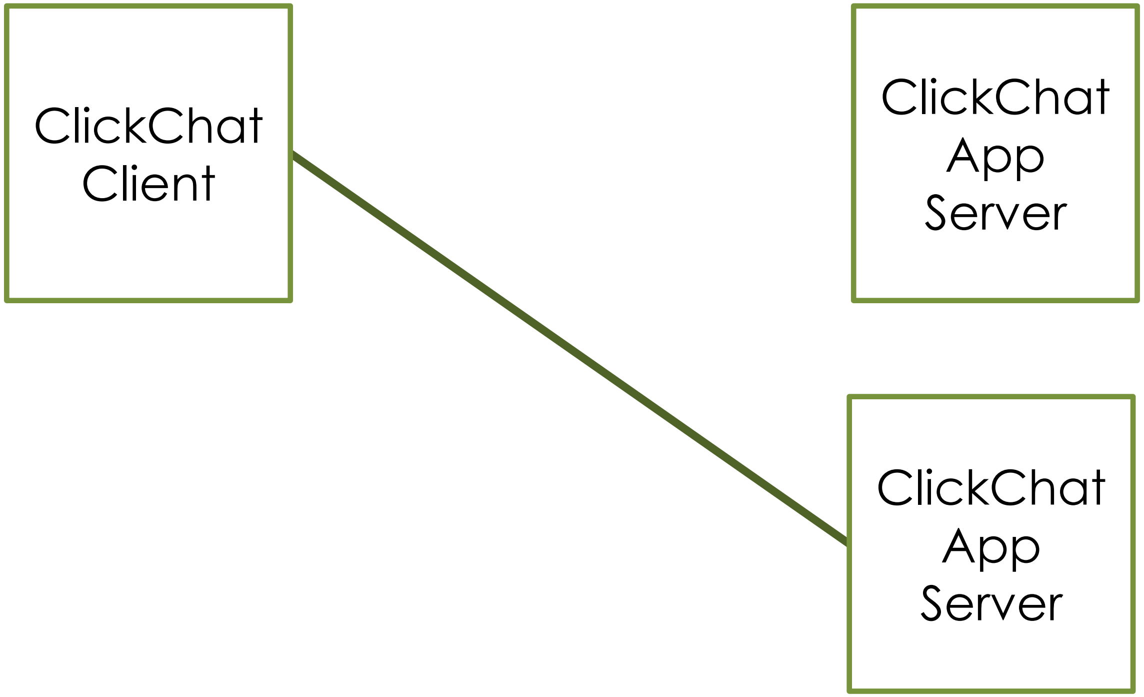

ClickChat Client
ClickChat App Server

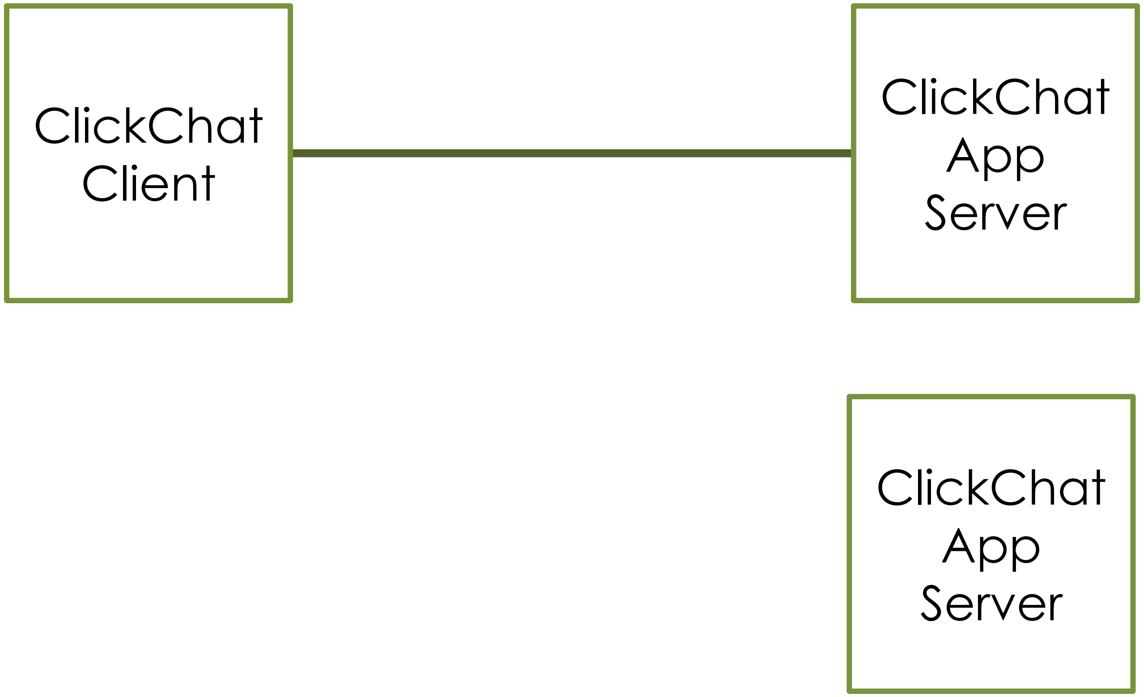


ClickChat Client

ClickChat App Server

ClickChat App Server

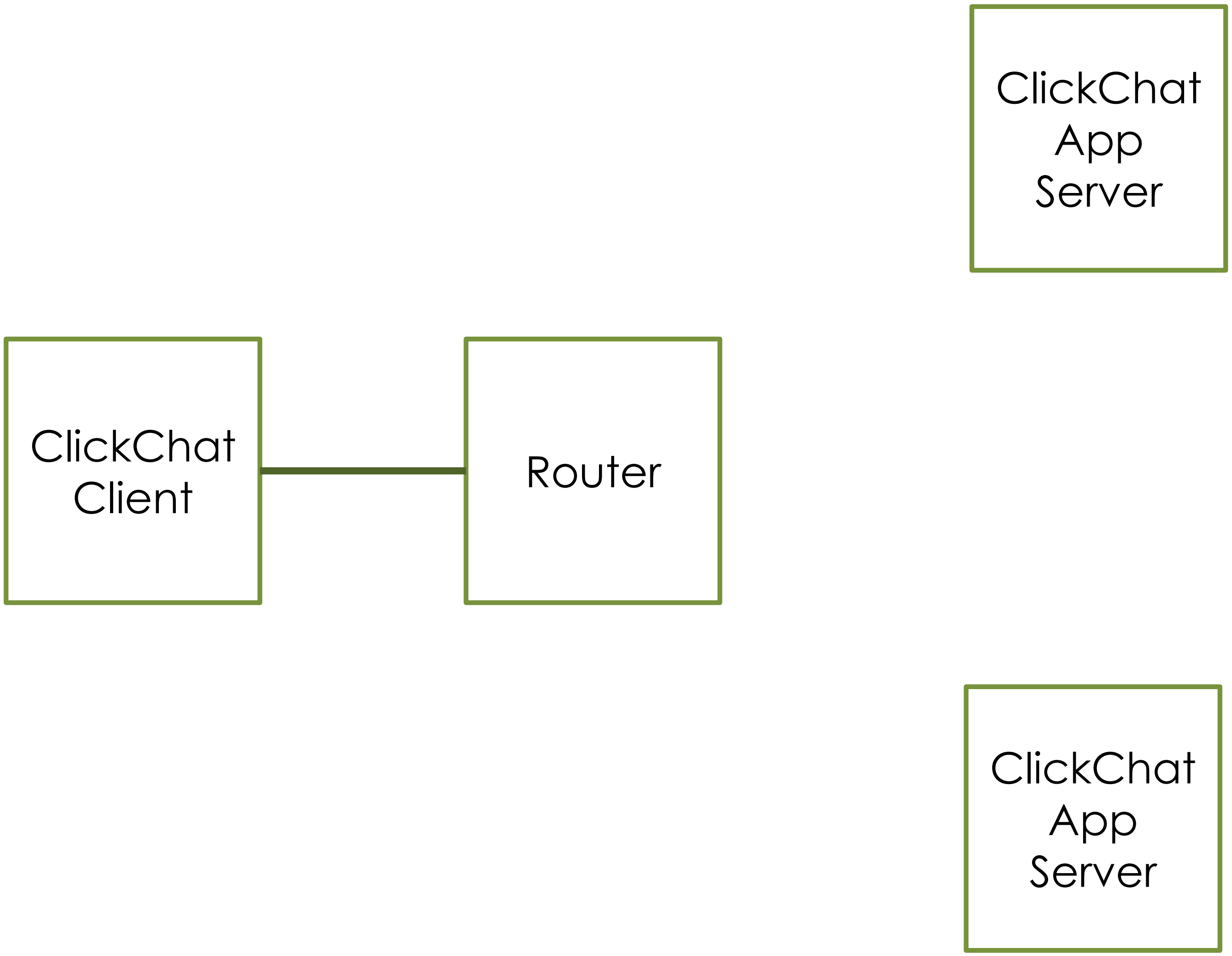

ClickChat Client
ClickChat App Server
ClickChat App Server

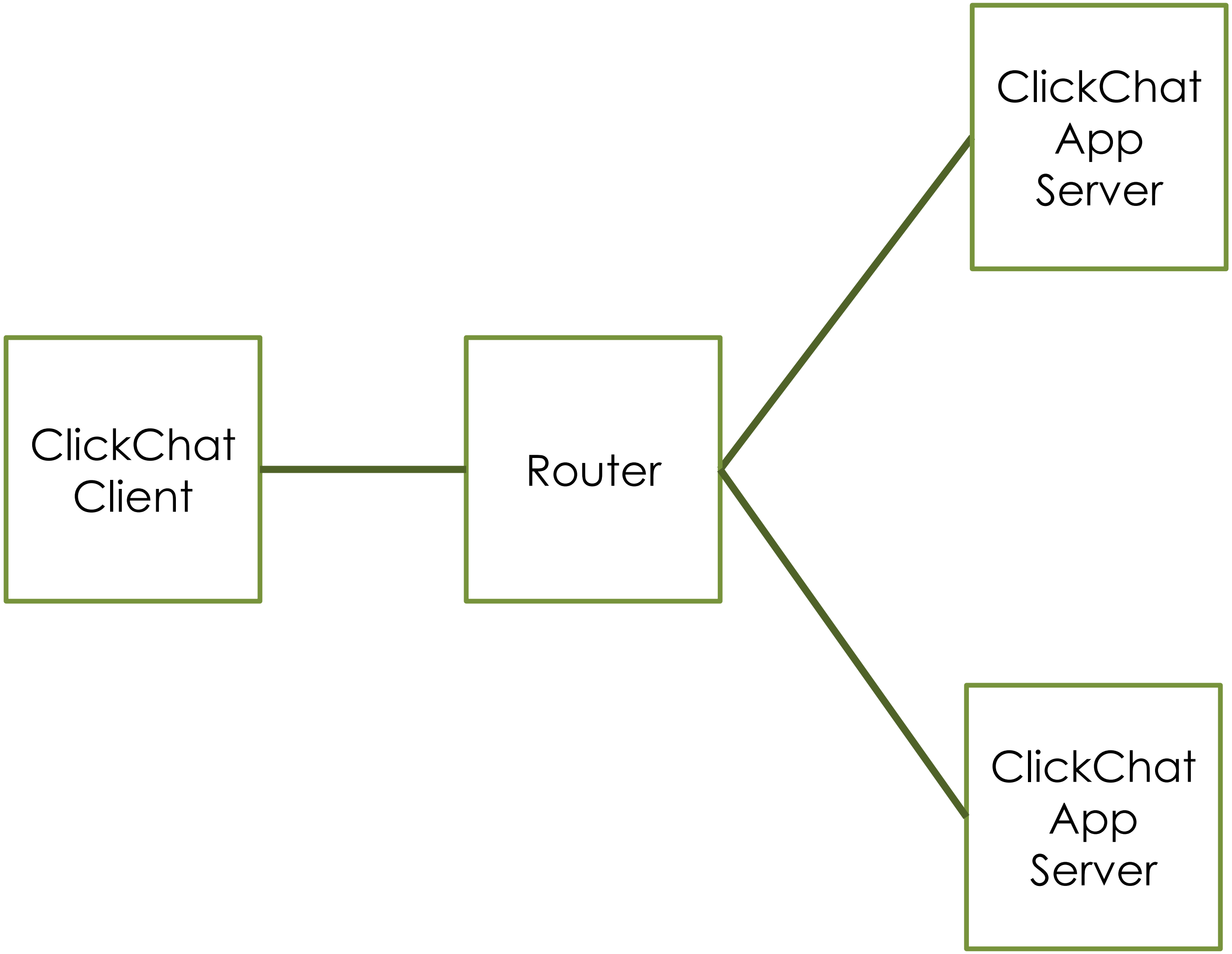


ClickChat Client

ClickChat App Server

ClickChat App Server


ClickChat App Server
ClickChat Client
Router
ClickChat App Server

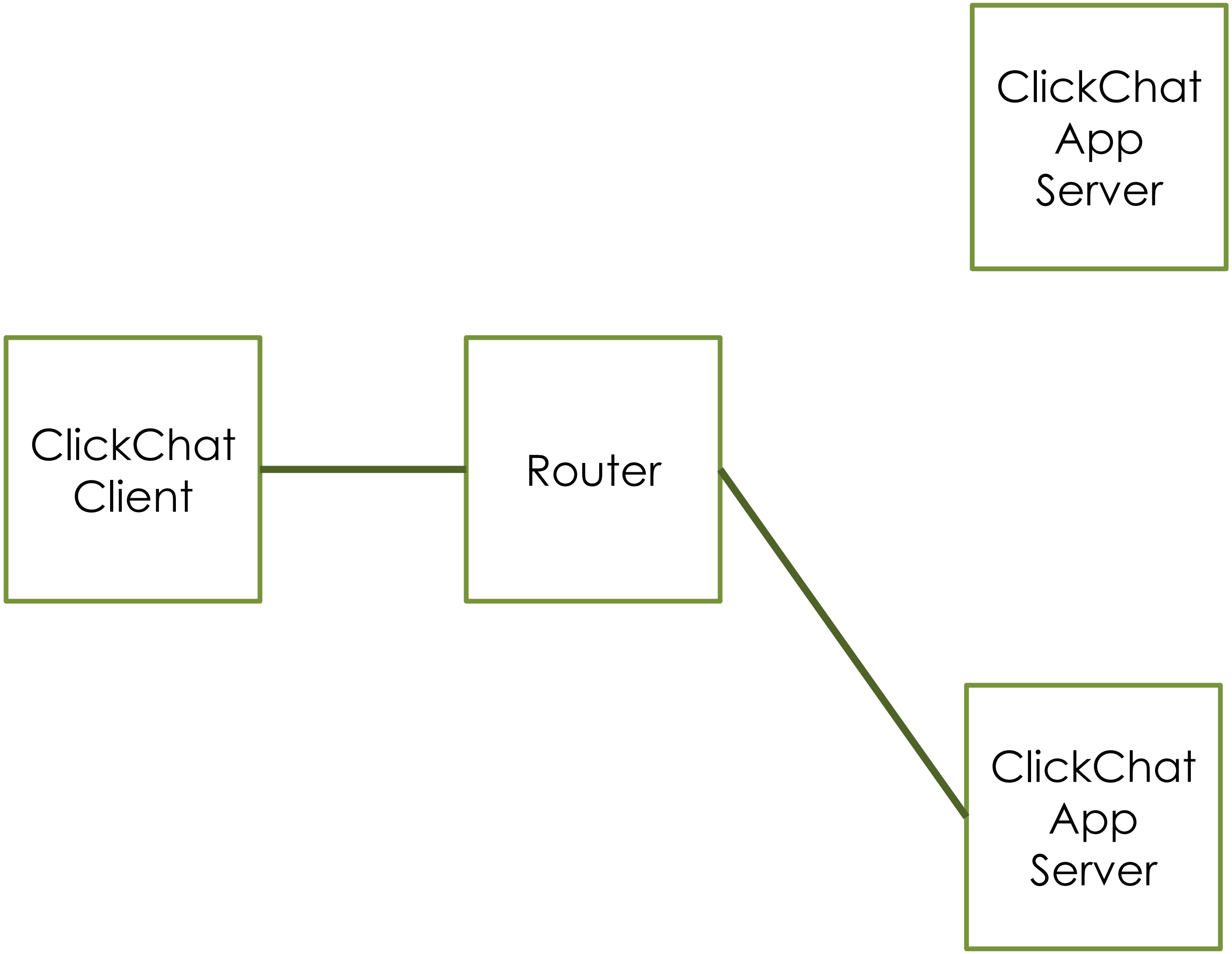


ClickChat Client

Router

ClickChat App Server

ClickChat App Server

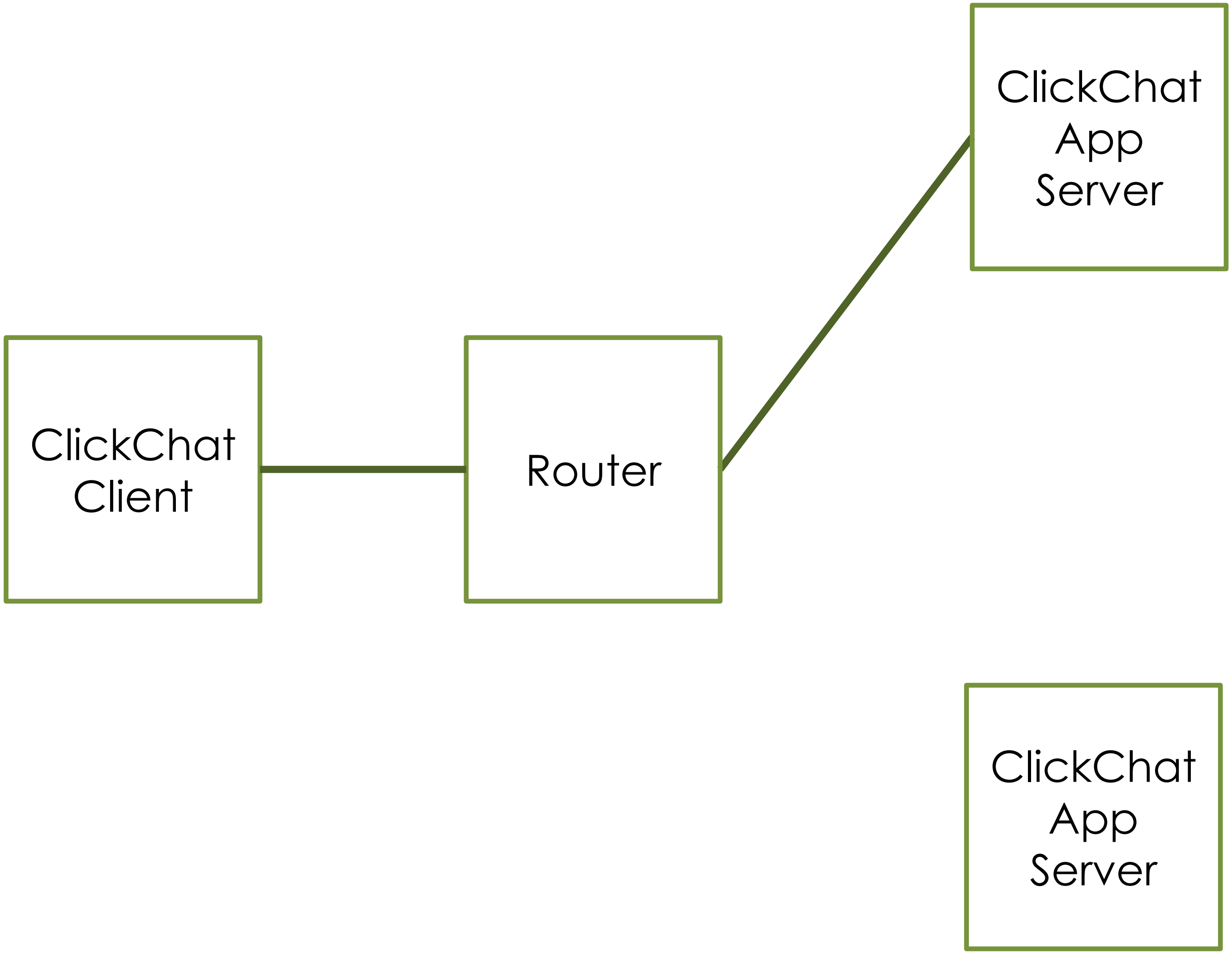


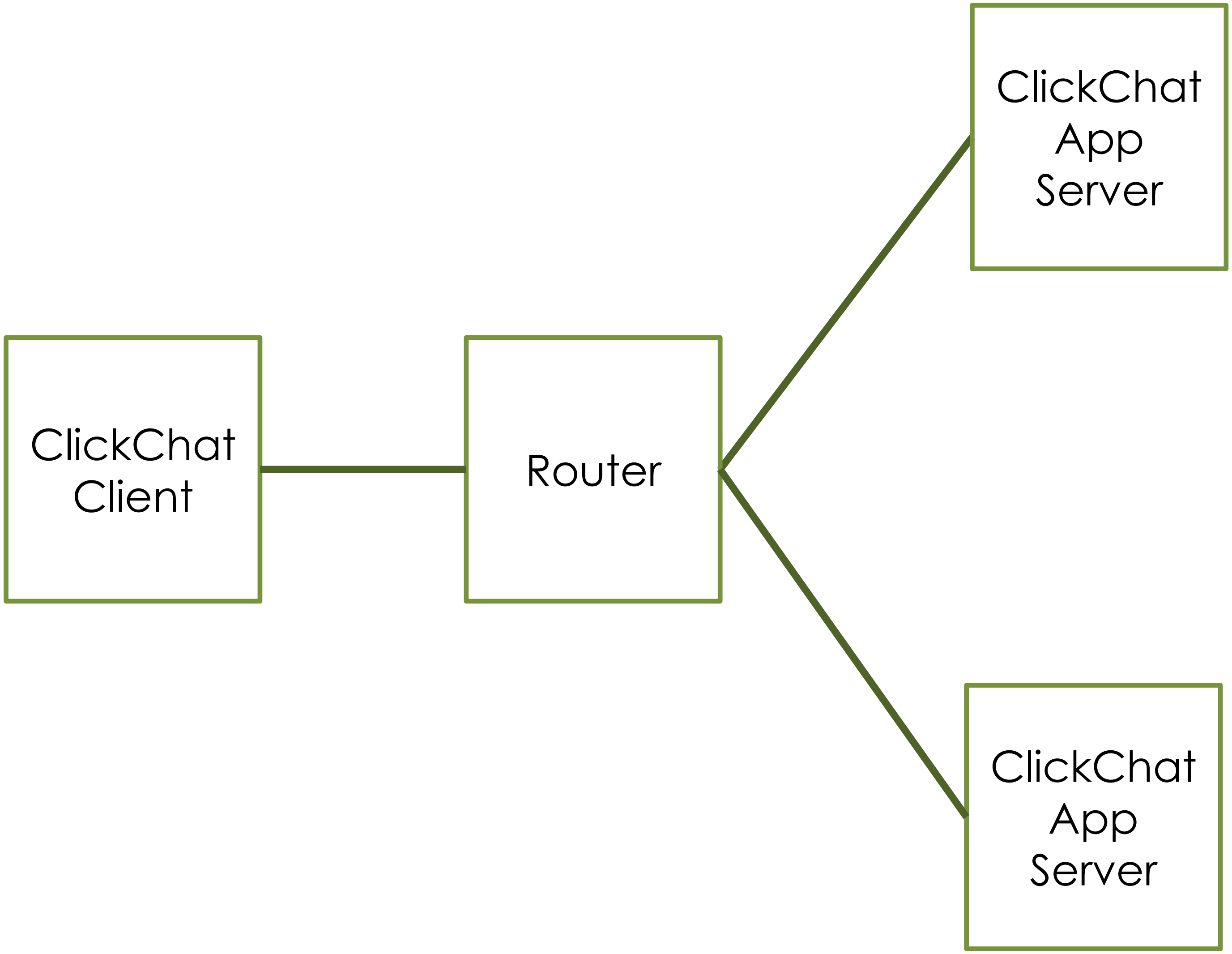
ClickChat App Server

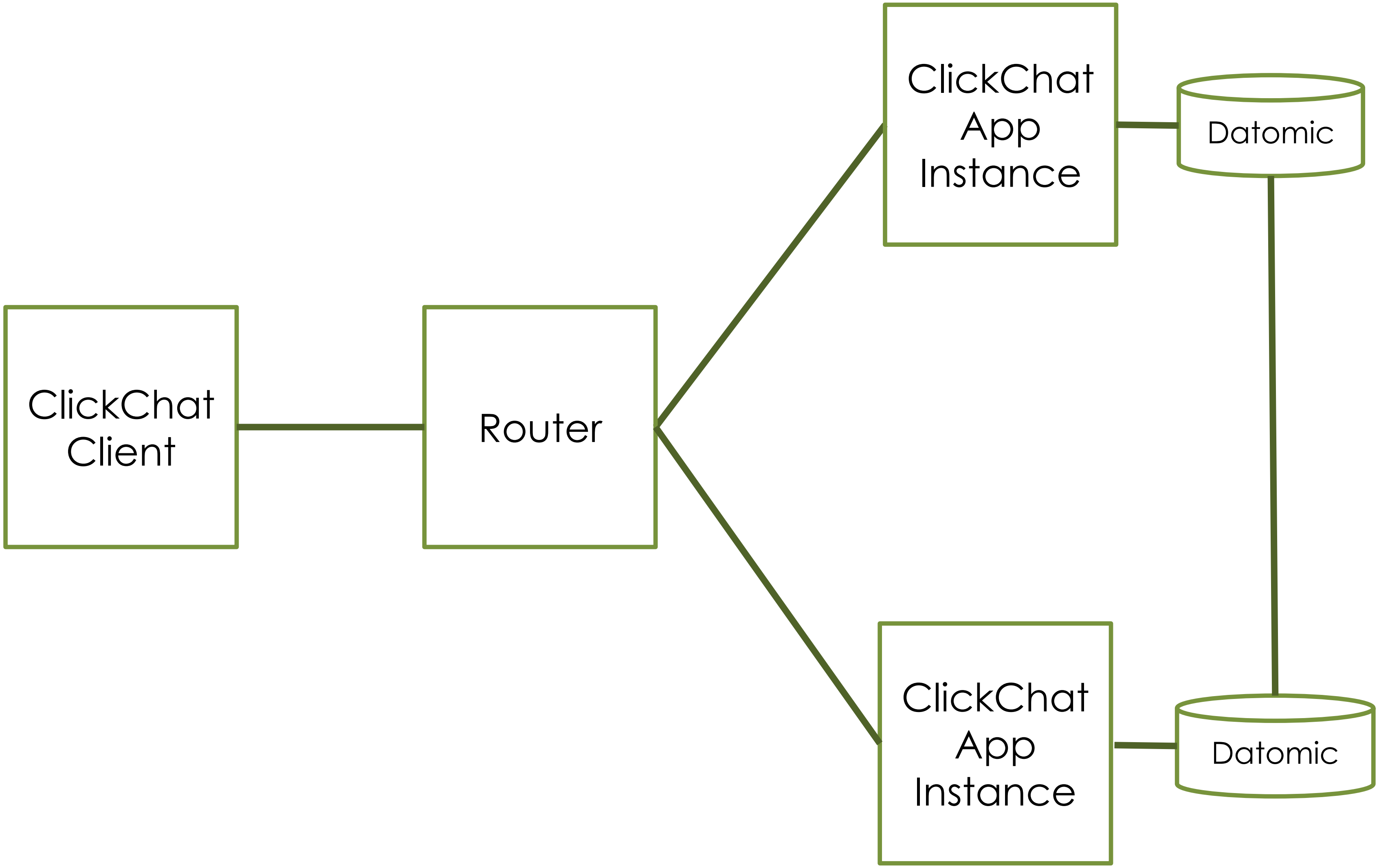
ClickChat Client

Router

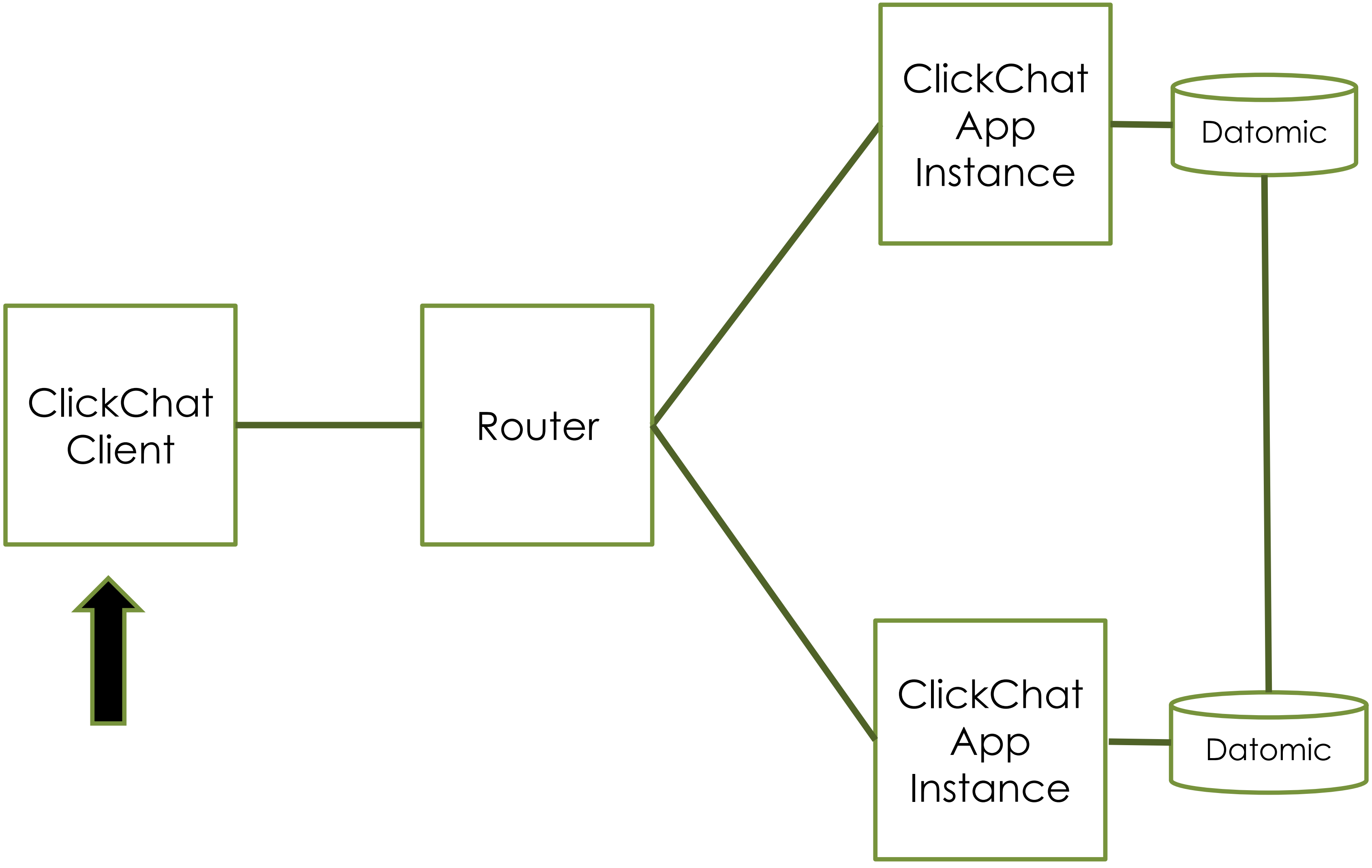
ClickChat App Server


ClickChat App Server
ClickChat Client
Router
ClickChat App Server

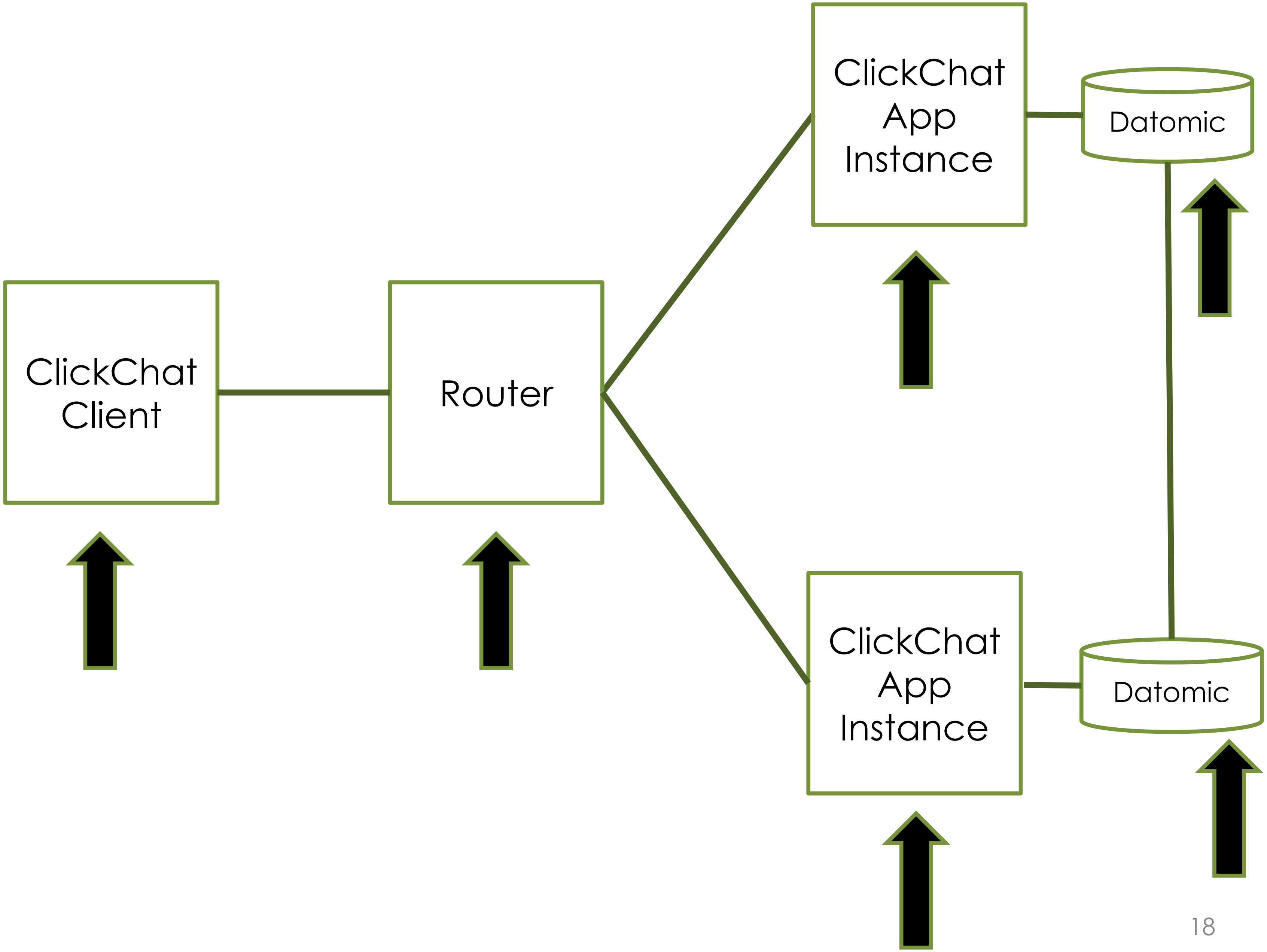

ClickChat Client
Router
ClickChat App Server
ClickChat App Server


ClickChat Client
Router
ClickChat App Instance
Datomic
ClickChat App Instance
Datomic

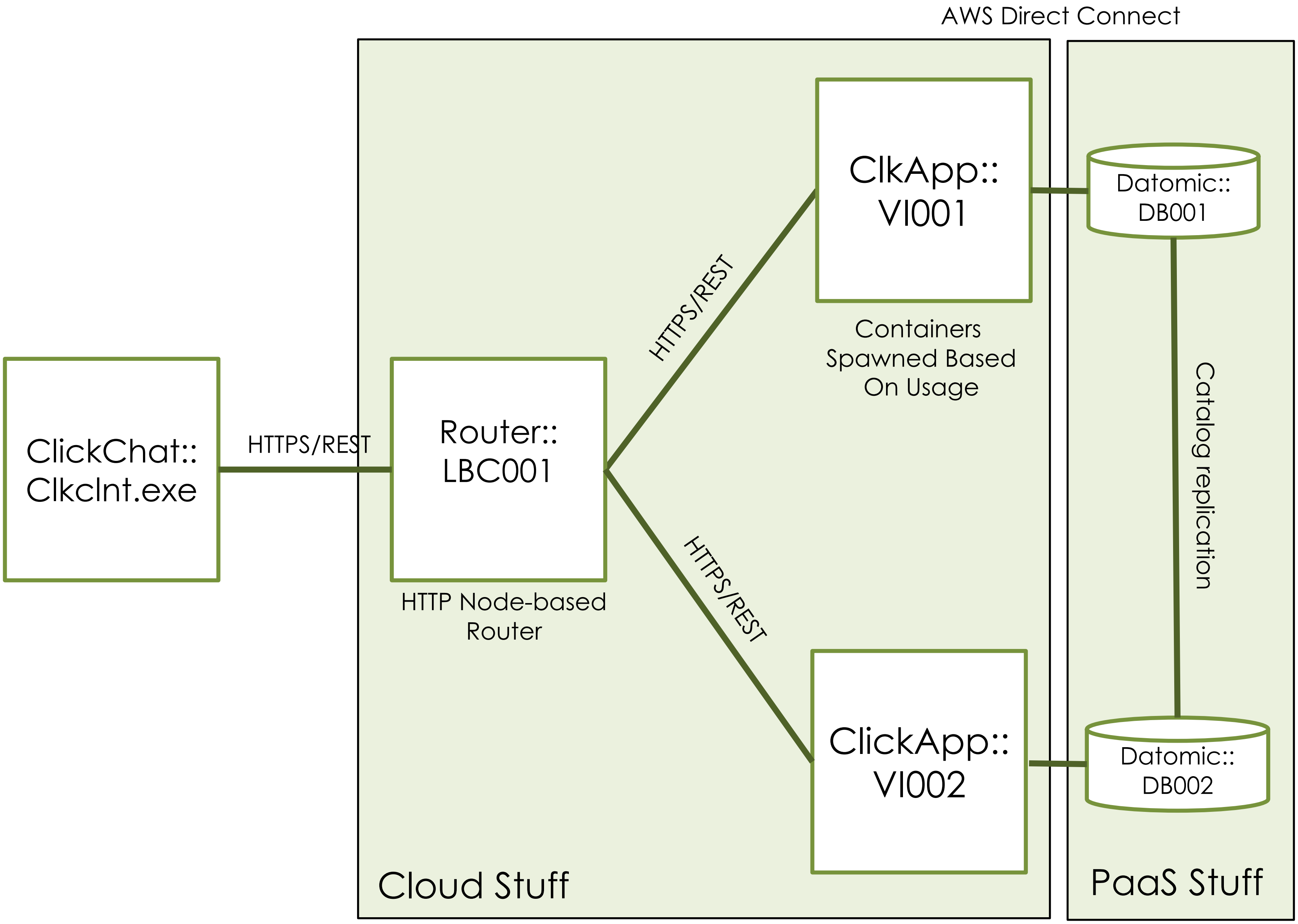
What did we just do?



How did we approach system resiliency?



We architect to **optimize**
the **whole**,
not just the pieces.



AWS Direct Connect

ClickChat::
ClkcInt.exe

HTTPS/REST

Router::
LBC001

HTTP Node-based
Router

HTTPS/REST

ClkApp::
VI001

Containers
Spawned Based
On Usage

HTTPS/REST

ClickApp::
VI002

Datomic::
DB001

Catalog replication

Datomic::
DB002

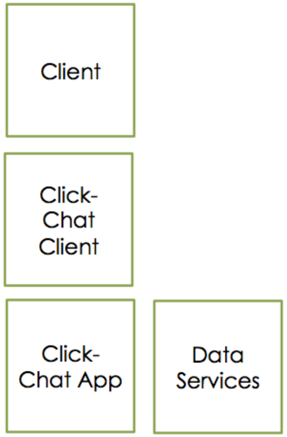
Cloud Stuff

PaaS Stuff

VIEWS

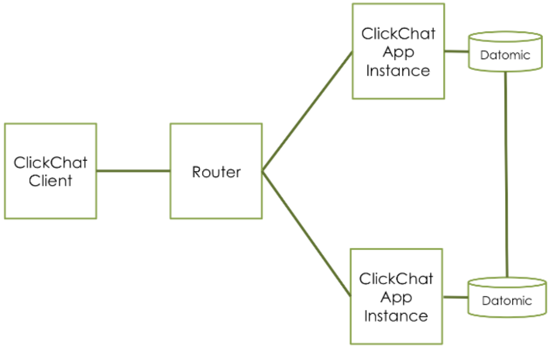
Different views for different audiences – Same Systems

Conceptual



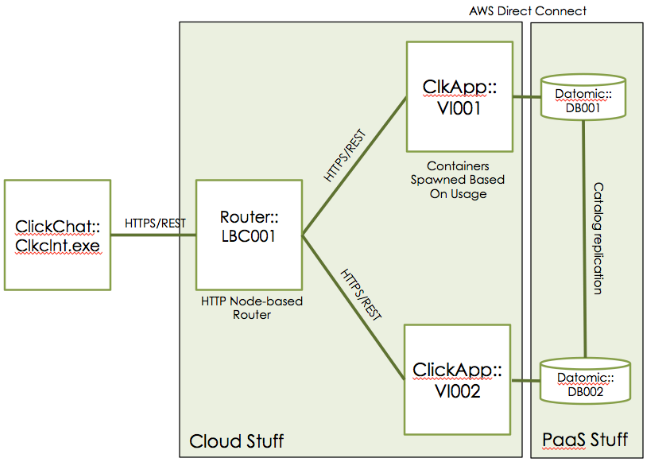
- More ideas
- Planning
- Management
- Gaps

Logical



- Relations
- Behavior
- Components
- Functions
- Integration
- Solutioning

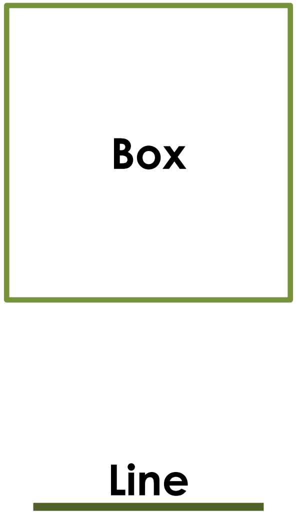
Physical



- Implementation
- Operations
- Solutioning

More Concrete

Galactic Modeling Language



Box

Line

Label

Label

GML

A small, cross-platform, but still robust approach for describing architecture

All Down-Hill From Here

- Understand WHY
- Understand your AUDIENCE and the VIEWS

C4 Hierarchical Model

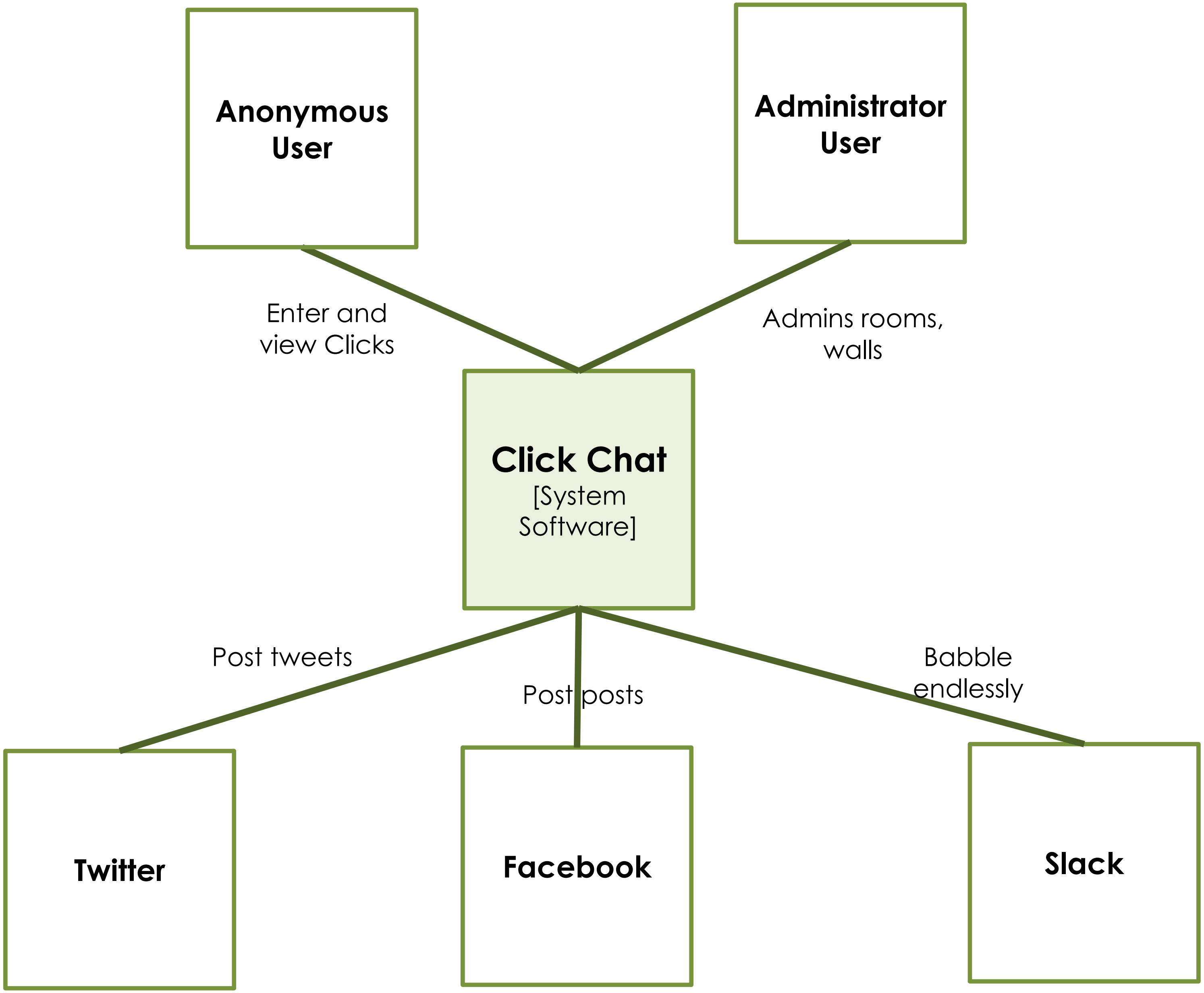
A little more specific than GML

Good for software architecture

Nested items

- **C**ontext
- **C**ontainers
- **C**omponents
- **Cl**asses or Code

C4 – Context (of the system)



Anonymous User

Administrator User

Enter and view Clicks

Admins rooms, walls

Click Chat
[System Software]

Post tweets

Post posts

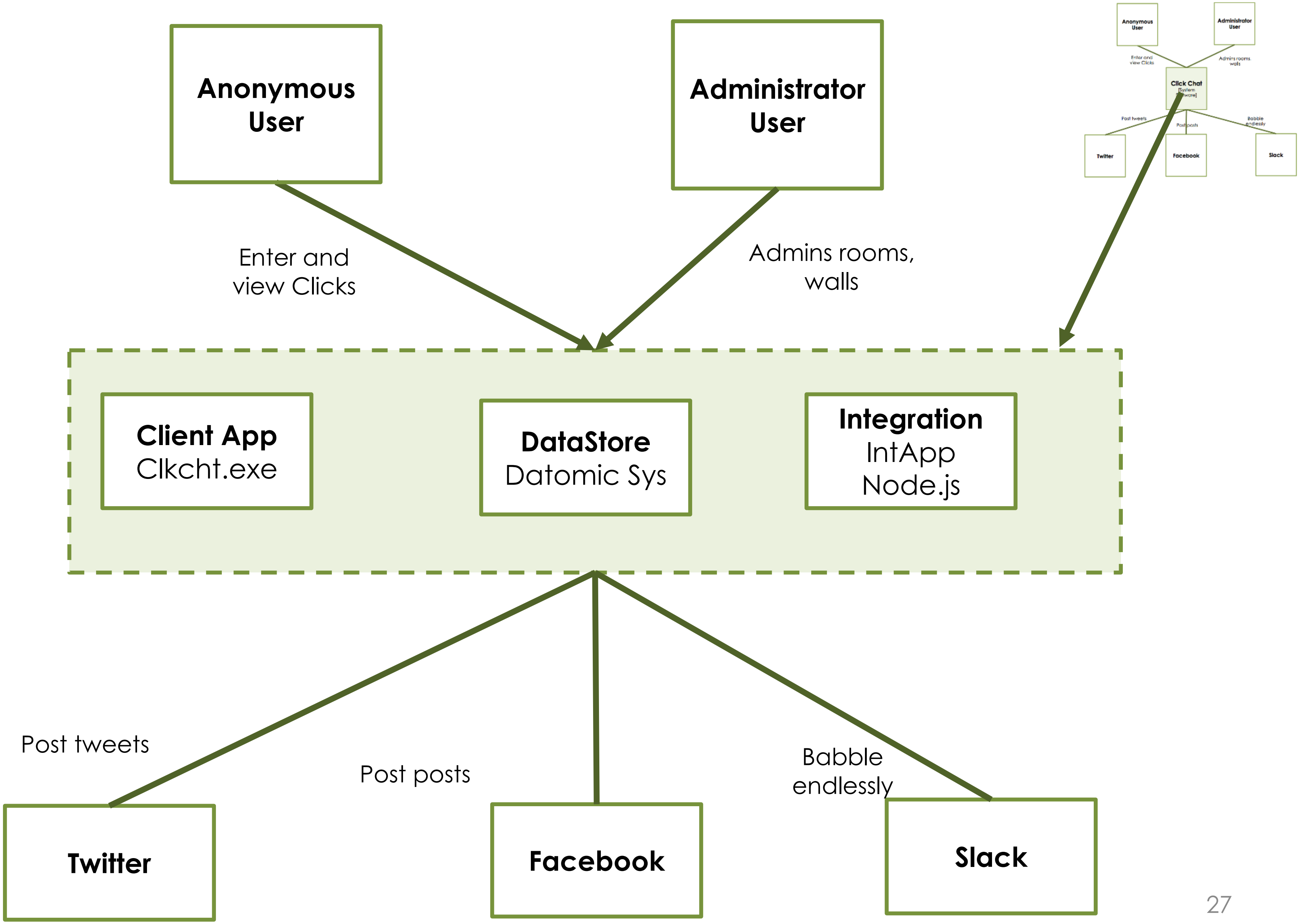
Babble endlessly

Twitter

Facebook

Slack

C4 - Container



Anonymous User

Administrator User

Enter and view Clicks

Admins rooms, walls

Client App Clkcht.exe

DataStore Datomic Sys

Integration IntApp Node.js

Post tweets

Post posts

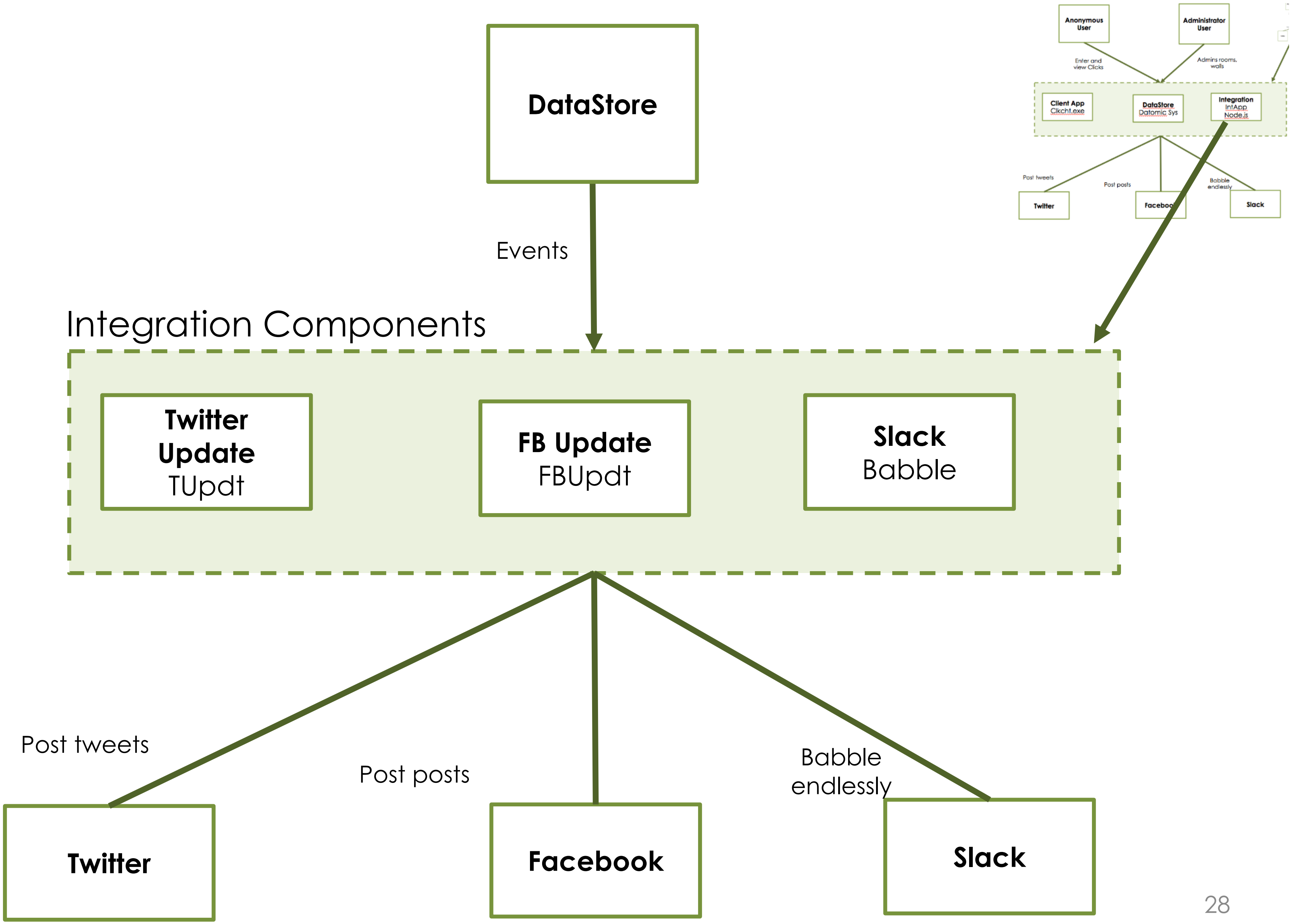
Babble endlessly

Twitter

Facebook

Slack

C4 - Component



C4 – Classes or Code

More boxes for Classes

Code

C4

Lightweight software architecture

Based on views of the pieces and usage

UML – Unified Modeling Language

UML – Unified Modeling Language

UML

Three Rules for Successful UML'ing

1. Pick out the good stuff
2. Only use the good stuff
3. Ignore anything that is not good stuff

Structural UML diagrams

- Class diagram
- Package diagram
- Object diagram
- Component diagram
- Composite structure diagram
- Deployment diagram

Behavioral UML diagrams

- Activity diagram
- Sequence diagram
- Use case diagram
- State diagram
- Communication diagram
- Interaction overview diagram
- Timing diagram

Three of my favorites

1. Class Diagrams
2. Sequence Diagrams
3. Component Diagrams

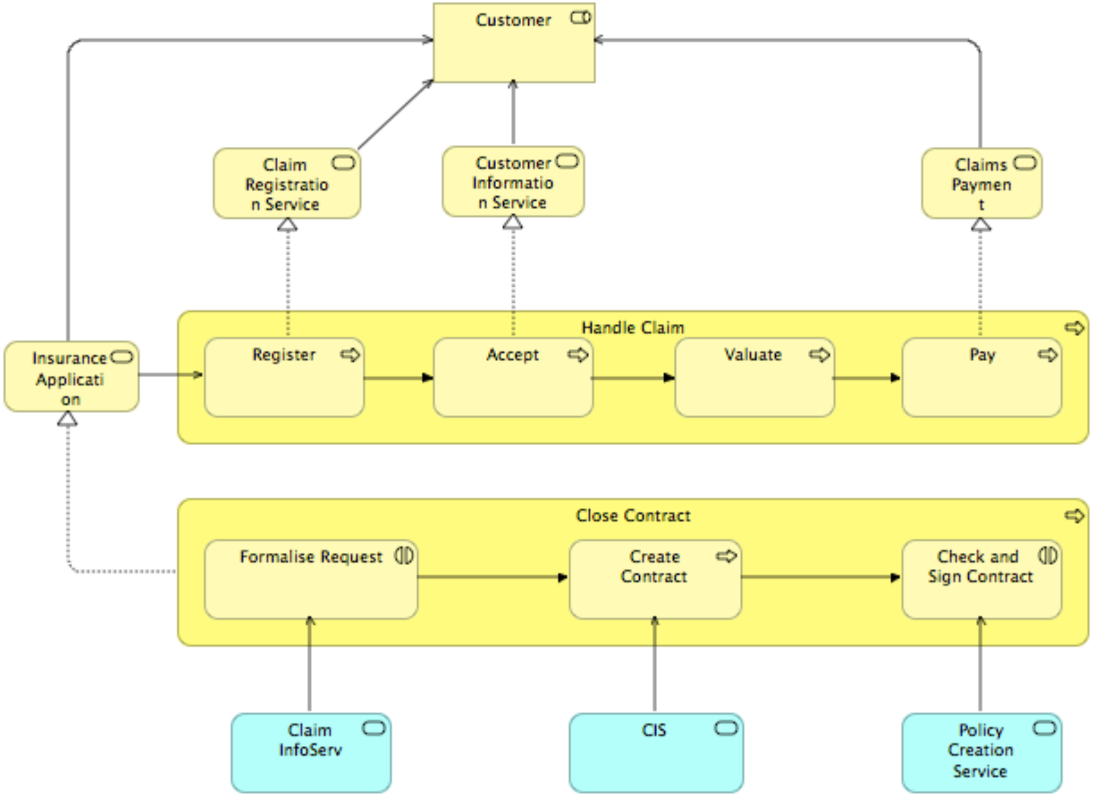
Honorable mention to Use Case diagrams but User Stories are better

Archimate 3.0 (AR-ki-mayt)

Built by The Open Group

Strength is in connected views

Not for low level software modeling

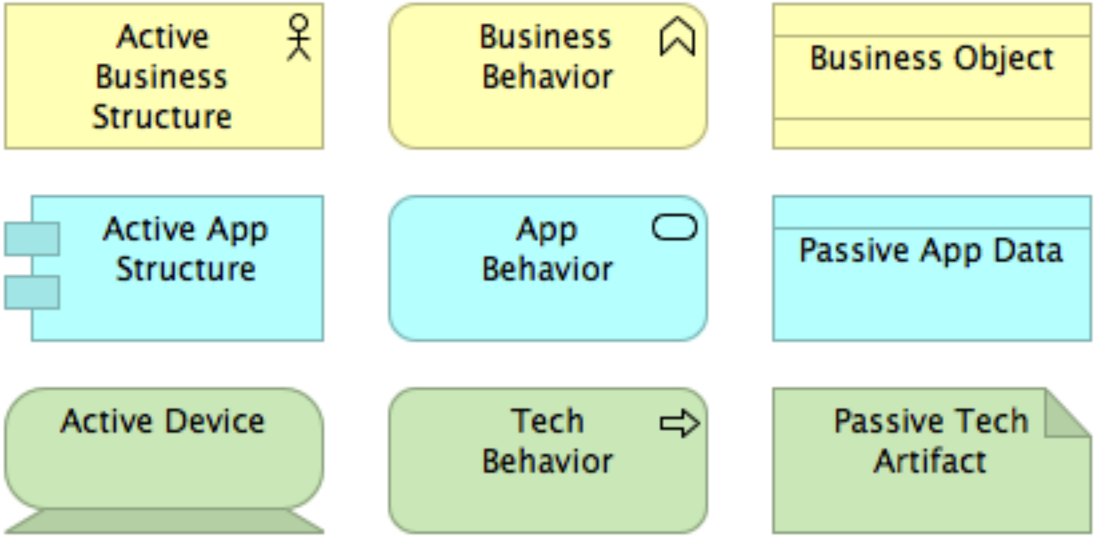


Archimate Elements

Notation Types

- Motivation
- Strategy
- Business
- Application
- Technology
- Physical
- Implementation

Example Elements



Can be used for high level architecture and solution architecture because there is an emphasis on modeling behavior separately from implementation.

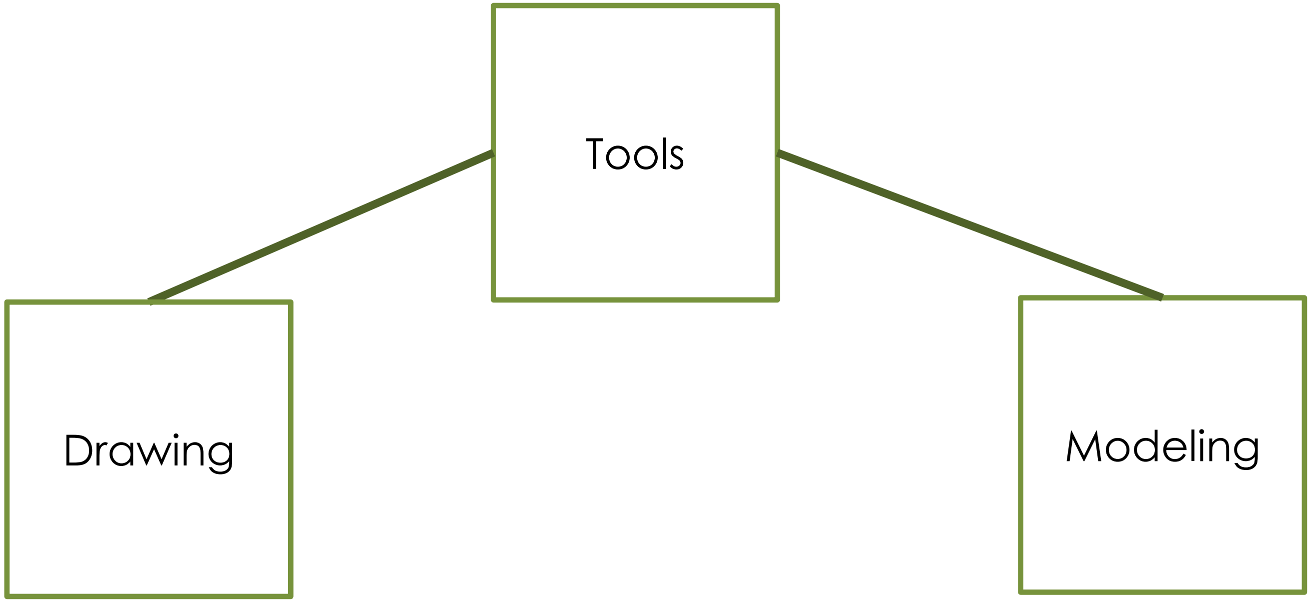
Frameworks

Four Examples

- GML
- UML
- C4
- Archimate

Now we need tools....

Tools



- No repository of elements
- Each diagram is standalone
- Very easy and portable
- Think Paper or Whiteboard

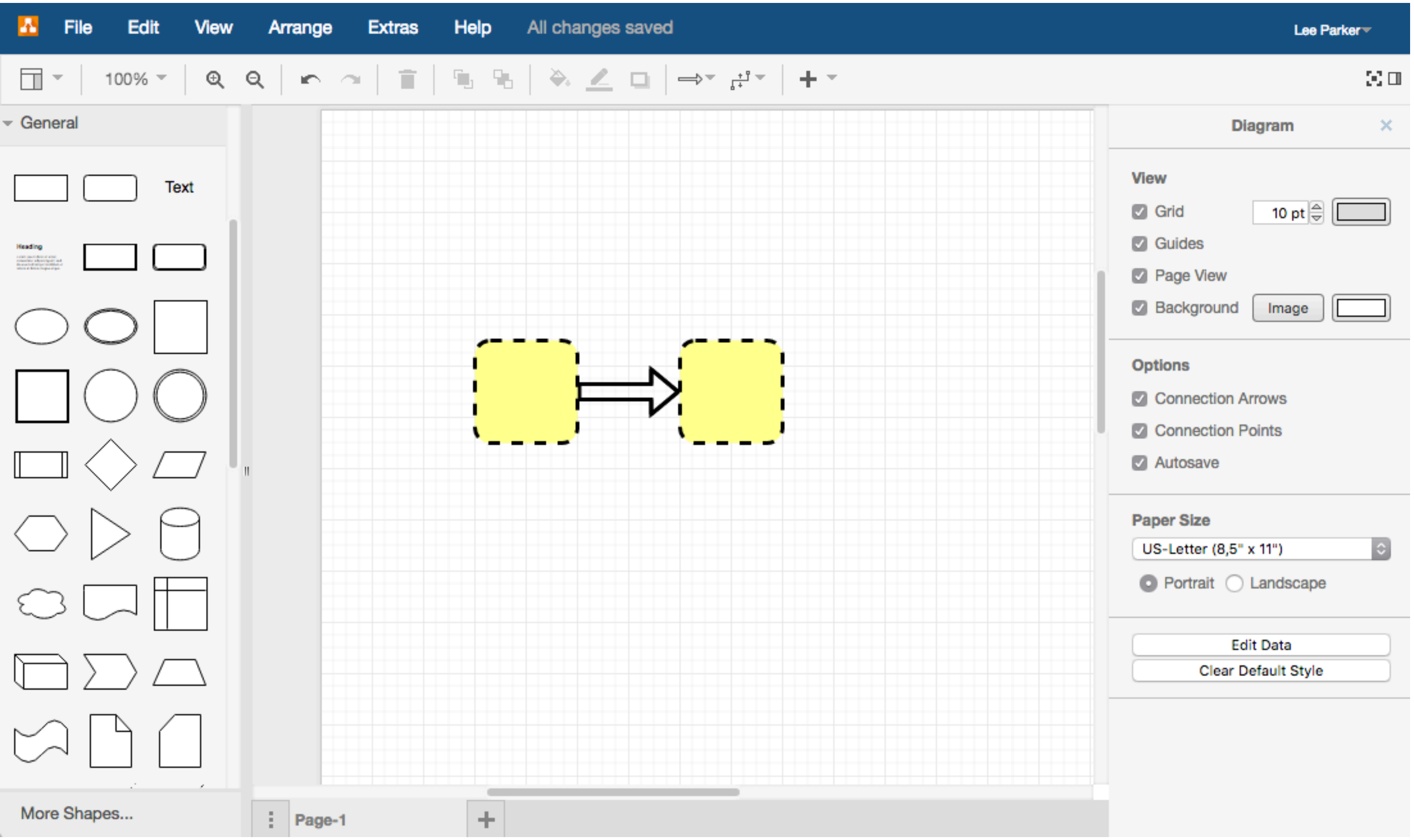
- Repository of elements
- Tied to a framework
- Lots of reuse
- Maybe too many rules
- Diagrams relate

Drawing: Paper and Pencil

Using whatever is at hand to capture an idea is better than not capturing the idea at all.

- *@scmunk*

Drawing: Draw.io



View

- ☑ Grid 10 pt
- ☑ Guides
- ☑ Page View
- ☑ Background Image

Options

- ☑ Connection Arrows
- ☑ Connection Points
- ☑ Autosave

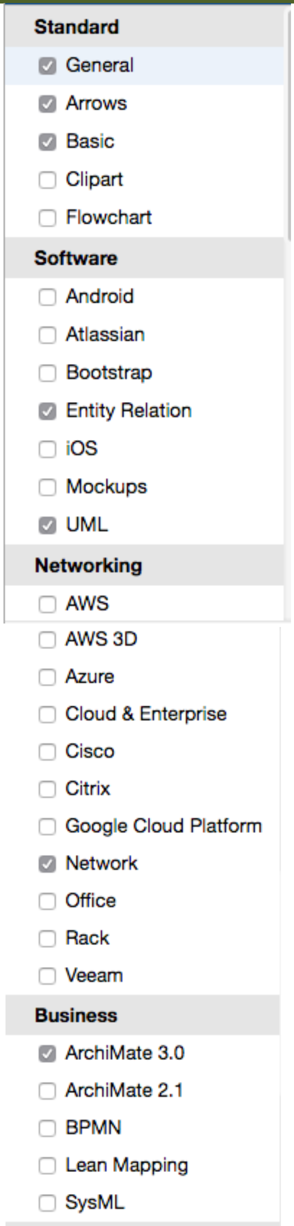
Paper Size

US-Letter (8,5" x 11")

● Portrait ○ Landscape

Edit Data

Clear Default Style



Standard

- ☑ General
- ☑ Arrows
- ☑ Basic
- ☐ Clipart
- ☐ Flowchart

Software

- ☐ Android
- ☐ Atlassian
- ☐ Bootstrap
- ☑ Entity Relation
- ☐ iOS
- ☐ Mockups
- ☑ UML

Networking

- ☐ AWS
- ☐ AWS 3D
- ☐ Azure
- ☐ Cloud & Enterprise
- ☐ Cisco
- ☐ Citrix
- ☐ Google Cloud Platform
- ☑ Network
- ☐ Office
- ☐ Rack
- ☐ Veeam

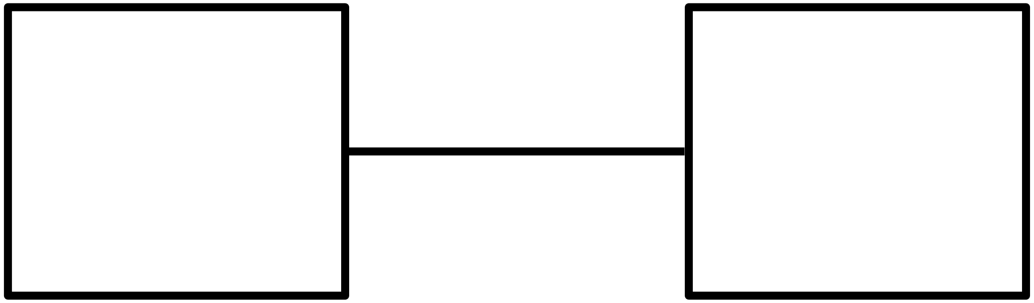
Business

- ☑ ArchiMate 3.0
- ☐ ArchiMate 2.1
- ☐ BPMN
- ☐ Lean Mapping
- ☐ SysML

- 100% cloud-based
- Connects to cloud storage or can use local
- Has shapes for UML, Archimate, AWS, Azure, UML…
- Save multiple diagrams per file
- Easily exports diagrams to images

Drawing: PowerPoint (Not free but maybe at hand)

Supports GML well

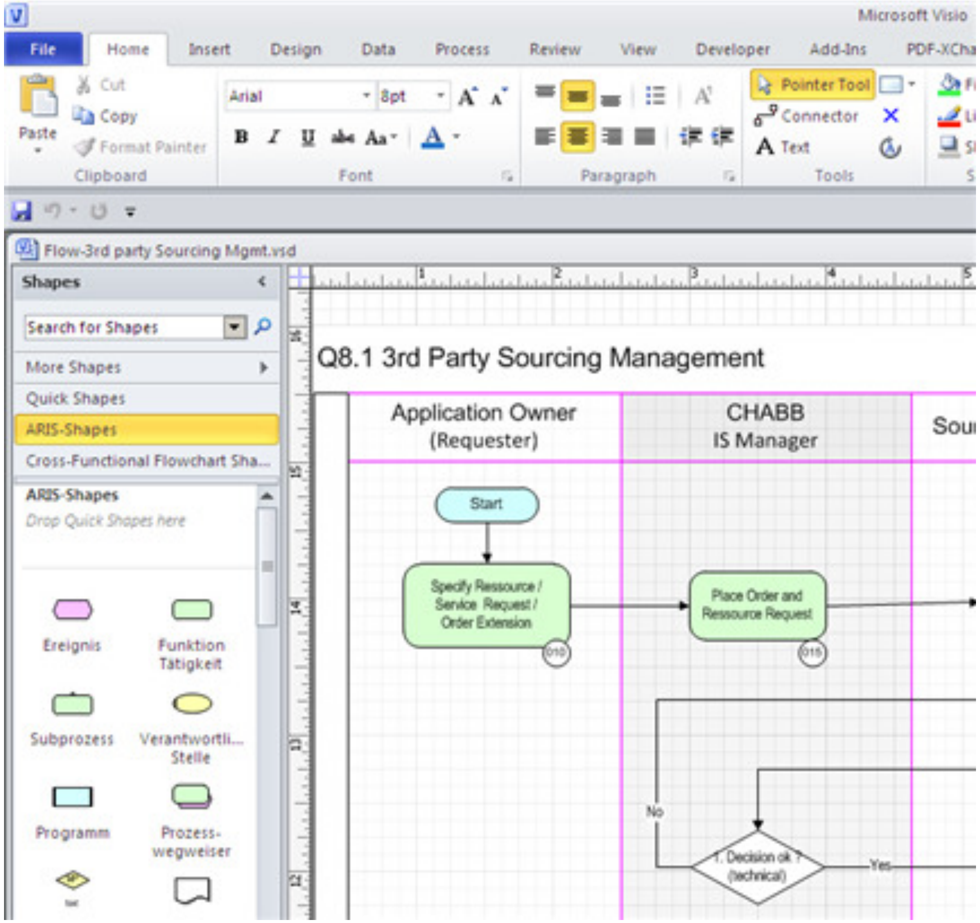


Easily Import Your Own Graphics

Active Business Structure

- Works well at the conceptual level
- Especially if you are working on associated material
- Graphics and templates are widely available but not built-in
- Exports to multiple formats

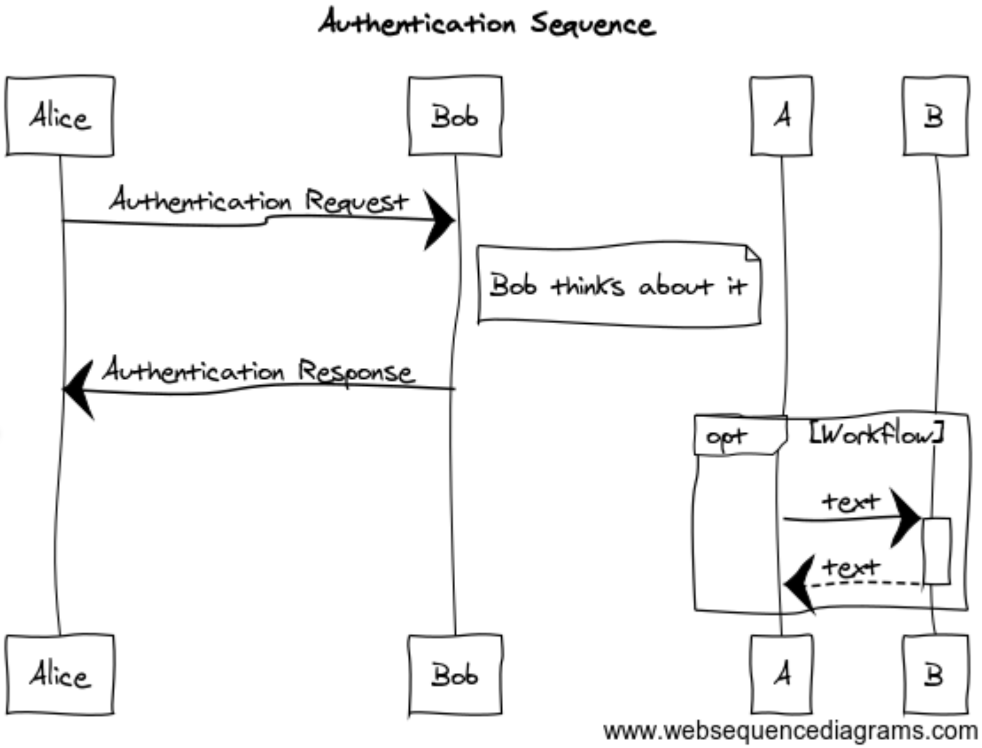
Drawing: Visio (Not free but maybe at hand)



- Top of the line drawing app
- Has stencils, free stencils, you can create your own
- Cloud options, tablet options, no Linux/OSX
- More features that you will probably never use
- Expensive

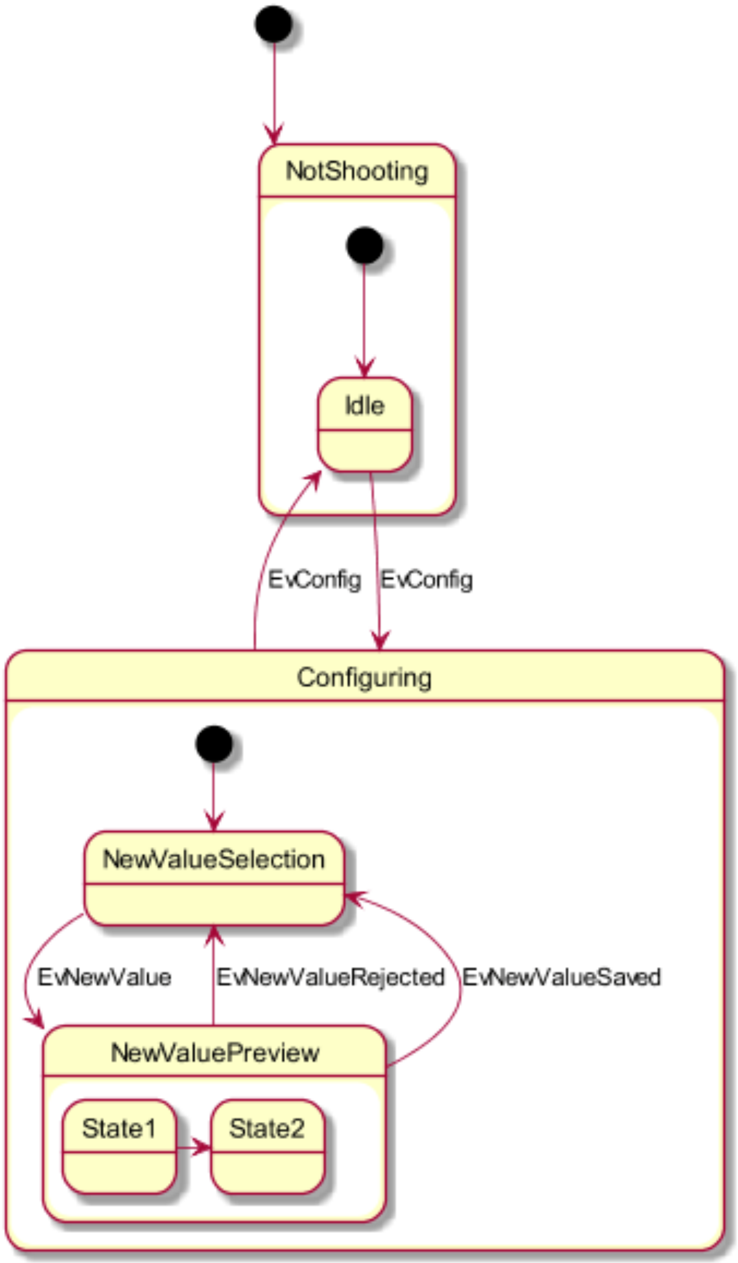
Drawing: WebSequenceDiagrams

```
title Authentication Sequence

Alice->Bob: Authentication Request
note right of Bob: Bob thinks
Bob->Alice: Authentication Response

opt Workflow
A->+B: text
B-->-A: text
end
```



- Easiest Sequence diagrams ever
- Can easily save you work by saving the text
- Has different styles for output

Drawing: Plant UML Generation

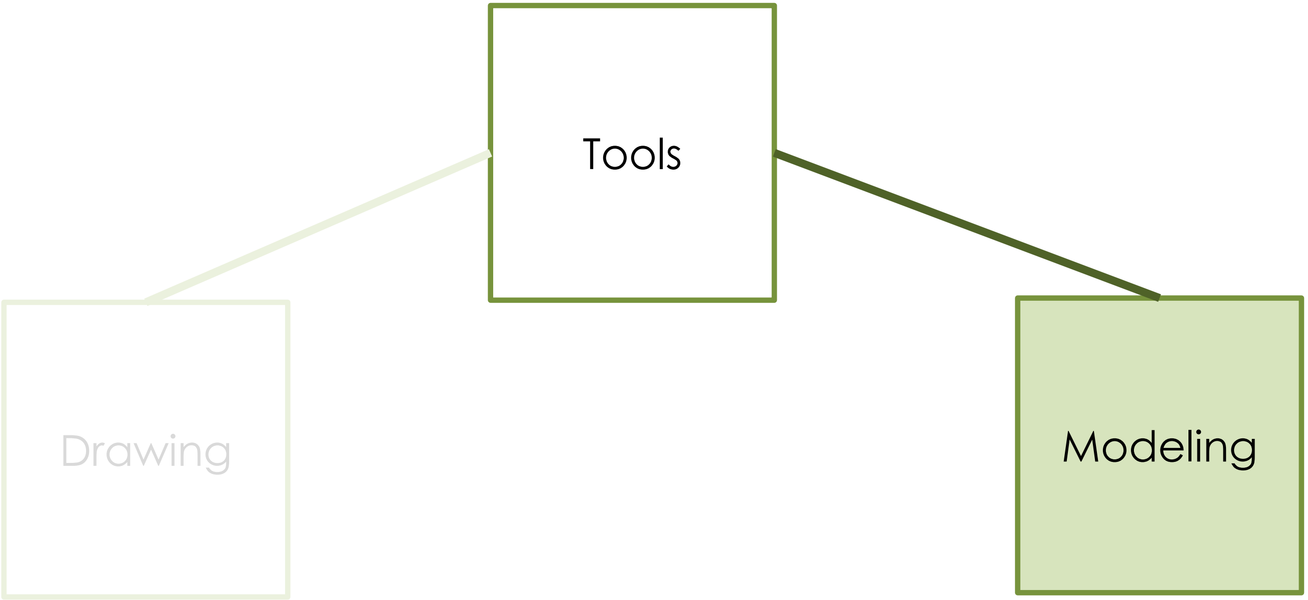


Major Diagrams

- Sequence
- Use case
- Class
- Activity
- Component
- State
- Object
- Deployment
- Timing
- Plus Archimate and others

- Text/CLI-based
- Can be installed locally

Modeling Tools



- No repository of elements
- Each diagram is standalone
- Very easy and portable
- Think Paper or Whiteboard

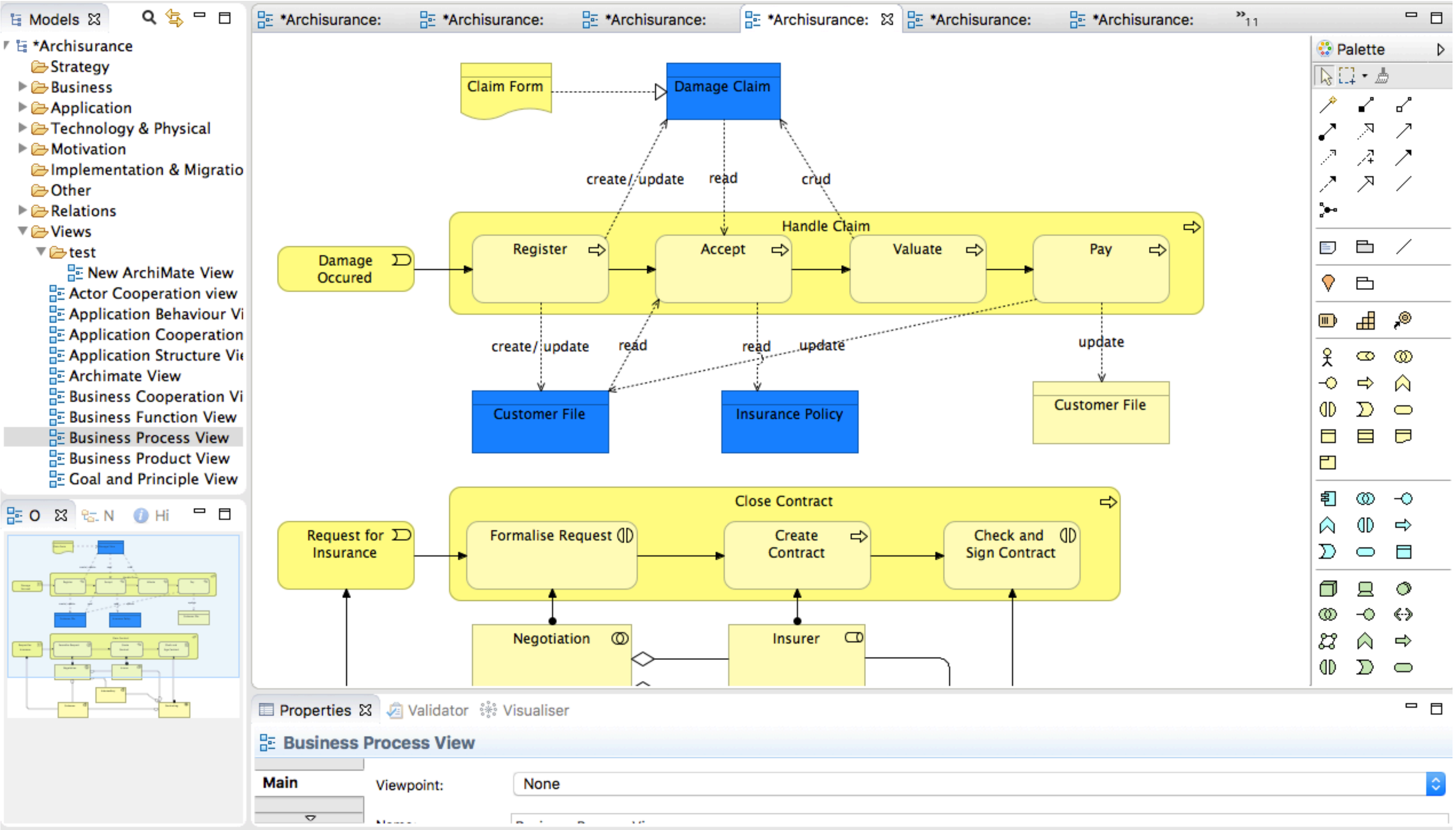
- Repository of elements
- Tied to a framework
- Lots of reuse
- Maybe too many rules
- Diagrams relate

Modeling Tools

Beware of the **Freemium** modeling tools.

You don’t want to pay for an in-app purchase to be able to do something like
“Print”

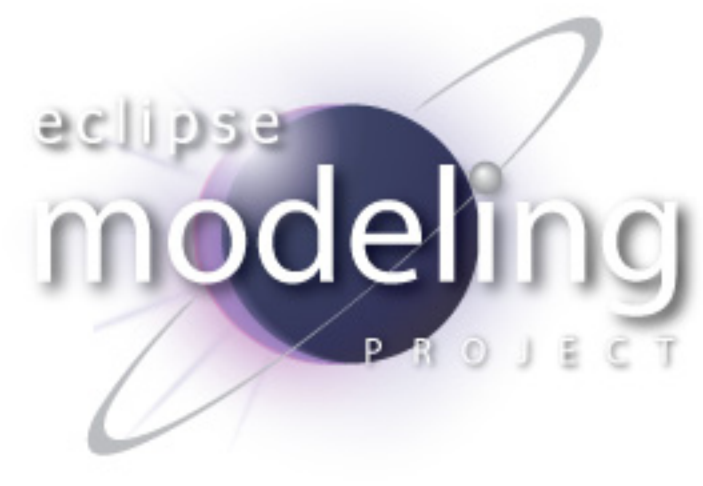
Modeling: Archi



Viewpoint: None

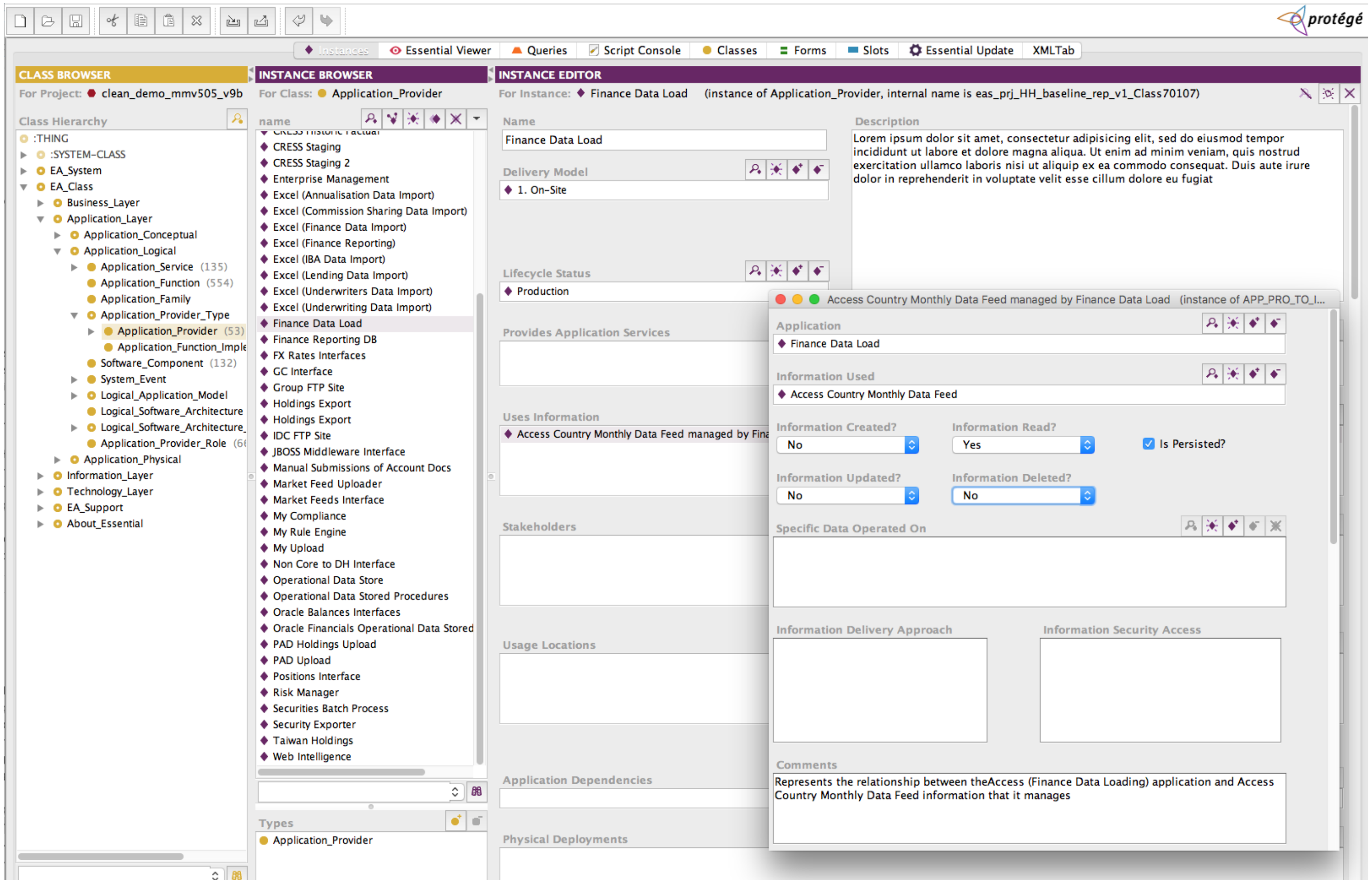
- Supports Archimate 3.0
- Java-based, cross platform
- True element/object modeling, keeps relations and attributes for all
- Element reuse
- Can create different element collection and views
- Not currently multi-user (recognized the issue and developing plugins)

Modeling: Eclipse



- Unleash the power of Eclipse
- Core driver is system and code generation
- Complex, terminology, UI, overwhelming feeling of despair
- If you are “all in” for Eclipse you need to take a look

Modeling: Essentials



For Project: clean_demo_mmv505_v9b

Class Hierarchy

- :THING
- :SYSTEM-CLASS
- EA_System
- EA_Class
  - Business_Layer
  - Application_Layer
    - Application_Conceptual
    - Application_Logical
      - Application_Service (135)
      - Application_Function (554)
      - Application_Family
      - Application_Provider_Type
        - Application_Provider (53)
        - Application_Function_Imple
      - Software_Component (132)
      - System_Event
      - Logical_Application_Model
      - Logical_Software_Architecture
      - Logical_Software_Architecture_
      - Application_Provider_Role (6
    - Application_Physical
  - Information_Layer
  - Technology_Layer
  - EA_Support
  - About_Essential

For Class: Application_Provider

name

- CRESS Historic Factual
- CRESS Staging
- CRESS Staging 2
- Enterprise Management
- Excel (Annualisation Data Import)
- Excel (Commission Sharing Data Import)
- Excel (Finance Data Import)
- Excel (Finance Reporting)
- Excel (IBA Data Import)
- Excel (Lending Data Import)
- Excel (Underwriters Data Import)
- Excel (Underwriting Data Import)
- Finance Data Load
- Finance Reporting DB
- FX Rates Interfaces
- GC Interface
- Group FTP Site
- Holdings Export
- Holdings Export
- IDC FTP Site
- JBOSS Middleware Interface
- Manual Submissions of Account Docs
- Market Feed Uploader
- Market Feeds Interface
- My Compliance
- My Rule Engine
- My Upload
- Non Core to DH Interface
- Operational Data Store
- Operational Data Stored Procedures
- Oracle Balances Interfaces
- Oracle Financials Operational Data Stored
- PAD Holdings Upload
- PAD Upload
- Positions Interface
- Risk Manager
- Securities Batch Process
- Security Exporter
- Taiwan Holdings
- Web Intelligence

Types

- Application_Provider

For Instance: Finance Data Load (instance of Application_Provider, internal name is eas_prj_HH_baseline_rep_v1_Class70107)

Name: Finance Data Load

Description: Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat

Delivery Model: 1. On-Site

Lifecycle Status: Production

Provides Application Services

Uses Information: Access Country Monthly Data Feed managed by Fina

Stakeholders

Usage Locations

Application Dependencies

Physical Deployments

Access Country Monthly Data Feed managed by Finance Data Load (instance of APP_PRO_TO_I...

Application: Finance Data Load

Information Used: Access Country Monthly Data Feed

Information Created?: No

Information Read?: Yes

☑ Is Persisted?

Information Updated?: No

Information Deleted?: No

Specific Data Operated On

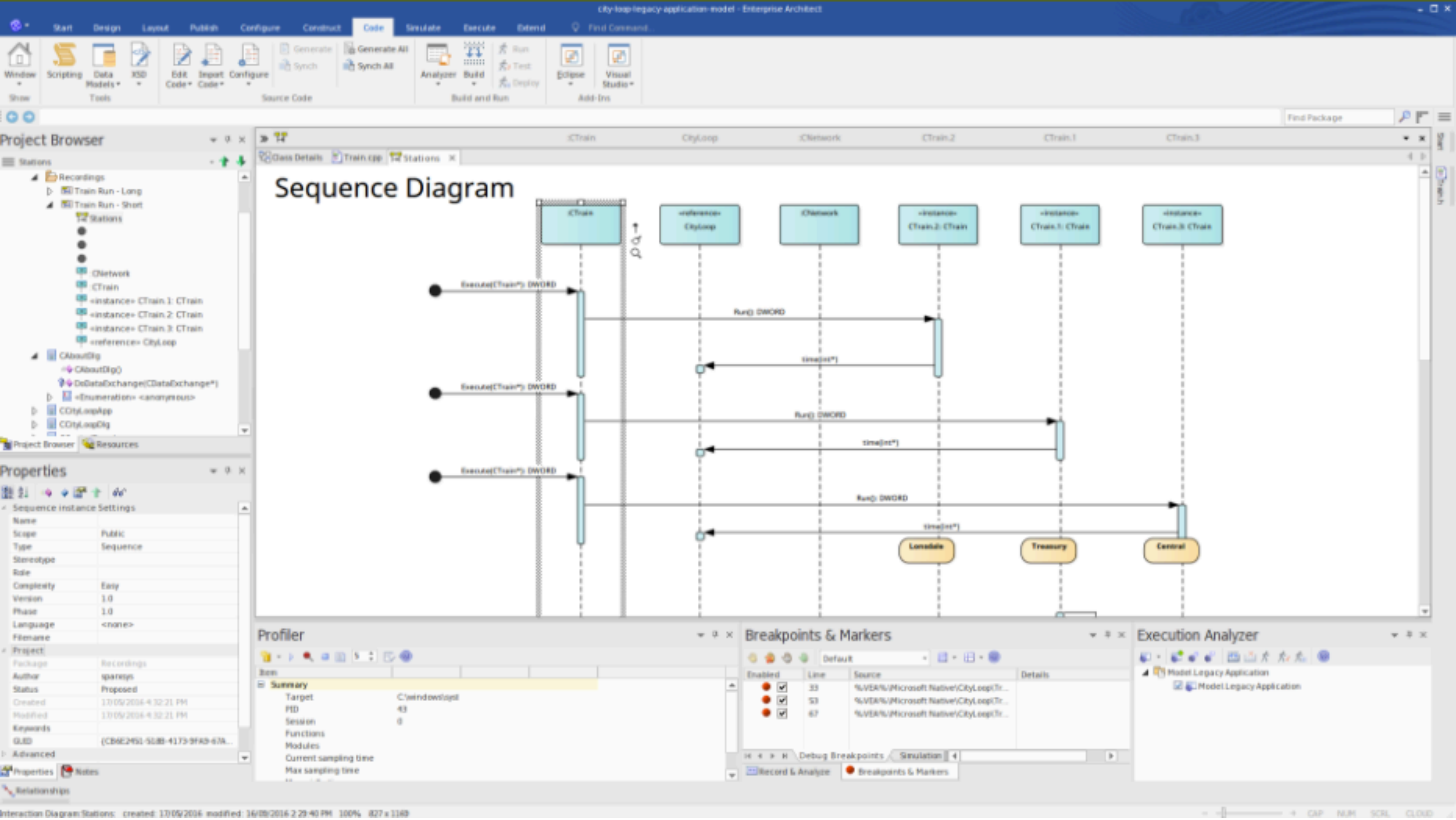
Information Delivery Approach

Information Security Access

Comments: Represents the relationship between theAccess (Finance Data Loading) application and Access Country Monthly Data Feed information that it manages

- Includes both a framework and a tools
- The actual tool, Essential Modeler, is very complex but includes reporting, the Essential Viewer (also complex)
- The framework concept is interesting and could be used on its own

Modeling: Sparx Enterprise Architect (Honorable Mention)



Properties

Name	
Scope	Public
Type	Sequence
Stereotype	
Role	
Complexity	Easy
Version	1.0
Phase	1.0
Language	<none>
Filename	
Project	
Package	Recordings
Author	sparxsys
Status	Proposed
Created	17/05/2016 4:32:21 PM
Modified	17/05/2016 4:32:21 PM
Keywords	
GUID	{CB6E2451-5088-4173-9FA9-67A...
Advanced	

Profiler

Item	
Summary	
Target	C:\windows\sysl
PID	43
Session	0
Functions	
Modules	
Current sampling time	
Max sampling time	

Breakpoints & Markers

Enabled	Line	Source	Details
☑	33	%VEA%\Microsoft Native\CityLoop\Tr...	
☑	53	%VEA%\Microsoft Native\CityLoop\Tr...	
☑	67	%VEA%\Microsoft Native\CityLoop\Tr...	

Interaction Diagram Stations: created: 17/05/2016 modified: 16/08/2016 2:29:40 PM 100% 827 x 1169

- Top of the line drawing app
- Low end model is very capable and less than $200
- Supports many frameworks including TOGAF, Zachman, Archimate, UML
- Best documentation

REFERENCES

Framework/Languages

- Archimate
  - http://pubs.opengroup.org/architecture/archimate3-doc/
- C4
  - http://www.codingthearchitecture.com/
- GML
  - http://wiki.c2.com/?GalacticModelingLanguage
- UML
  - http://www.uml.org/

Tools

- Eclipse
  - https://eclipse.org/modeling/
- Essentials
  - https://www.enterprise-architecture.org/home.php
- Plant UML
  - http://plantuml.com/
- Sparx
  - http://www.sparxsystems.com/
- Visio
  - https://products.office.com/en-us/visio/flowchart-software?tab=tabs-1

Ready, Set, Architect!

300 seconds to free architecture

Thanks!

Secret
Chipmunk

Ron Parker
@scmunk
http://www.secretchipmunk.com