

BLENDING PRODUCT THINKING WITH ARCHITECTURE

Joel Tosi

Slides: <https://www.dojoandco.com/speaking>

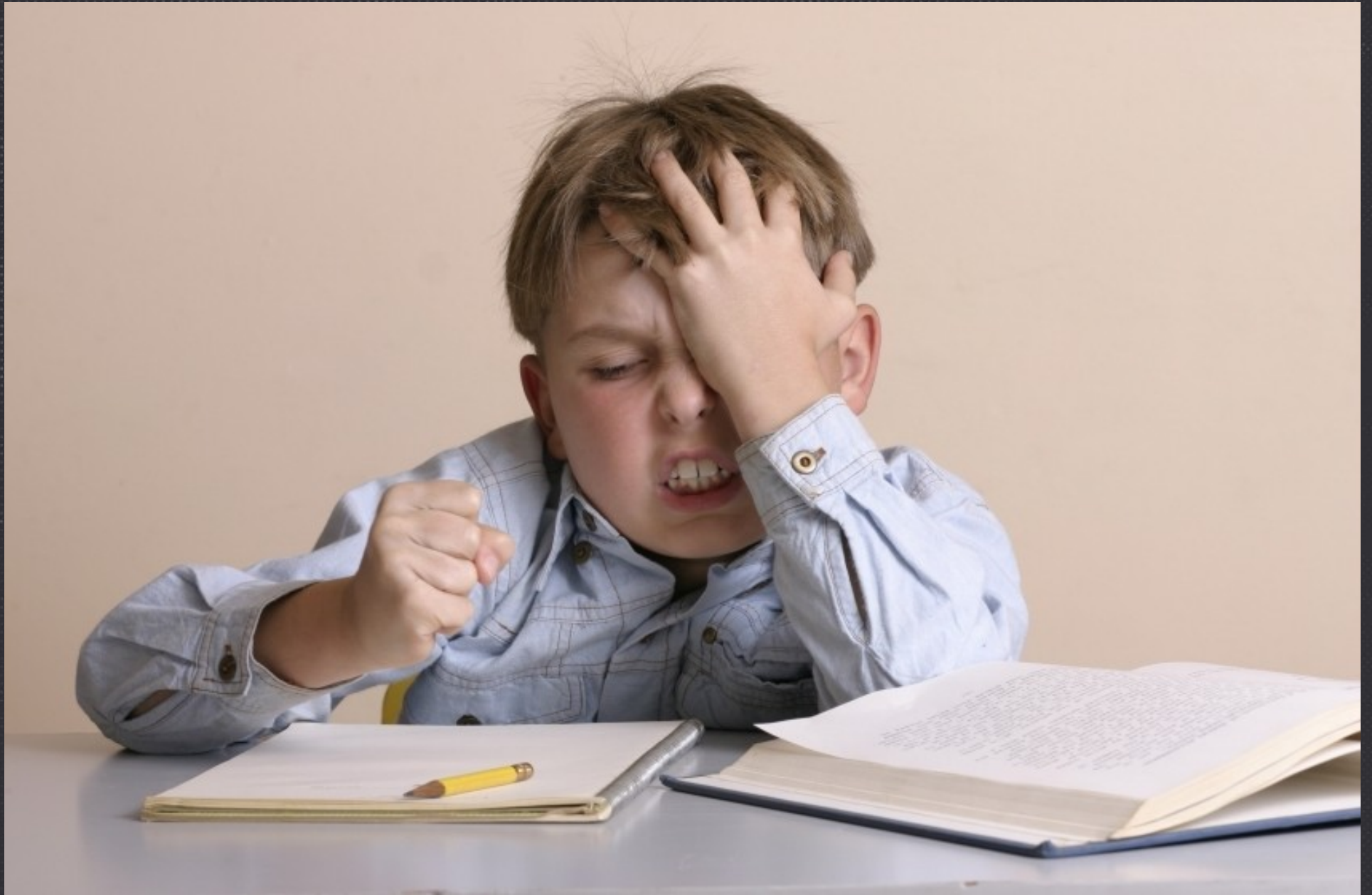


@joeltosi @dojoandco

[dojoandco.com](https://www.dojoandco.com)

Joel.tosi@dojoandco.com

MY PERSPECTIVE



Behind Every Architecture Decision

There is a Product Assumption

EXPECTED OUTCOMES

- Simple Visuals You Can Use Now
- Less 'What Architecture' and More 'When'
- Leverage Product Context to Ground Decisions
- Ideas on Documenting Decisions

WHAT IS THE PROBLEM?

WHAT IS ARCHITECTURE?

Web

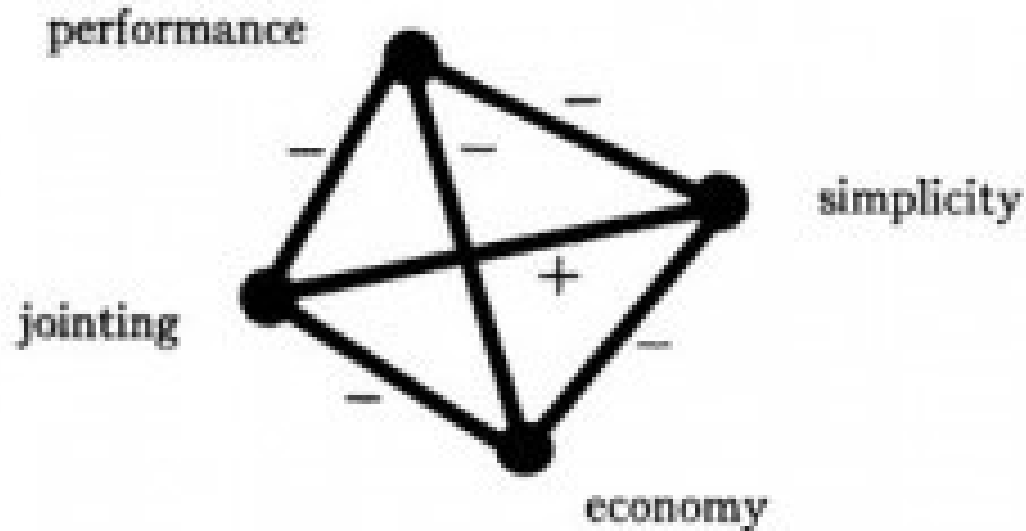
API

Logic / Calculations

Fast Data

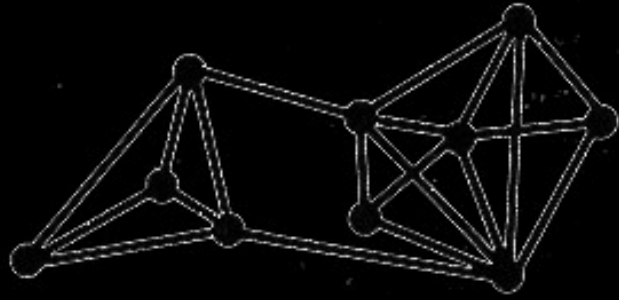
Slow Data

WHAT IS ARCHITECTURE?

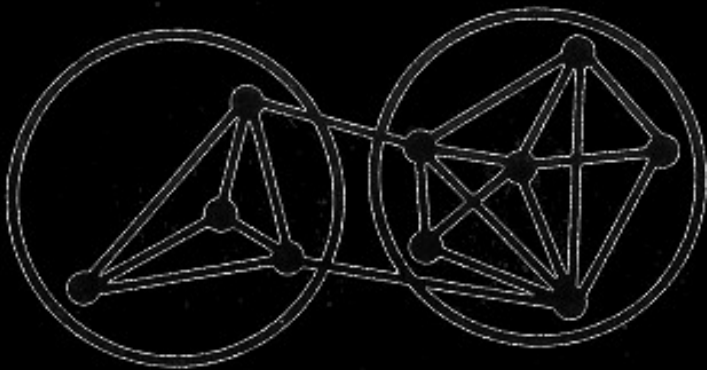


This is a typical design problem; it has requirements which have to be met; and there are interactions between the requirements, which makes the requirements hard to meet. This problem is simple to solve. It falls easily within the compass of a single man's intuition. But what about a more complicated problem?

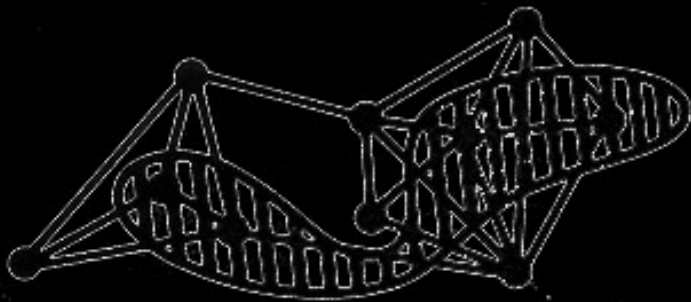
WHAT IS ARCHITECTURE?



Reflection of System Dependencies



Optimal decomposition / Grouping



Biased simplification - *mereotopology*

© Notes on the Synthesis of Form – Christopher Alexander

@joeltosi @dojoandco

APPLYING TO SOFTWARE

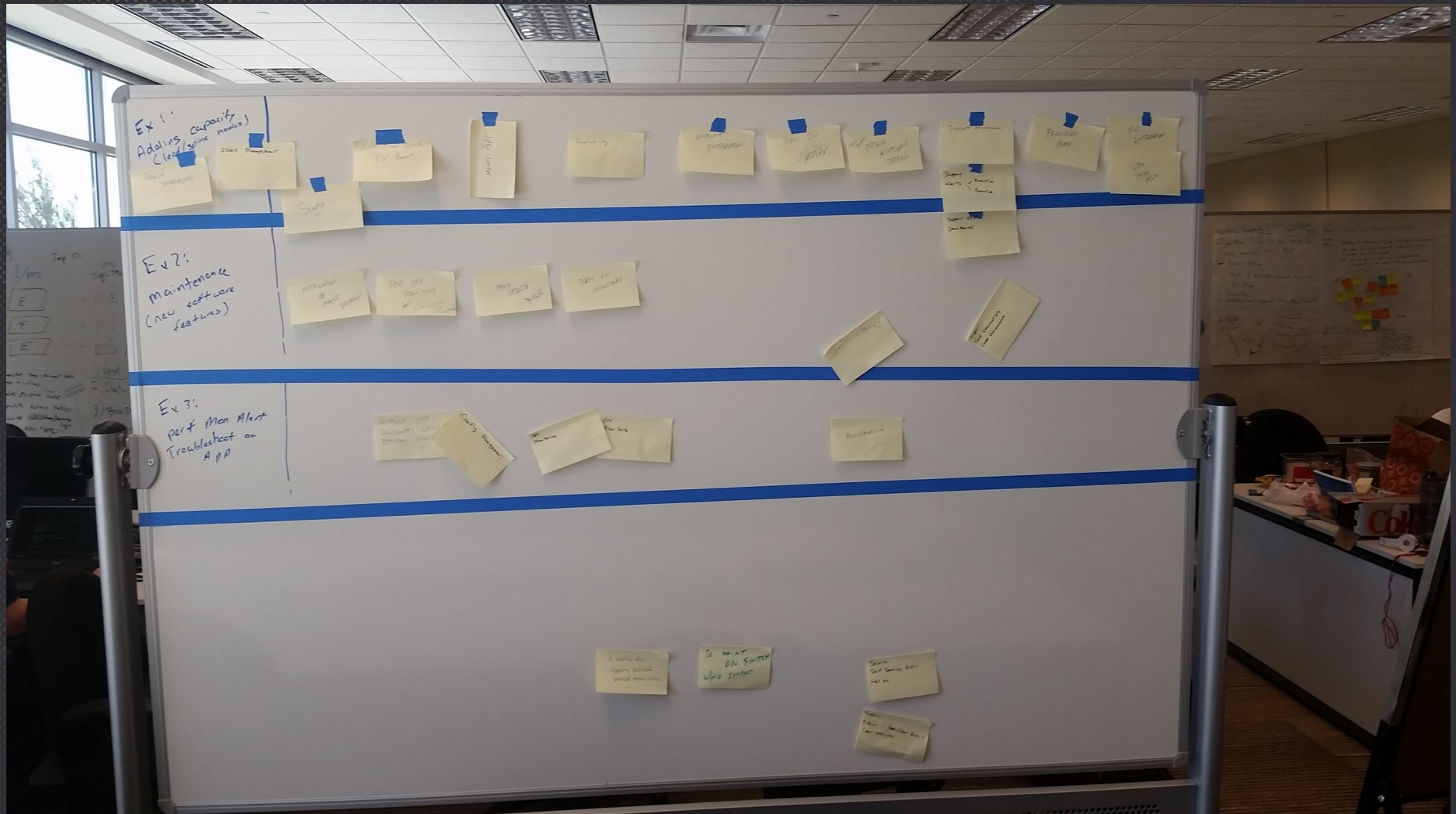
- Are the architectures you work on simple?
- Do the components have inter-dependencies & trade-offs?
- Does the language of the solution bias your decisions?

USING PRODUCT TO GROUND DECISIONS

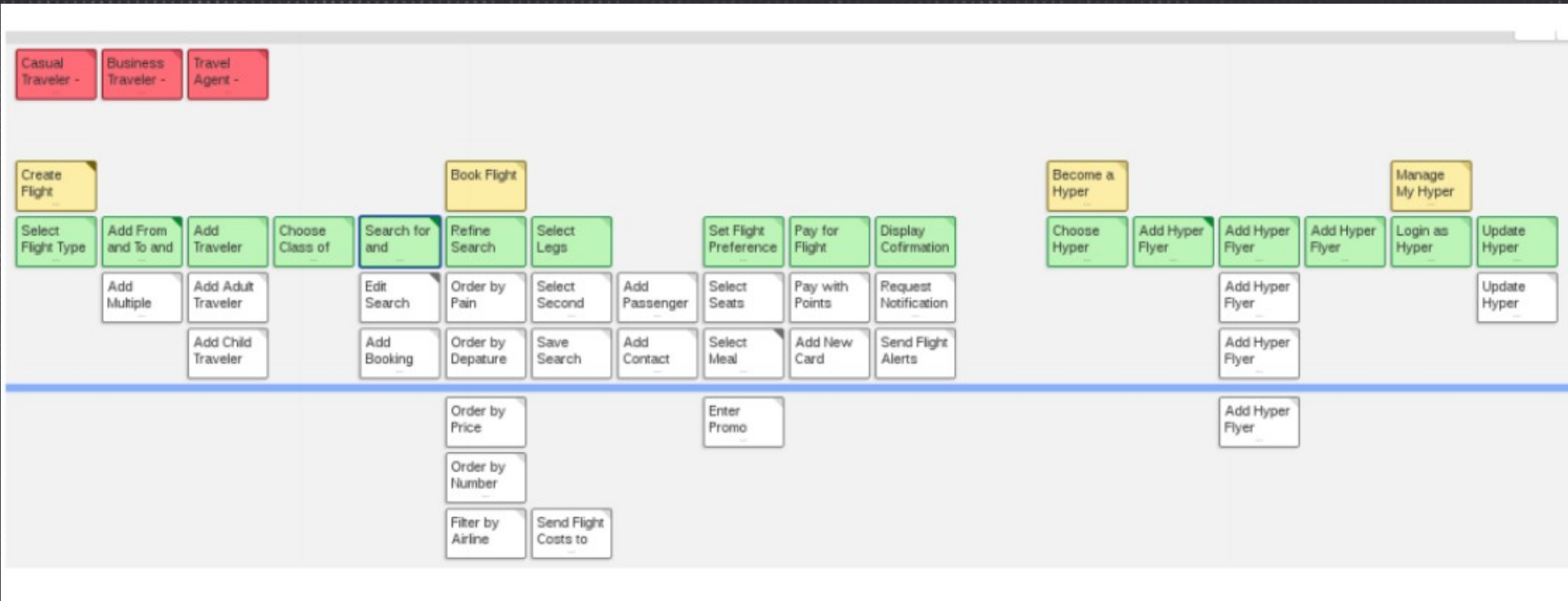
BLENDING PRODUCT WITH ARCHITECTURE



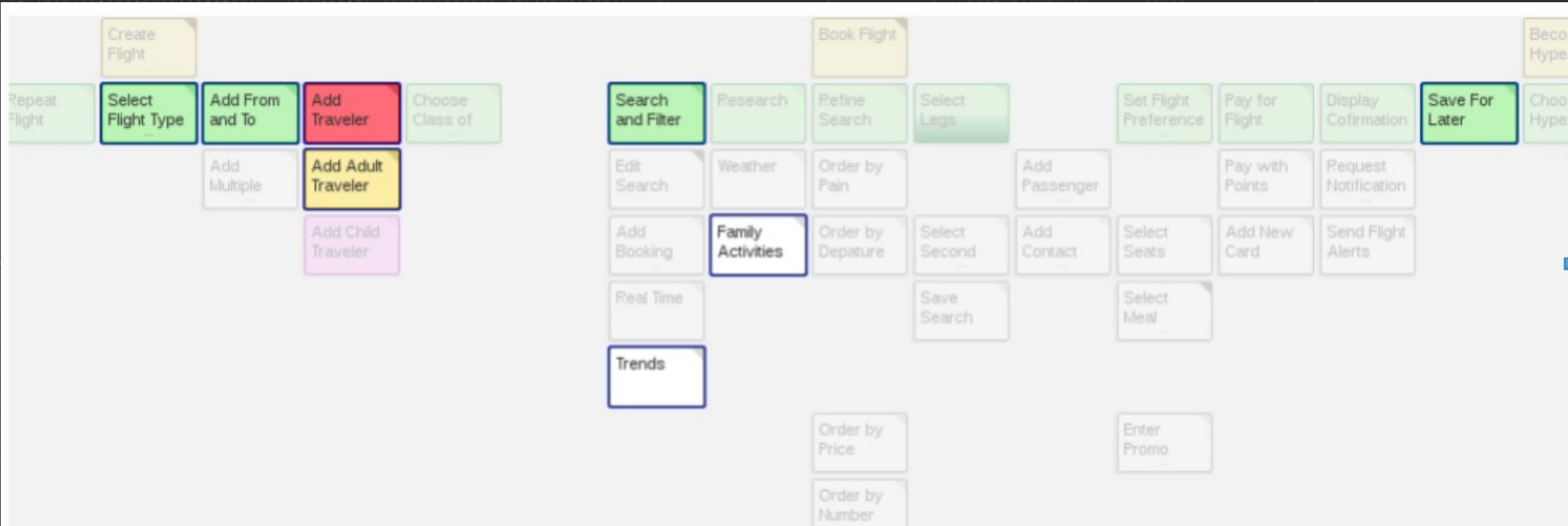
USING TARGETS AND JOURNEYS



ALL THE POSSIBILITIES



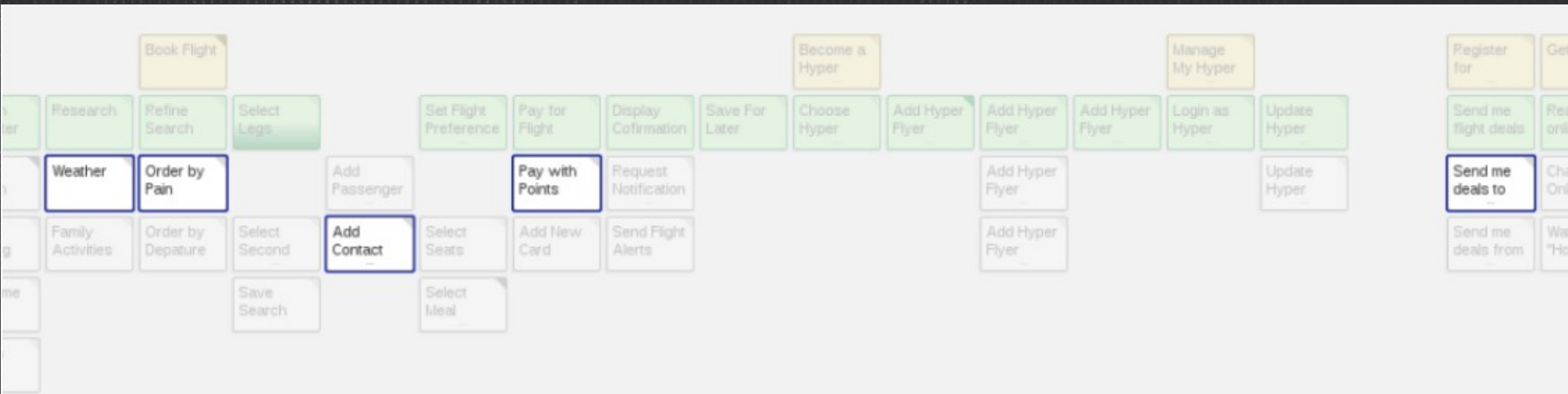
FAMILY TRAVELING



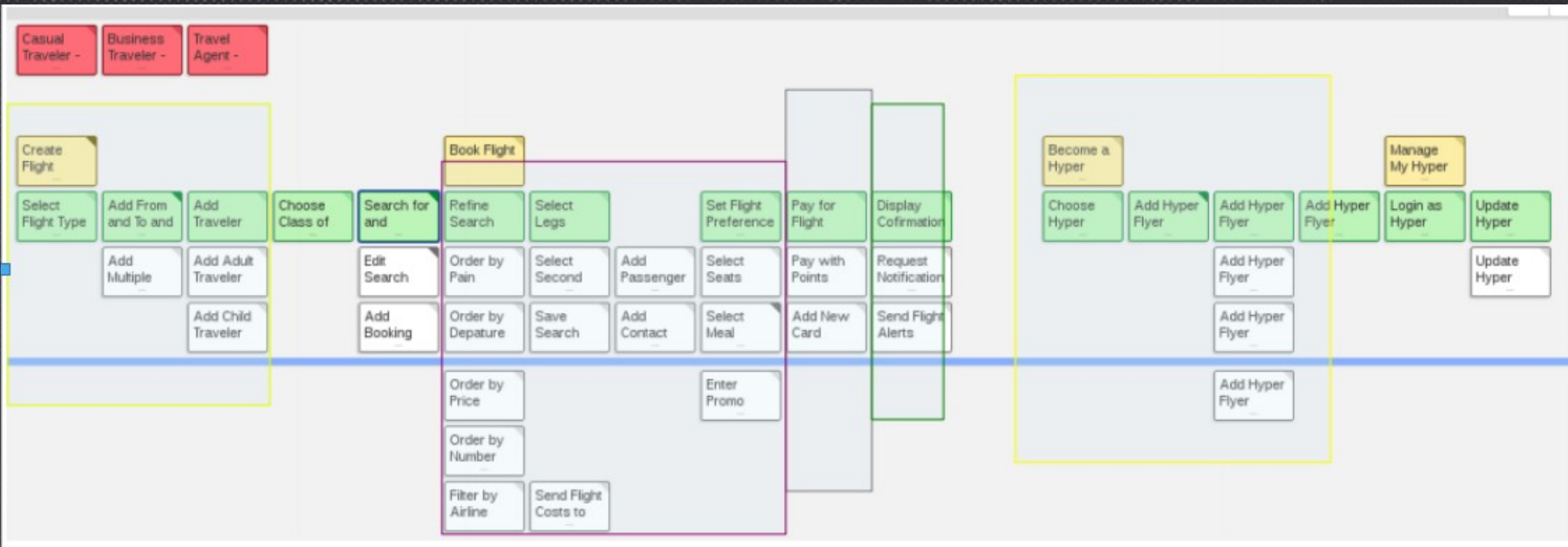
BUSINESS TRAVELER



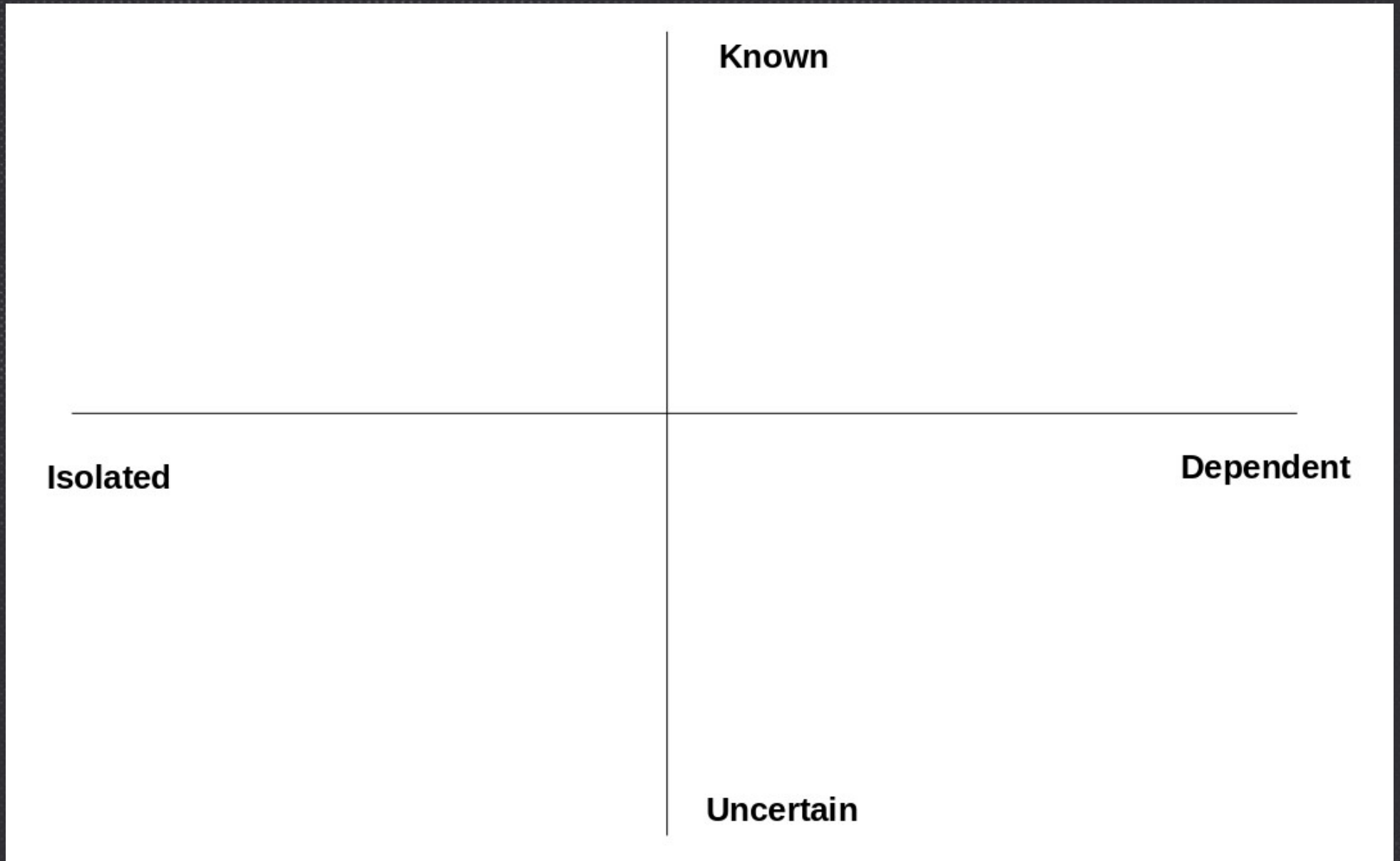
ADVENTURER



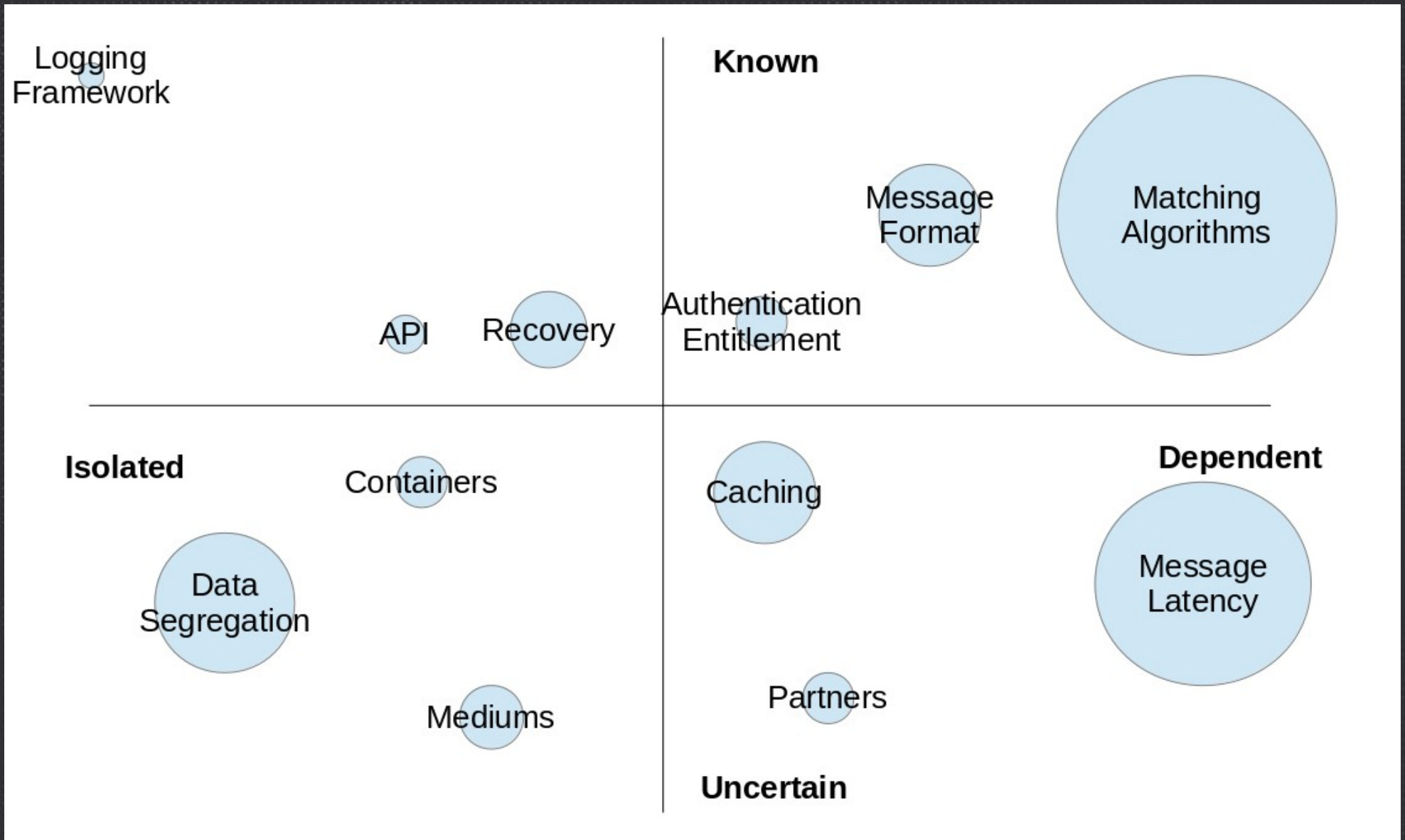
MICROSERVICES



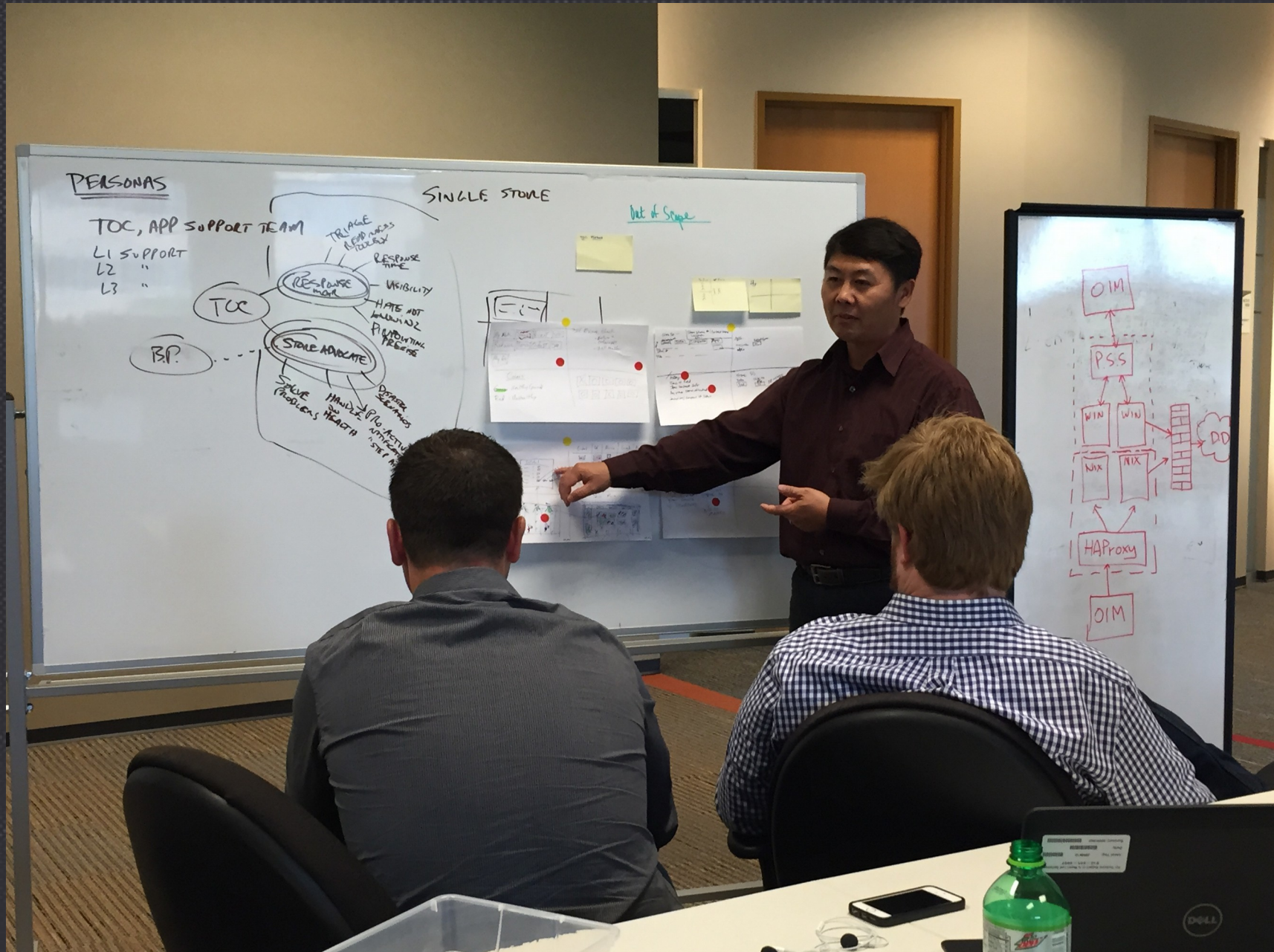
DETERMINING DECISION POINTS



SAMPLE MARKET DATA



IN ACTION



UNDERSTANDING THE ASSUMPTIONS OF YOUR ARCHITECTURE

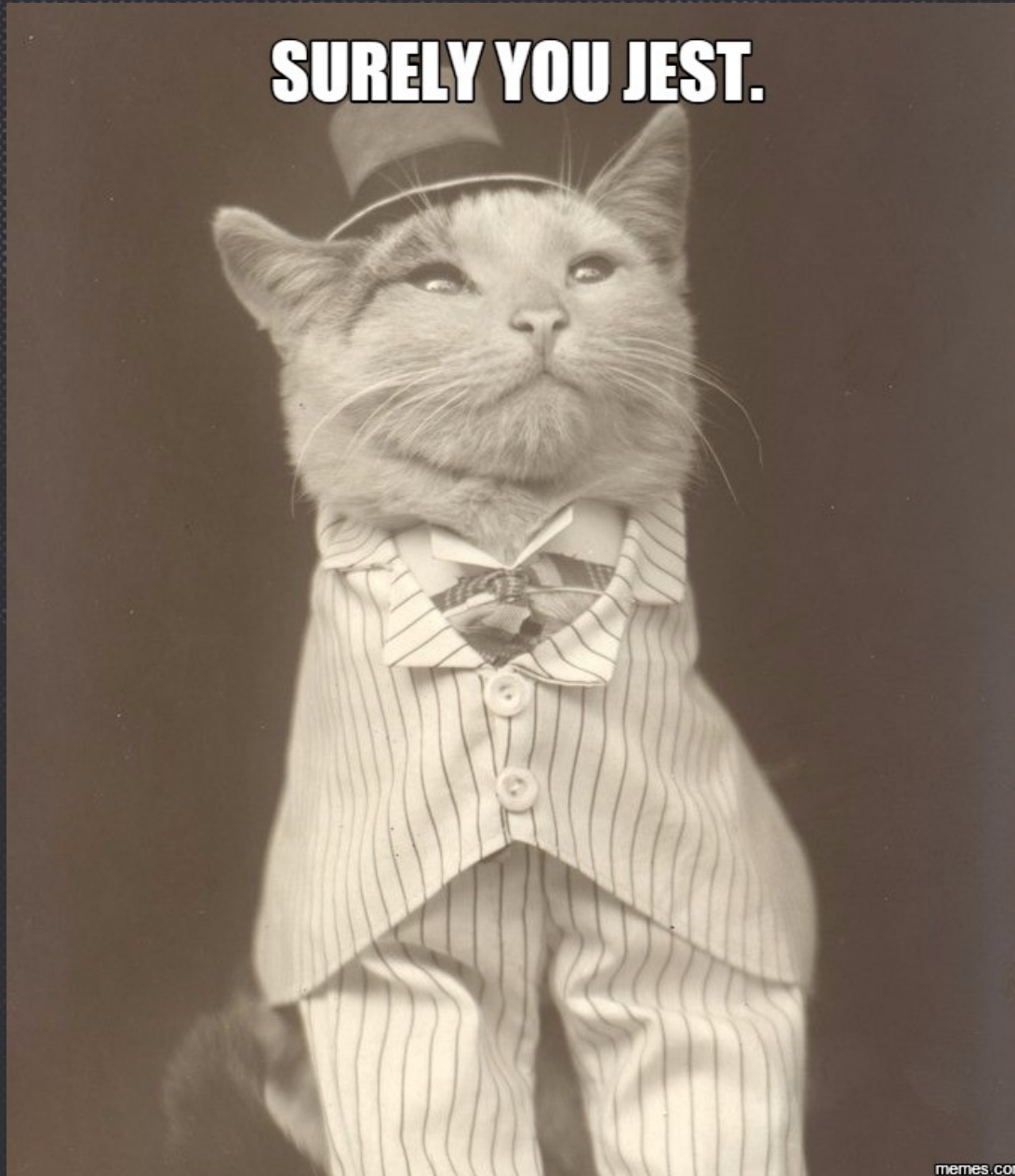
HOW WILL IT BE USED VS WHAT YOU WANT



@joeltosi @dojoandco

WHAT IF YOU WERE WRONG?

SURELY YOU JEST.



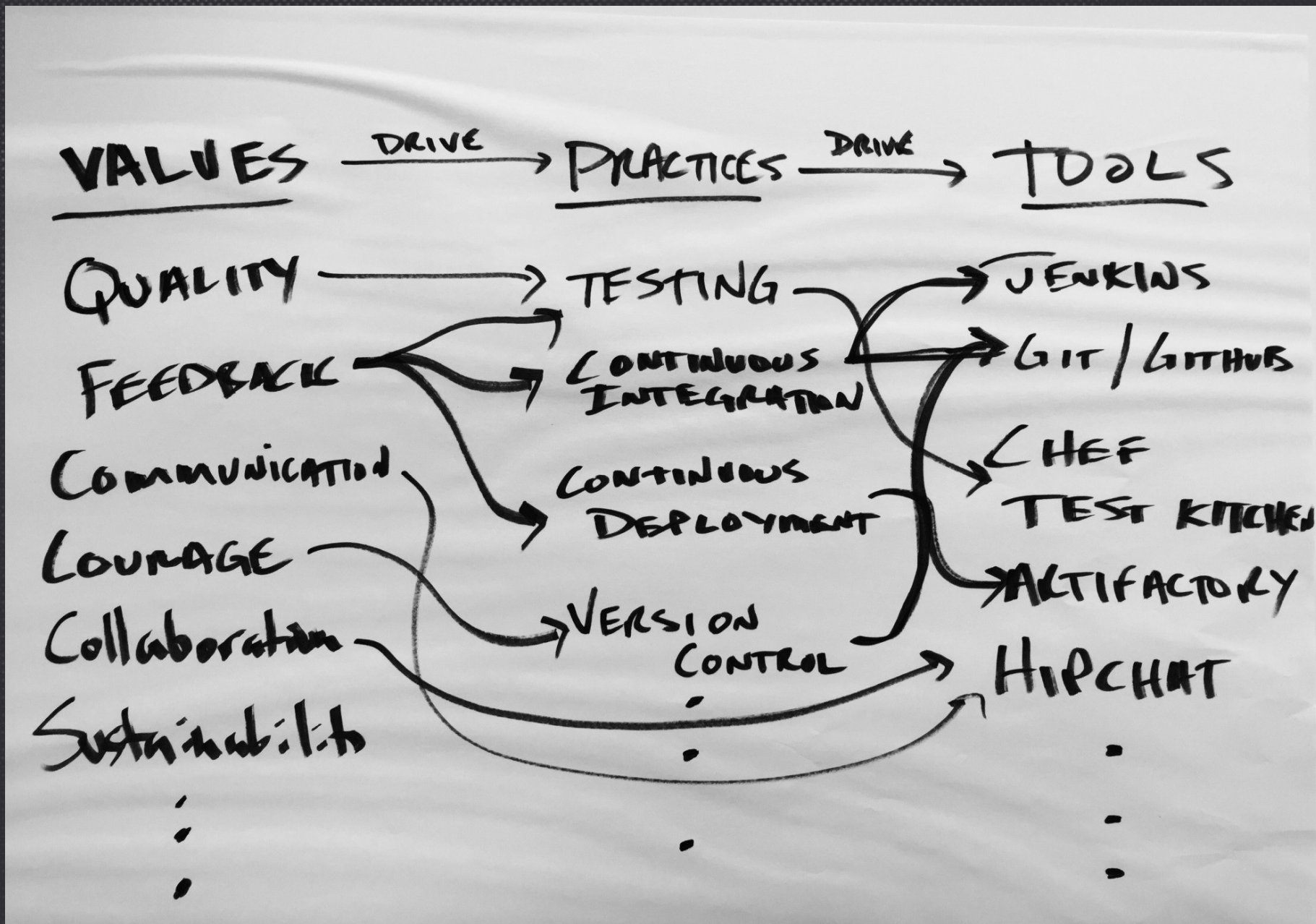
memes.com

@joeltosi @dojoandco

TRY THIS

- Document Product Assumptions You Are Making
- Document Technical Assumptions You Are Making
- Think of 3 Ways You Might Be Wrong
- For Each Error, What Would Change?

ROOT DECISIONS



WHAT ARE YOUR CONSTRAINTS

- Never been a perfect architecture
- Use Product Discussions to see what decisions when
- Grow context around the decisions - why / when

What Can You Do About It?
Necessary or Accidental?
Assumed or Real?

WOULD NEW CONSTRAINTS HELP

- Not All of the Databases
- No New Code
- Must Respond in Less Than 2 Seconds

OUR JOB

Take a World of Possibilities...

Apply Proper Product Understanding and Constraints...
To Deliver Maneuverable Systems

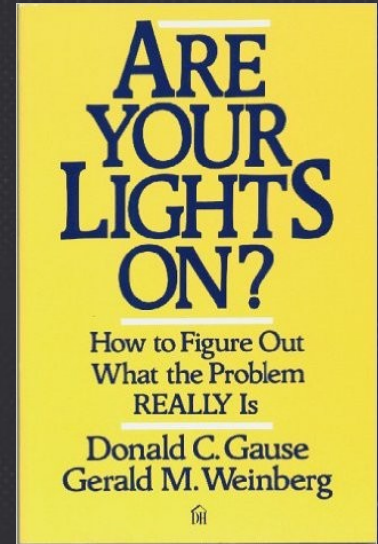
That Bring Joy

RECAP

- Be aware of models and dependencies
- Use Product Discussions to avoid language bias
- Use Journeys to find when decisions can be made
- Understand parts of the system need to be right / emerge

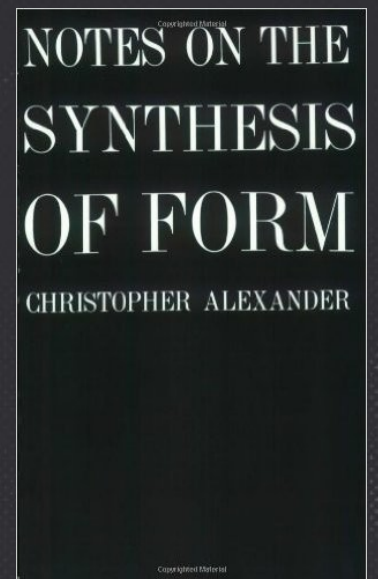
RECOMMENDED READING

Jerry Weinberg - Are Your Lights On?



Christopher Alexander -

Notes on the Synthesis of Form



@joeltosi @dojoandco

"I've seen firsthand the profound inspiration and improvement that Joel and Dion's dojo approach has brought to companies. They go beyond coaching and workshoping to help organizations build competence with deliberate practice and reflection. Creating a "learning organization" has been an aspirational goal for many companies. Joel and Dion finally provide an approach for doing just that."

—JEFF PATTON, AUTHOR OF USER STORY MAPPING

Transforming a company's digital product development capabilities is a monumental task. Keeping teams up to date with modern software engineering and product discovery practices is daunting enough, but when you add in adopting agile and lean methods, operating model changes, and cultural transformation, it's easy to see how even the most seasoned leaders feel completely overwhelmed.

Traditional approaches to training—from one-day workshops to week-long trainings—fail to have the impact you want. Instead of transformation, you get piecemeal improvements that don't lead to significant change.

Here's the truth: if you want learning that sticks, you need a dojo.

In *Creating Your Dojo*, experienced dojo coaches Joel Tosi and Dion Stewart guide you through creating a dojo—an immersive learning environment—within your organization. The dojo enables your teams to learn new skills within the context of doing their real-world work. You'll create a thriving product development culture where team members feel empowered to solve their own problems. Your organization will improve the quality of the products they deliver, reduce delivery cycle time, and create innovative products with better product/market fit.



Dion Stewart and Joel Tosi are the co-founders of Dojo and Co.

DION STEWART has coached software teams at Fortune 50 companies in multiple domains over the past twenty years: healthcare, finance, investment banking, retail, entertainment, and more. Given his developer's background and transformation consulting experience, Dion is both a highly effective coach and a sought-after speaker. He's given talks at QCon, Agile Development West, Craft Conf, and DDD Europe. He lives in rural Pennsylvania with his wife and enjoys playing and recording music.



JOEL TOSI worked as an engineer and architect at the Chicago Mercantile Exchange, leaving after ten years and kickstarting their evolution. He then spent a few years as the lead North American architect for Red Hat Middleware. He's been coaching software teams and organizations on improving product development capabilities for fifteen years. An in-demand speaker, Joel has spoken at the Agile Alliance Conference, DevOps Days, SQE Agile Development West, and Dev Nexus. He lives in Chicago with his wife and three children.



CREATING YOUR DOJO

DION STEWART & JOEL TOSI



CREATING YOUR DOJO

Upskill Your Organization
for Digital Evolution

DION STEWART & JOEL TOSI

WHAT ARE YOUR QUESTIONS?

THANK YOU

Joel Tosi

Slides: <https://www.dojoandco.com/speaking>



@joeltosi @dojoandco [dojoandco.com](https://www.dojoandco.com) Joel.tosi@dojoandco.com