

When Customers Don't Share a Language



A Domain-Driven Design Journey



chrissimon.au



@ChrisSimonAu



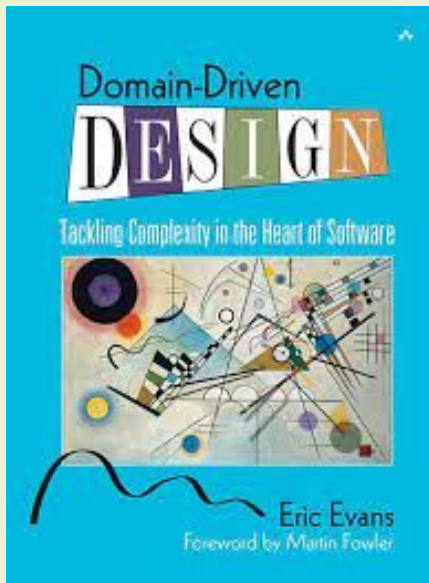
@ChrisSimonAu@hachyderm.io

*Empowering Technology
Teams to Drive Business
Success*

About Chris

- 2 x Co-founding Startup CTO
- Startup CTO Coach
- Domain-Driven Design Trainer/Consultant
- Advisor, TDD trainer, meetup organiser, open source developer, speaker





*“This book provides a **framework** for making design decisions and a **technical vocabulary** for discussing **domain design**. It is a synthesis of widely accepted best practices along with my own insights and experiences. Software development teams facing complex domains can use this framework to **approach domain-driven design systematically**.”*

- Eric Evans, 2003

INSIDE YOU THERE ARE TWO WOLVES



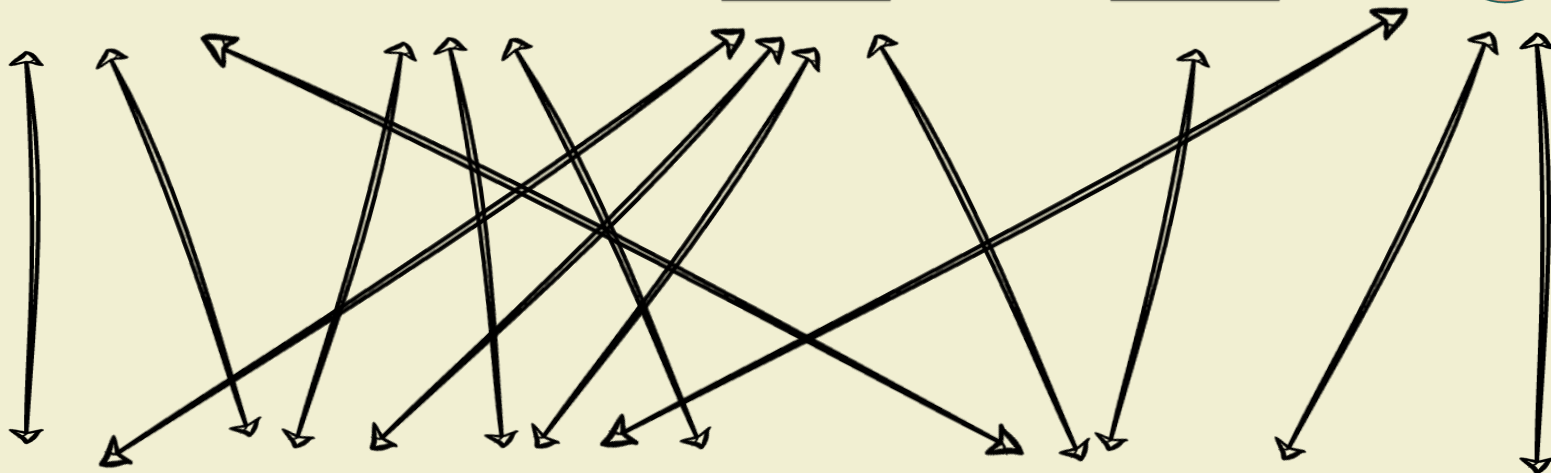


*"How much
will you pay
me to do XYZ
for Person
123?"*

"\$50"

*"I did XYZ for
Person 123"*

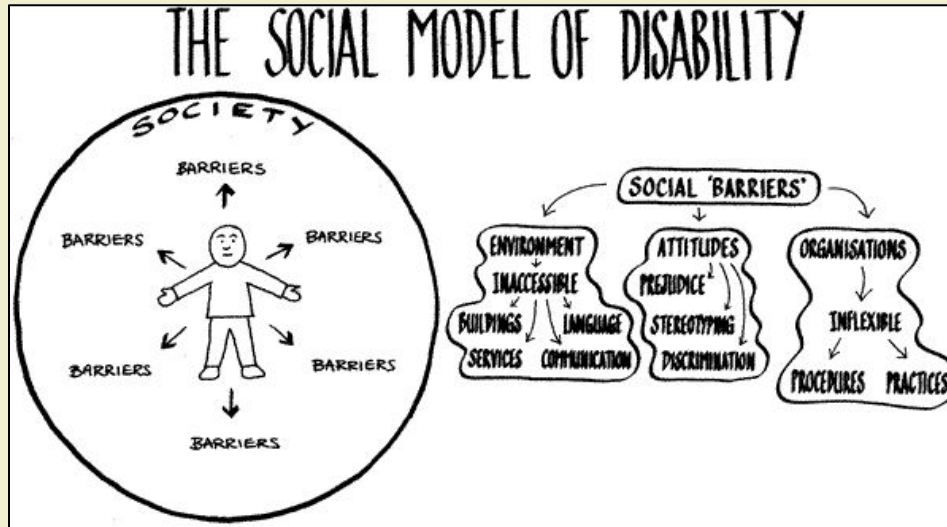






*"Here's \$50,000,
look after Person
123 for the next
year"*

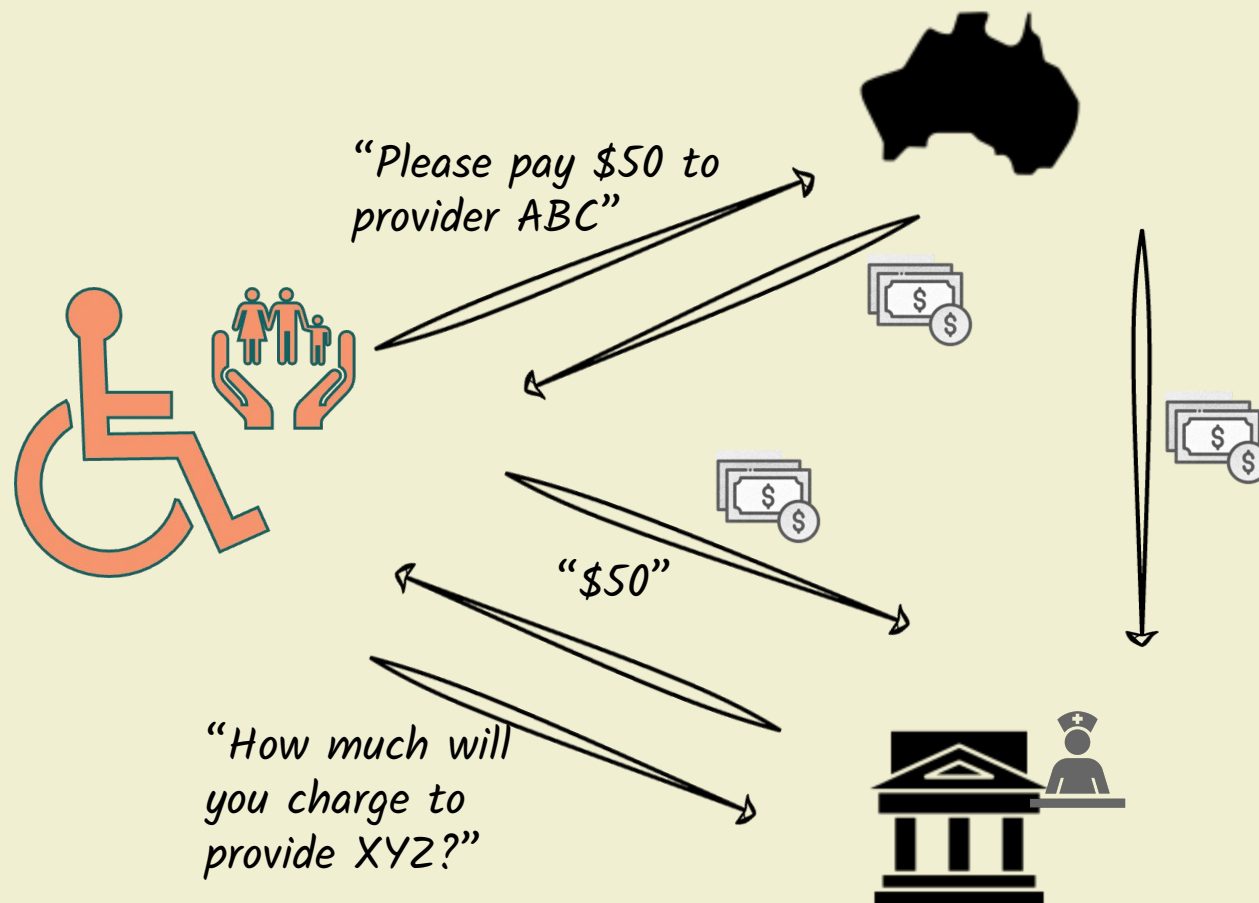


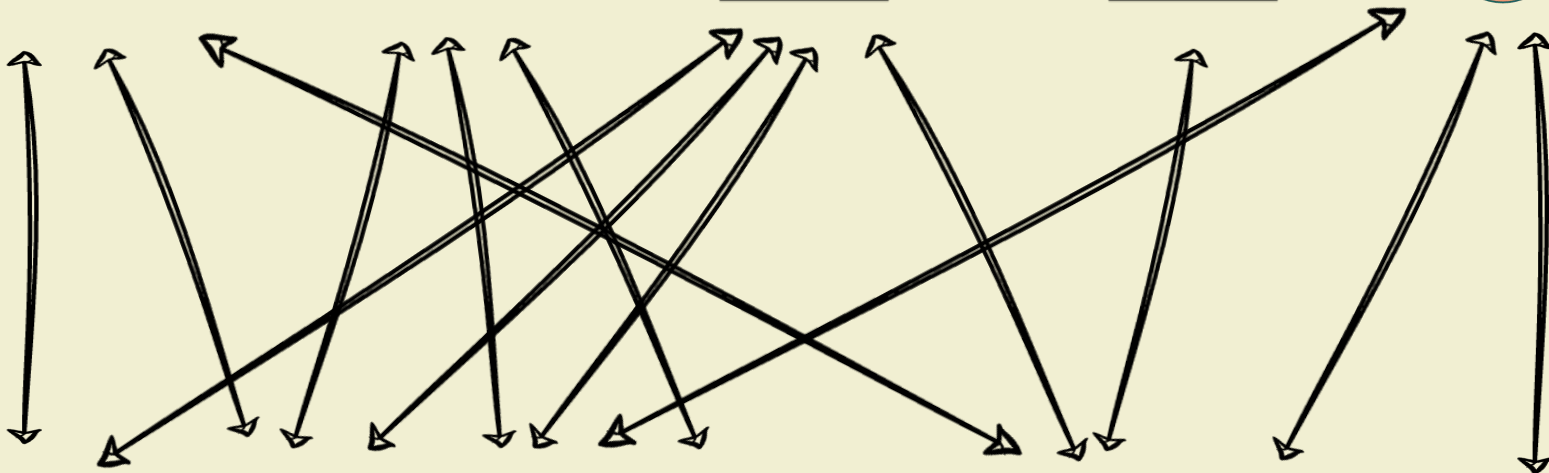


<http://ddsg.org.uk/taxi/social-model.html>

The Human Rights Model of Disability





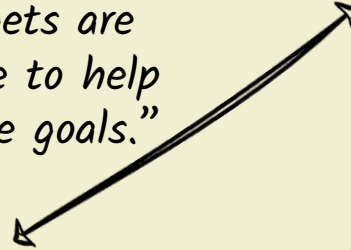


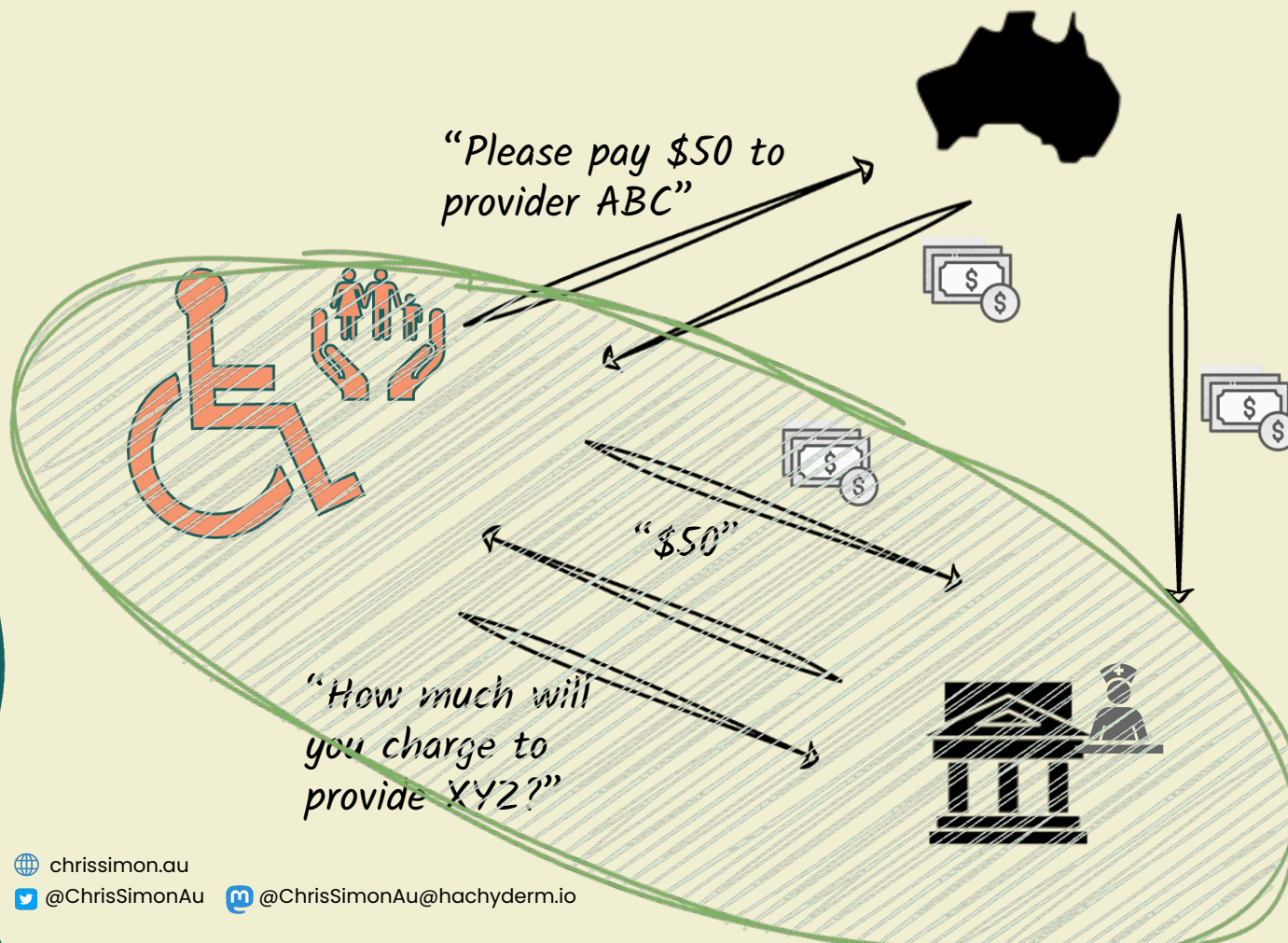
Mission:

Support the success of the new disability funding model by facilitating transparency, efficiency and ease of access to claiming and payments.



"Let's agree on a plan: your goals and what budgets are appropriate to help meet those goals."

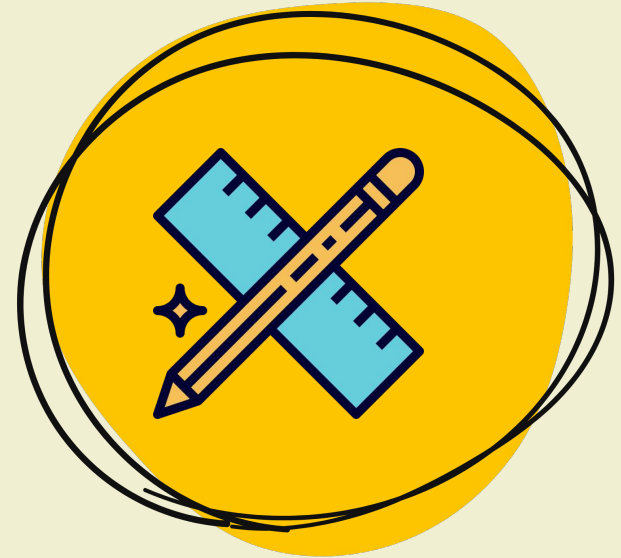


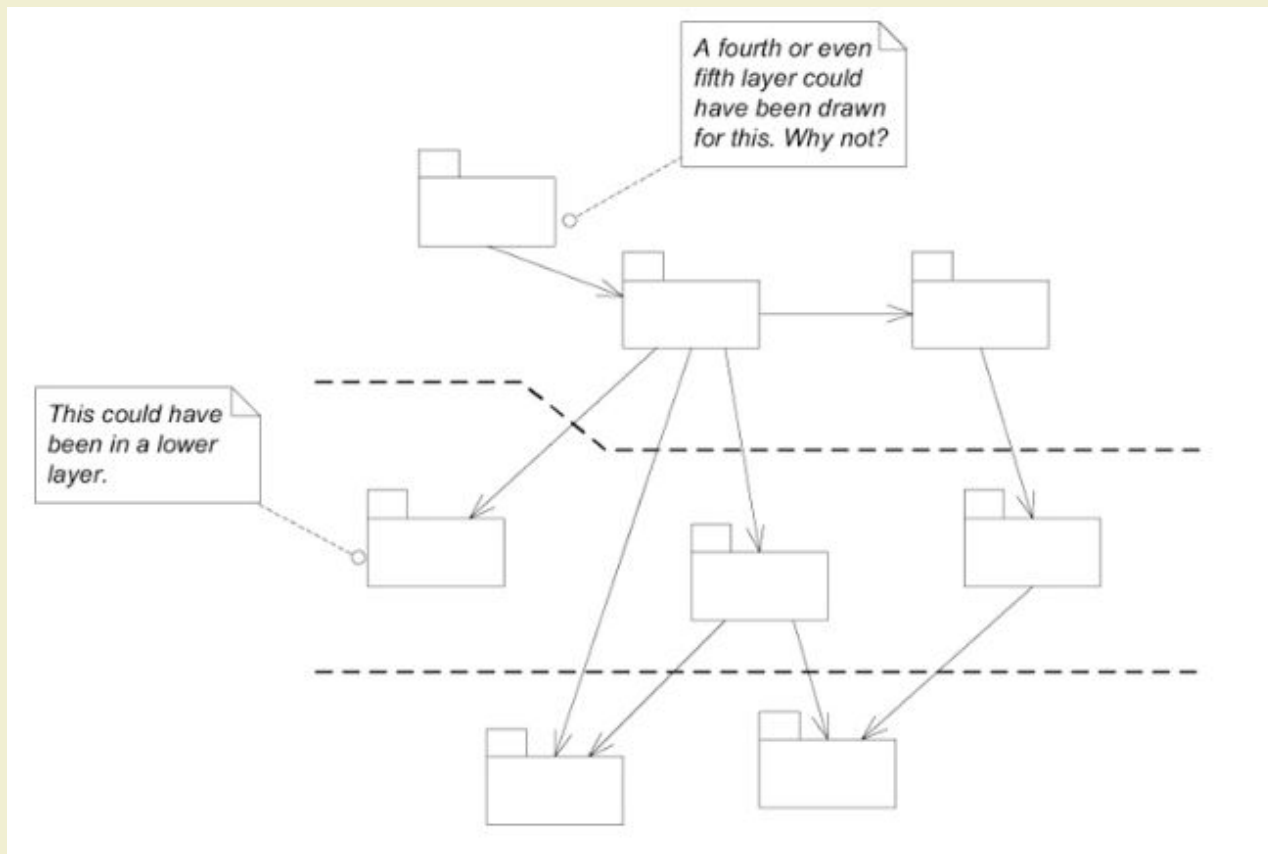


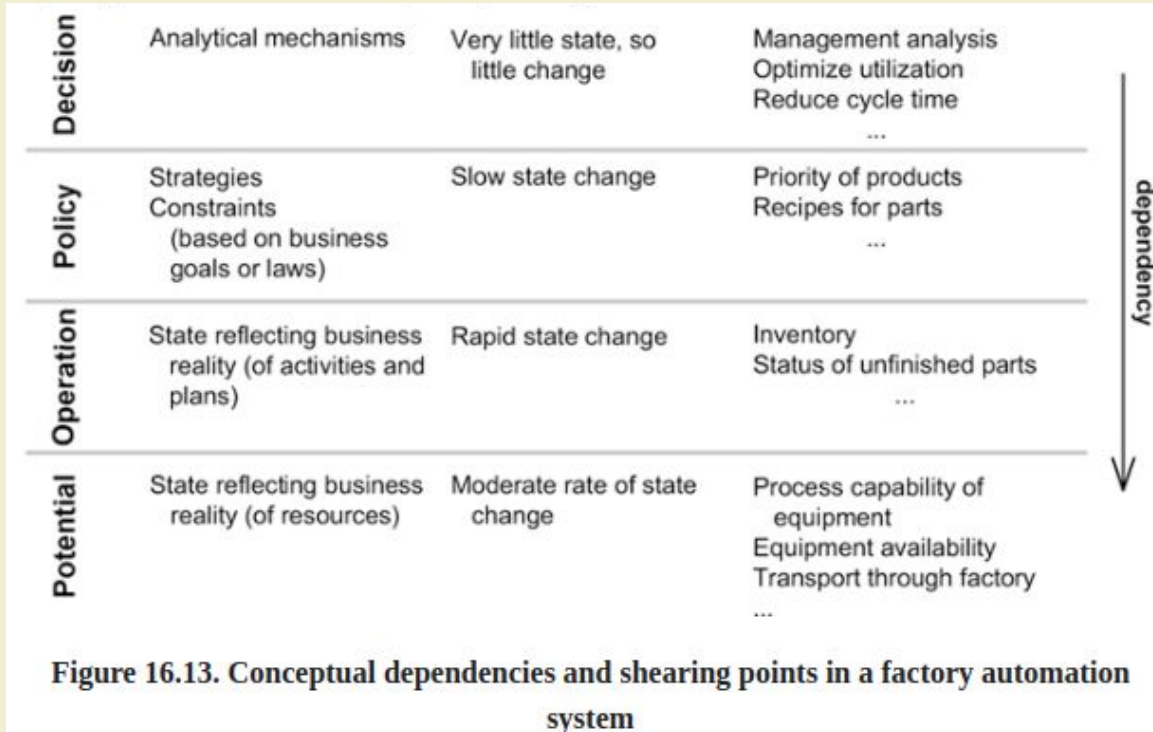
Initial Goal:

Support participants in
managing the use of their plans.

Phase 1 - Initial Design Concept







Finance

State reflecting reality of activity (money movements)

Operations

State reflecting reality of activity (claims, transaction history, remaining balance)

Admin

State reflecting reality of resources – individual plans (budgets)

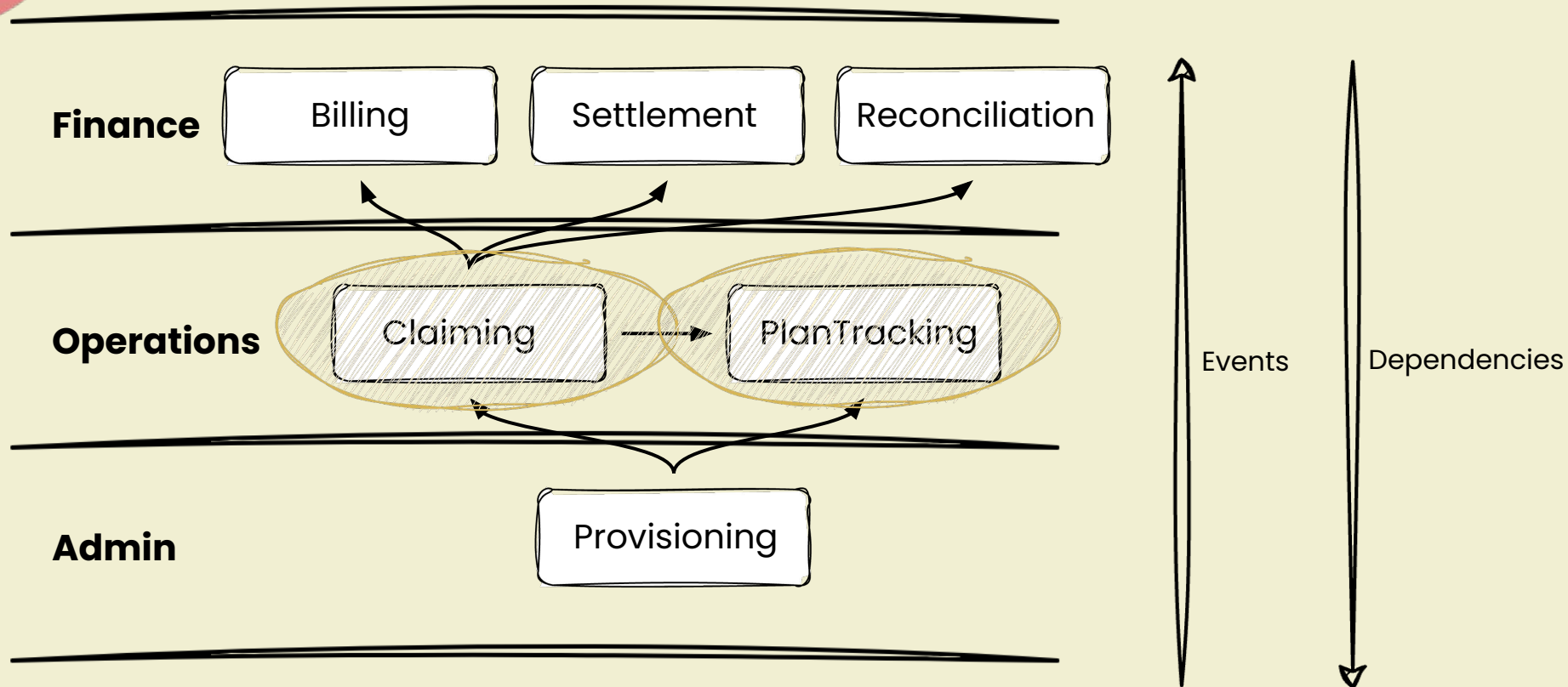


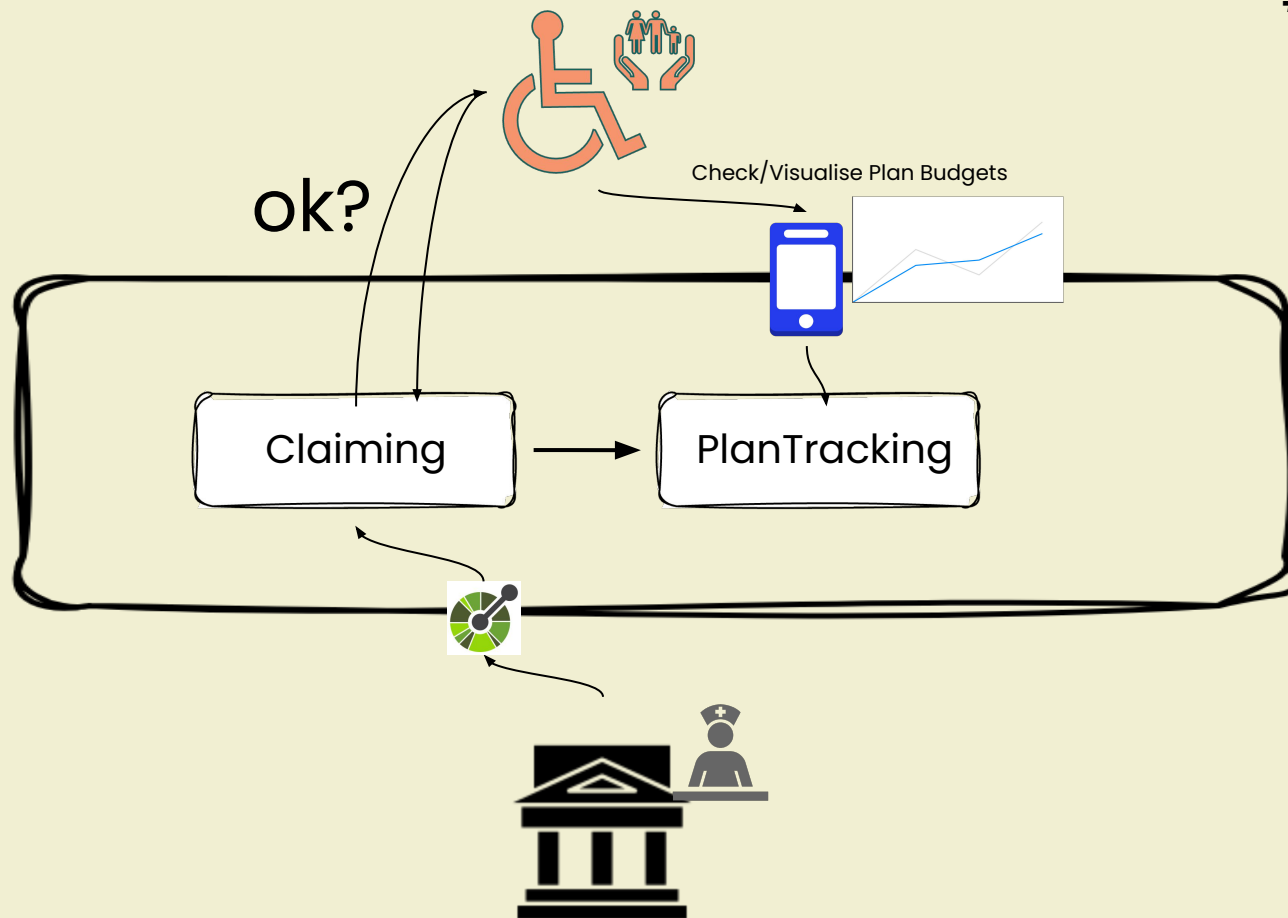
Events

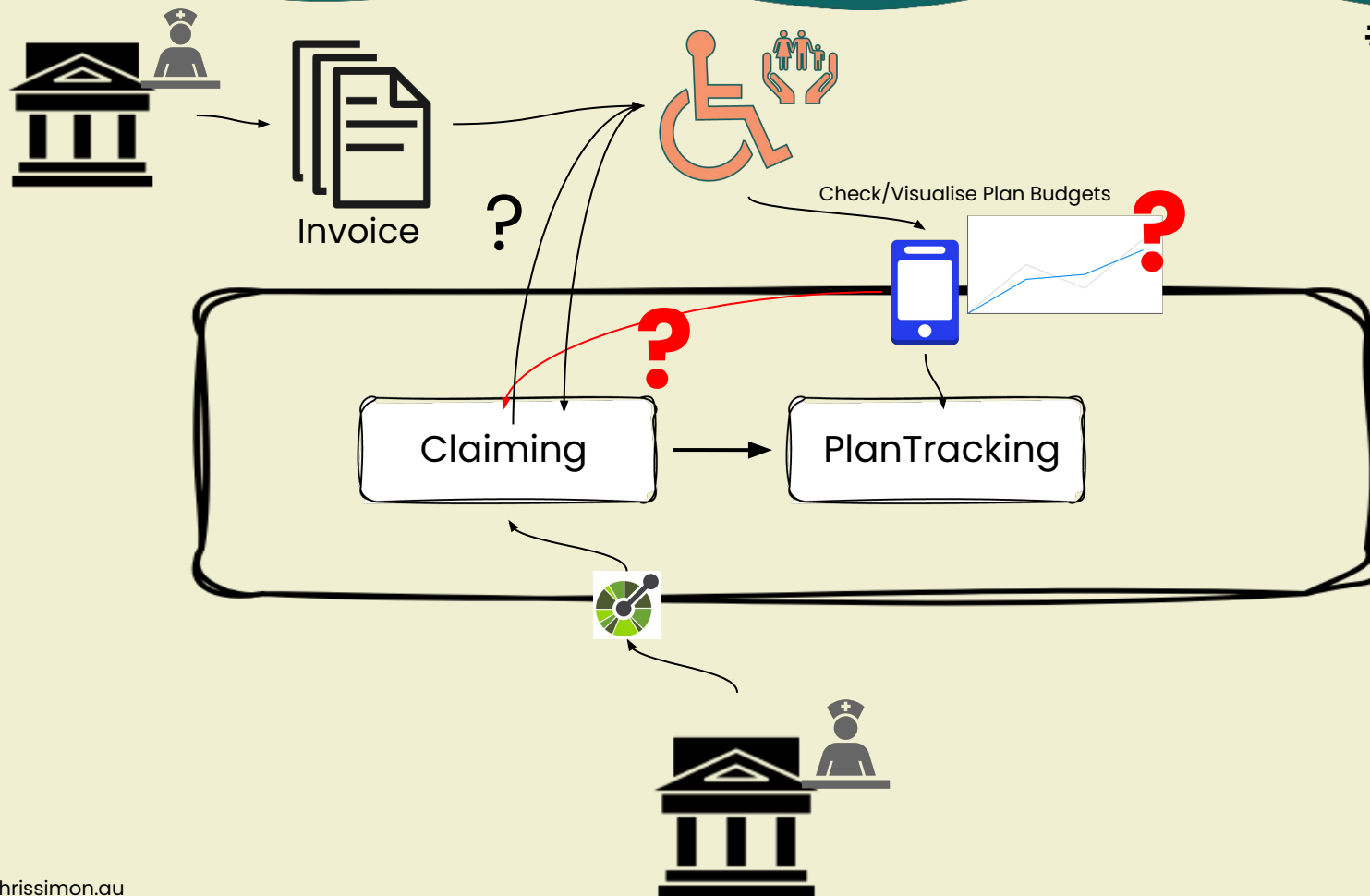


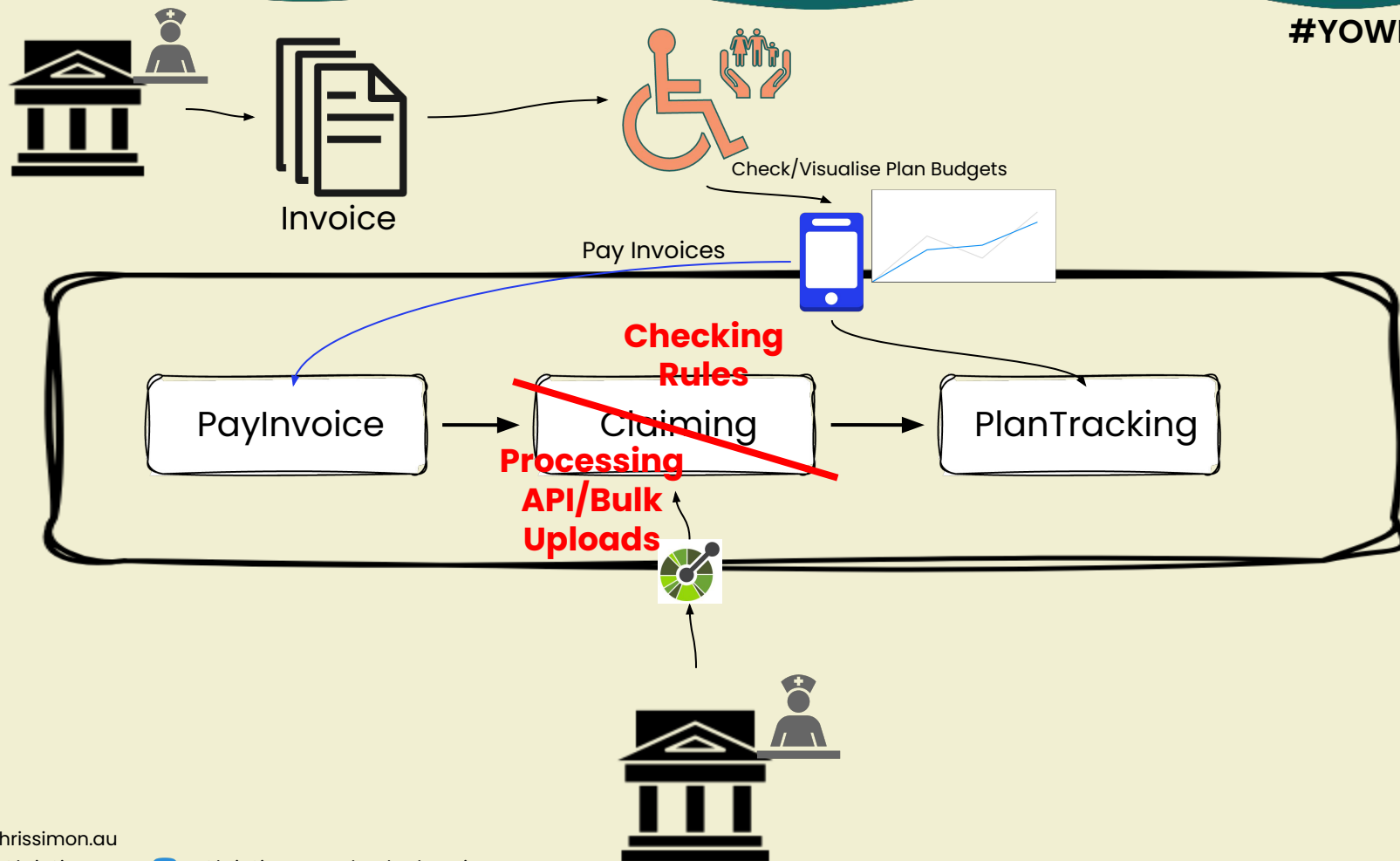
Dependencies

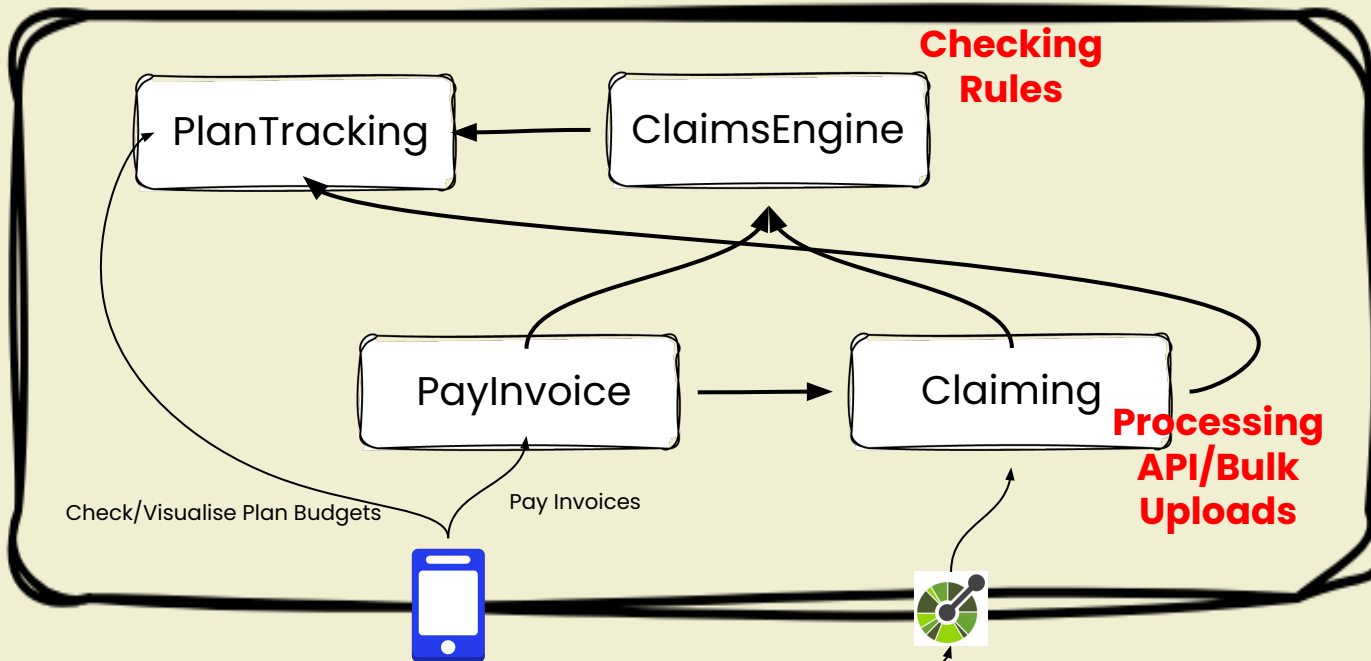


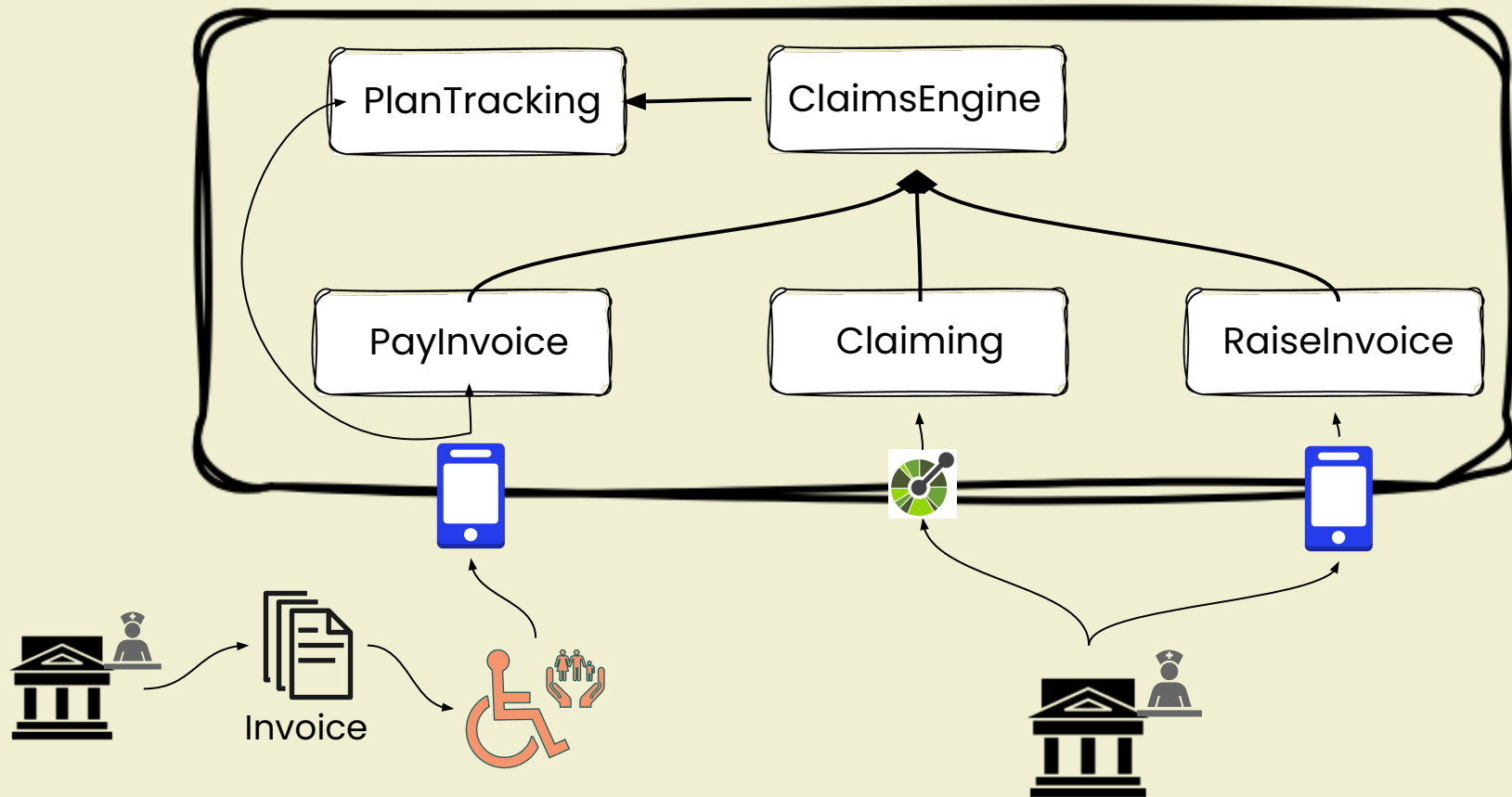


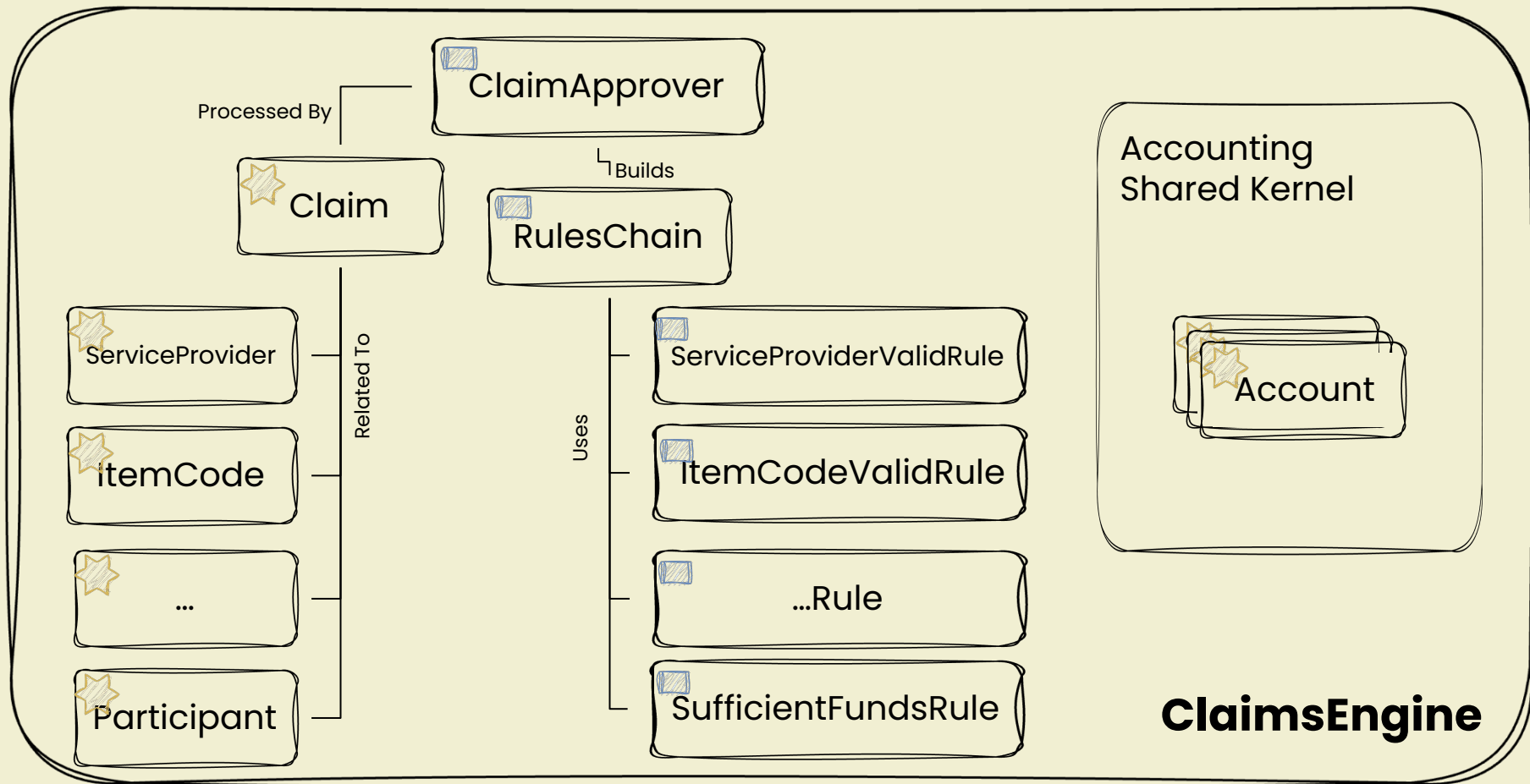













Accounting Shared Kernel

Program
(Liability)

 Account

Balance: \$0

Support Category
(Liability)

 Account

Balance: \$1500

Provider Payable
(Liability)

 Account

Balance: \$0

 Journal
(Claim Approved)

JournalEntry


Debit: \$50

Balance: \$1450

JournalEntry

Credit: \$50

Balance: \$50

 Journal
(Claim Paid)

JournalEntry

Debit: \$50

Balance: \$-50

JournalEntry

Debit: \$50

Balance: \$0

What is a database transaction?

Asked 13 years, 11 months ago Modified 15 days ago Viewed 136k times



Can someone provide a straightforward, yet not oversimplified explanation of a transaction as applied to computing?

173



database

concurrency

transactions

theory

failover



Share Edit Follow Close Flag

edited May 1 at 21:33

community wiki
7 revs, 5 users 40%
Vlad Gudim

[Add a comment](#)

[Start a bounty](#)

14 Answers

Sorted by: Highest score (default)



334



A transaction is a unit of work that you want to treat as "a whole." It has to either happen in full or not at all.

A classical example is transferring money from one bank account to another. To do that you have first to withdraw the amount from the source account, and then deposit it to the destination account. The operation has to succeed in full. If you stop halfway, the money will be lost, and that is Very Bad.



chrissimon.au

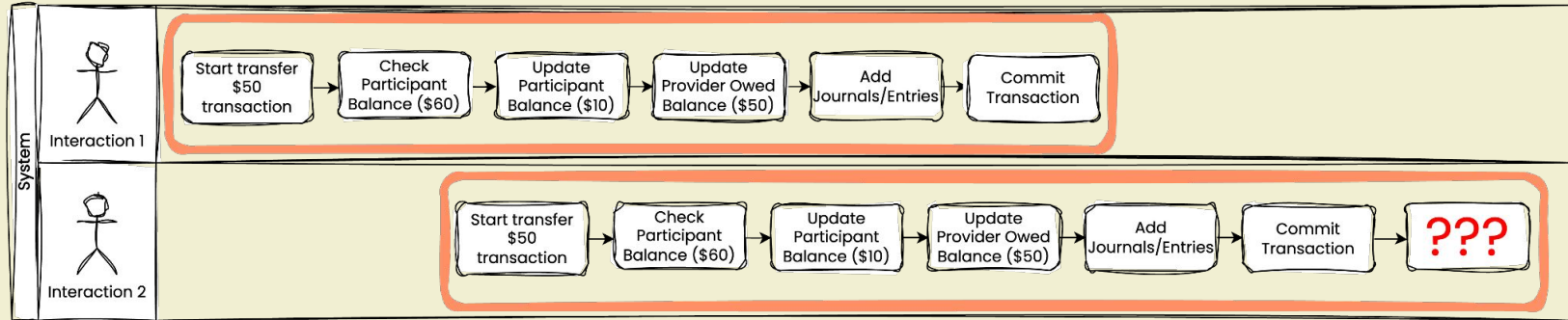


@ChrisSimonAu

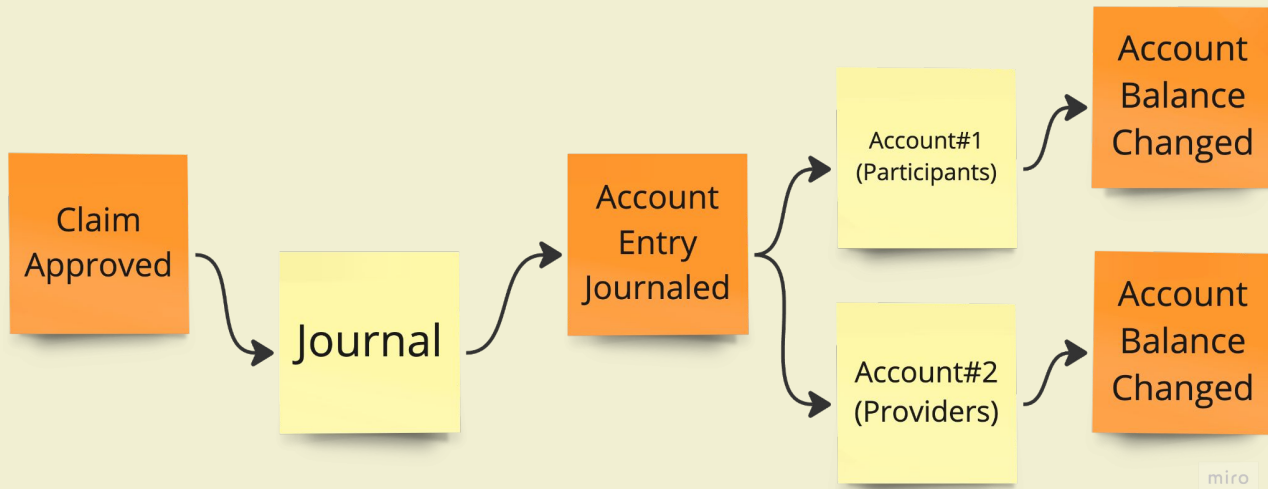


@ChrisSimonAu@hachyderm.io





1. Do nothing: **balances are incorrect**
2. Detect concurrent change and fail 2nd transaction: **terrible user experience**
3. Detect concurrent change and automatically retry 2nd transaction: **risk of delay in user experience (& how often do you retry?)**

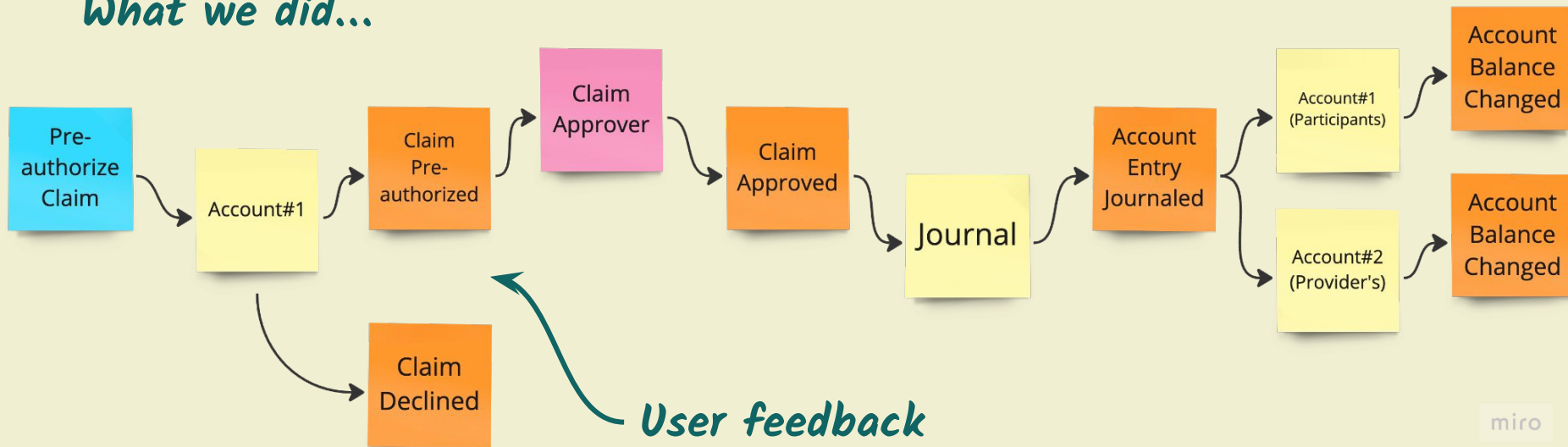


*But When Do
We Check the
Balance???*

*What if we approve the claim
(SufficientFundsRule) when the
balance is stale due to another
concurrent claim?*



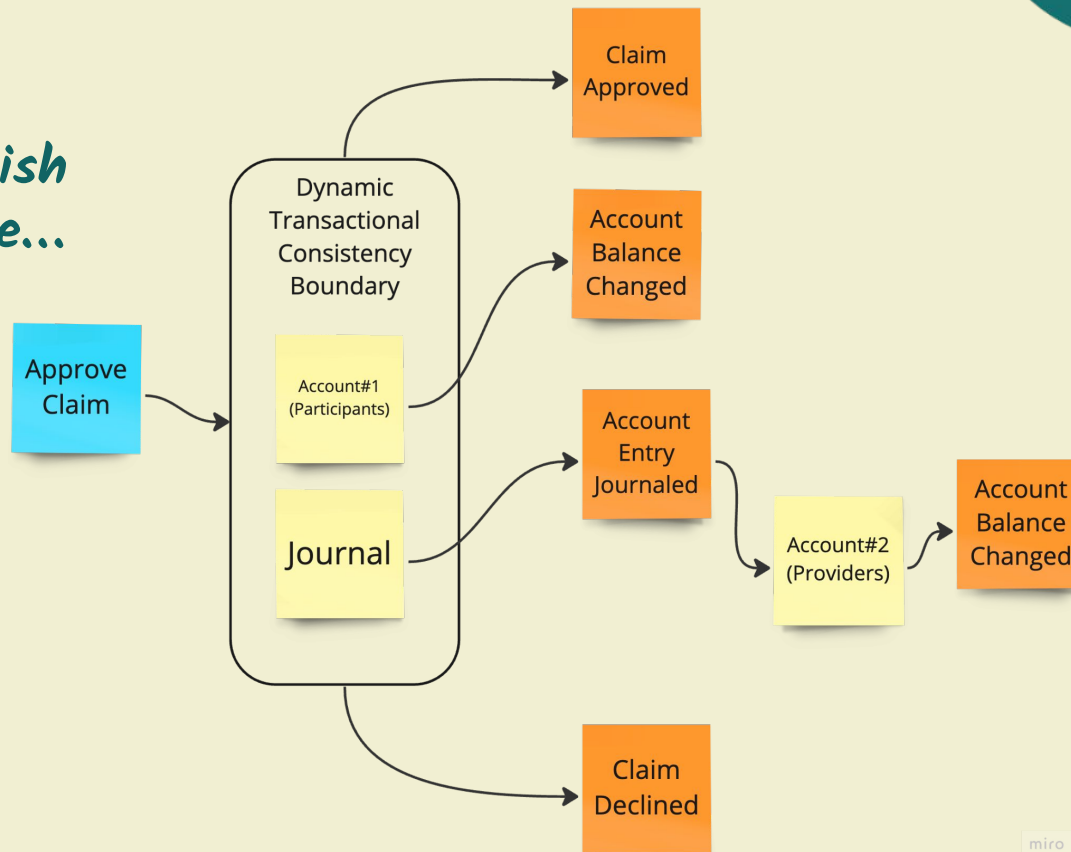
What we did...



miro

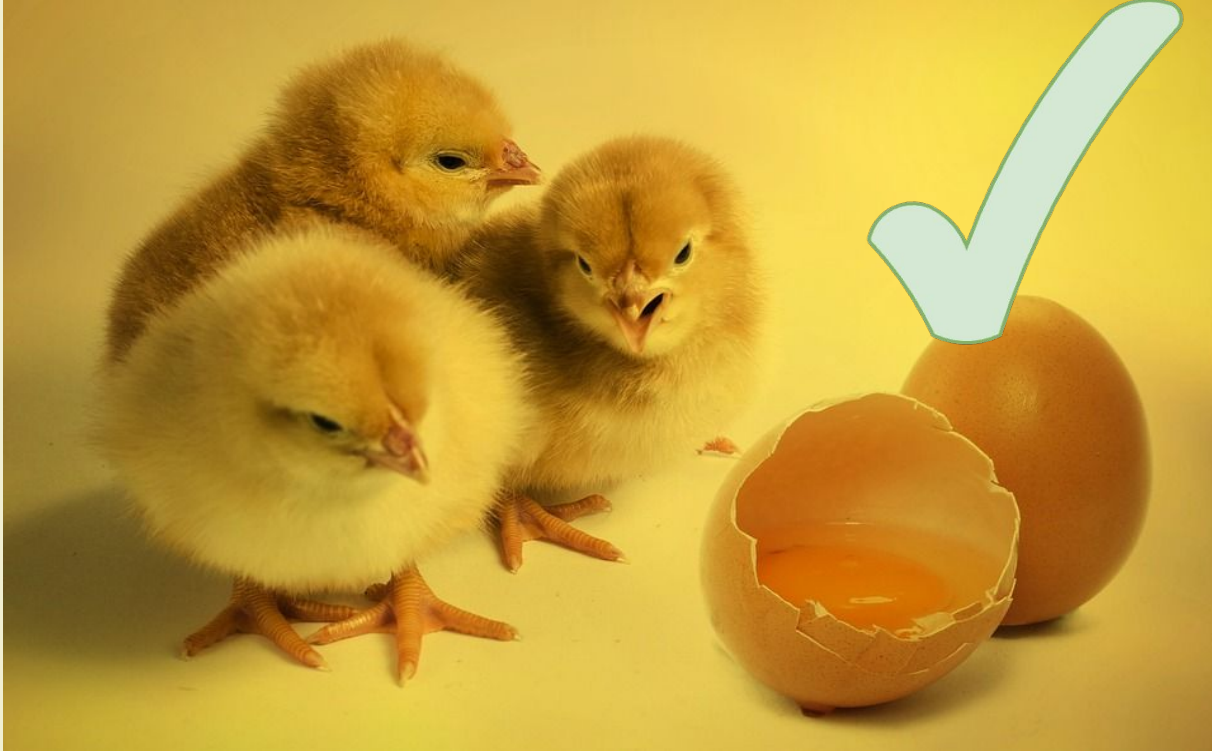


*What I wish
we'd done...*



Phase 2 - Validating the Initial Concepts

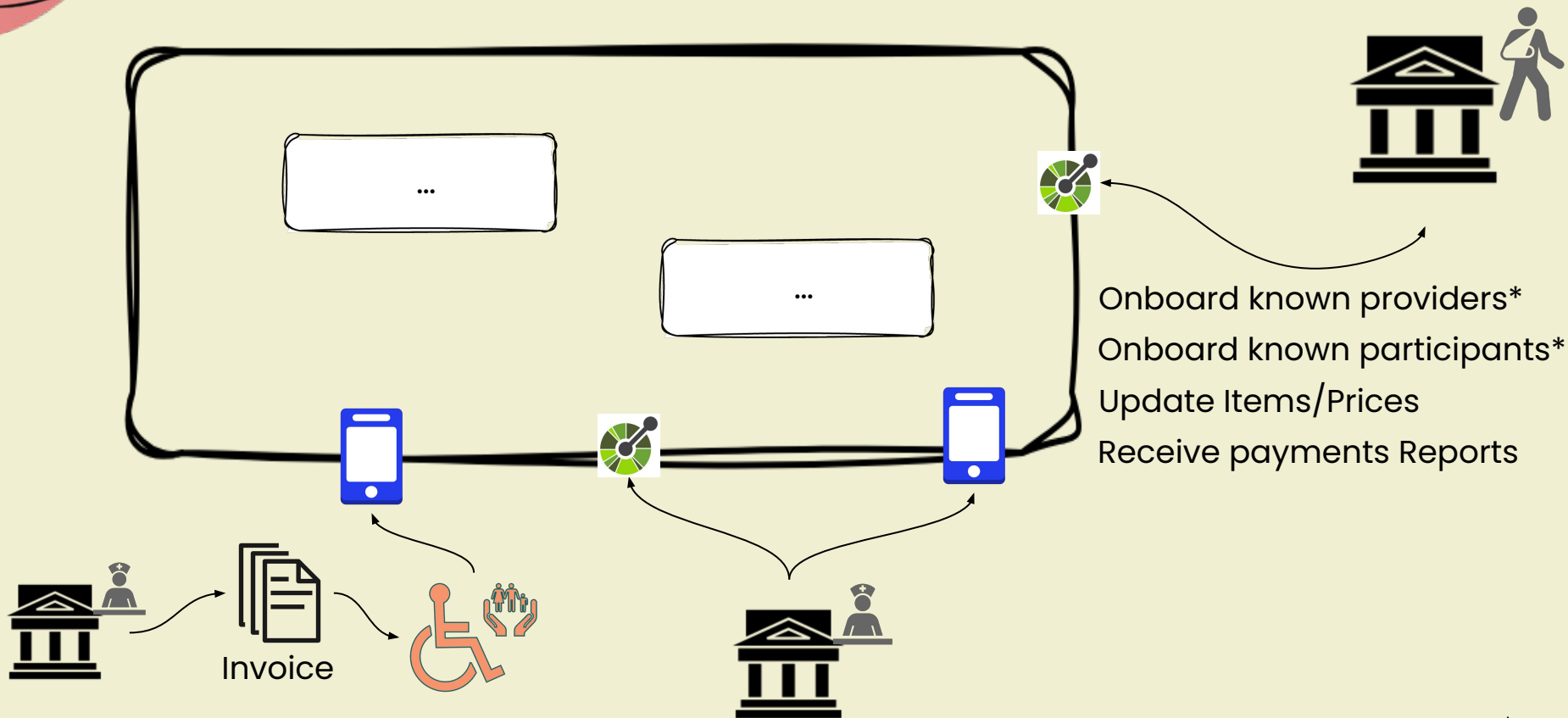




Second Goal:

Expand the pool of service providers who are enabled to provide services to people with a disability.

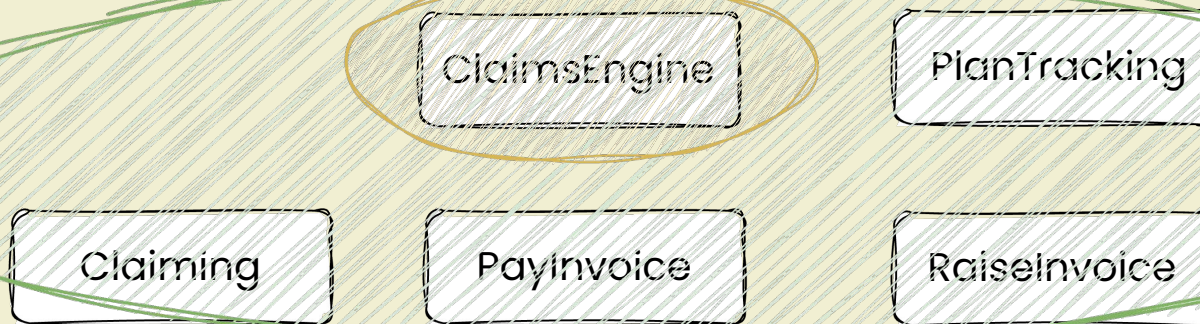
(To connect, what service providers wanted was more funding programs available.)



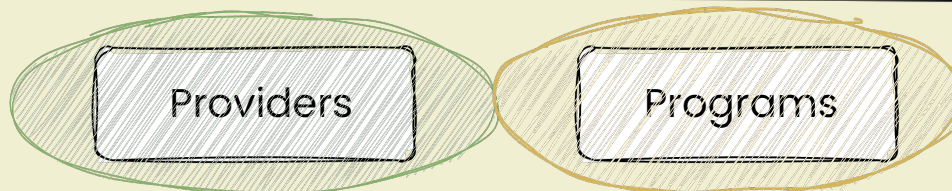
Finance



Operations



Admin

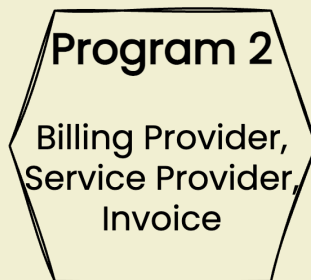
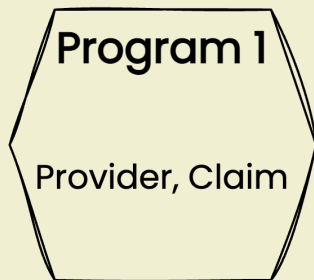


Term	Program 1	Program 2
Service Provider	Organisation providing supports (goods or services)	The entity providing the service - an individual when a registered healthcare provider, an organisation otherwise
Claim	Claim for payment by a provider for providing a single support	A funding block for an individual associated with a specific injury/event
Billing Provider		The taxable entity providing the service. Is the same as the provider if the provider is an organisation
Invoice		Request for payment by a biller for providing a collection of supports.

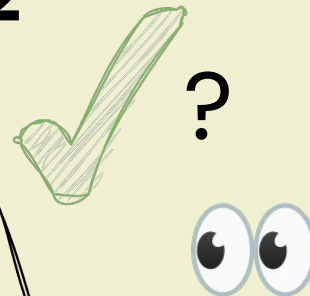




Option 1

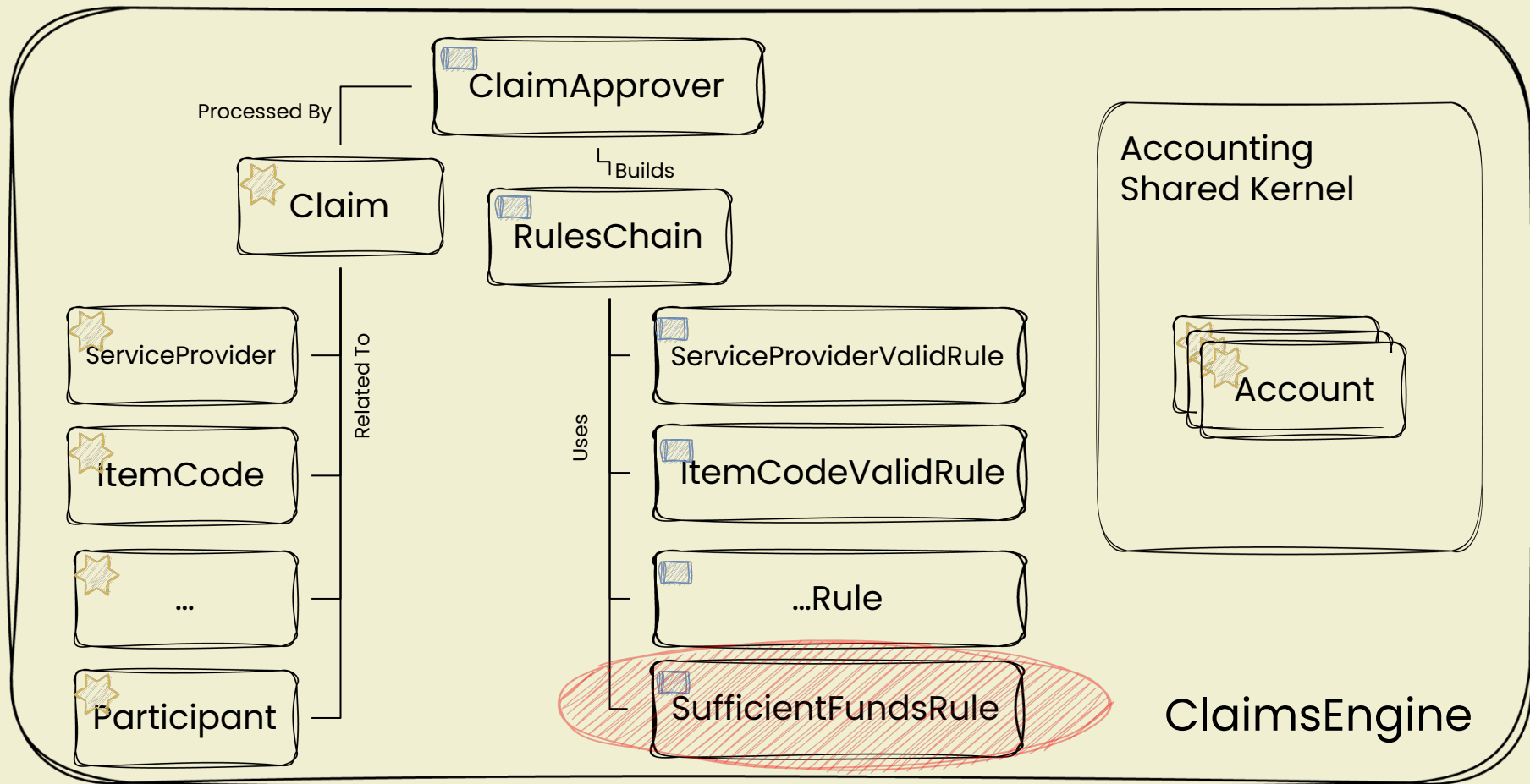


Option 2



Term	Definition
Program	Funding organisation (e.g. insurer, gov agency...)
Biller	The taxable entity providing the service. Is the same as the provider if the provider is an organisation
Provider	The entity providing the service - an individual when a registered healthcare provider, an organisation otherwise
Invoice	Collection of claims submitted at one time
Claim	Claim for payment by a provider for providing an item or service
Client	Individual receiving services
Membership	Relationship between the individual and the program







BalanceCheckRequired:
true/false

Balance: -\$100

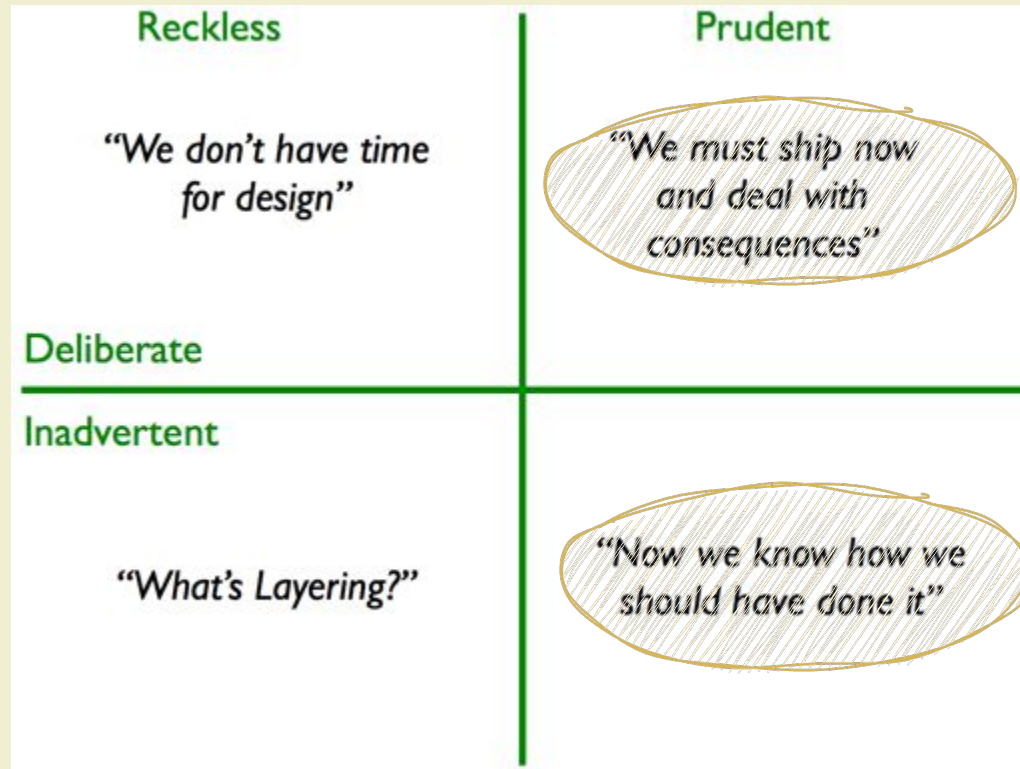




I'm never in favor of writing code poorly, but I am in favor of writing code to reflect your current understanding of a problem even if that understanding is partial.

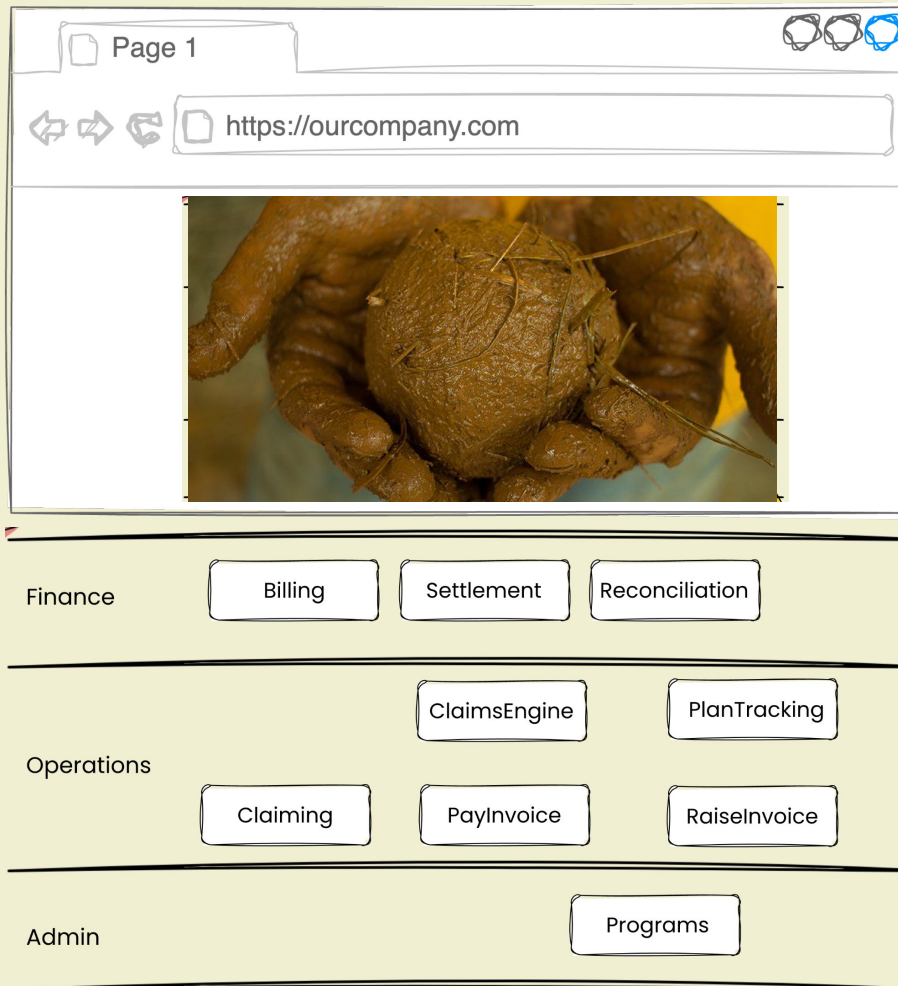


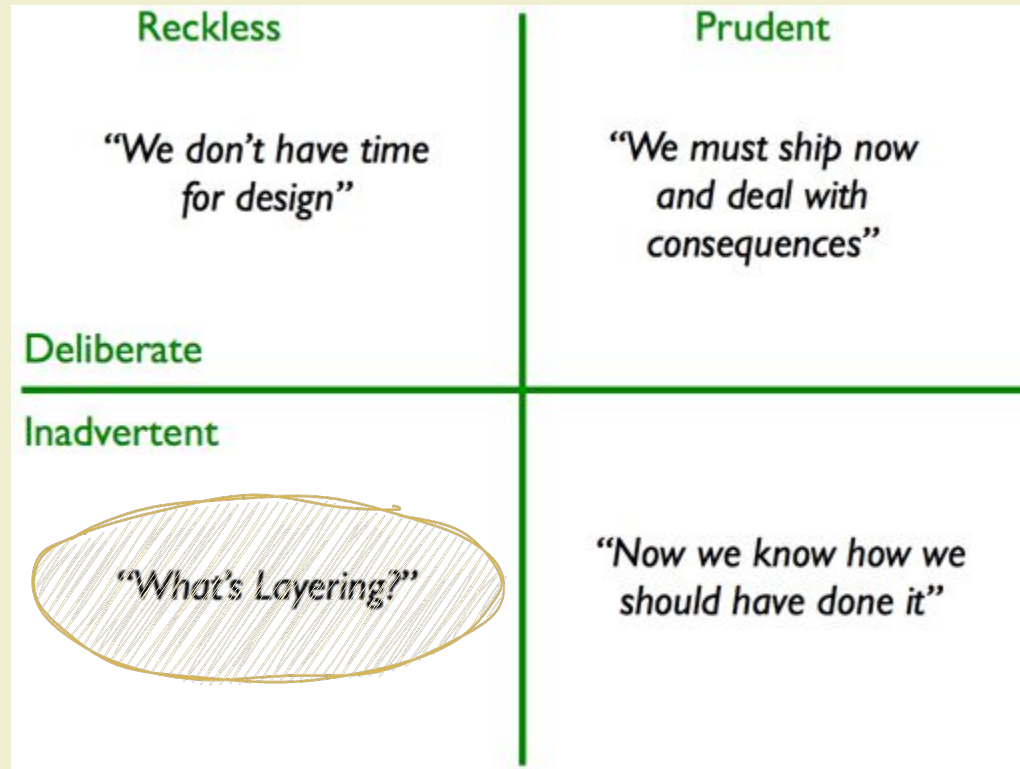
- **Ward Cunningham, 2009**
<https://www.youtube.com/watch?v=pqeJFYwnkjE>



<https://martinfowler.com/bliki/TechnicalDebtQuadrant.html>





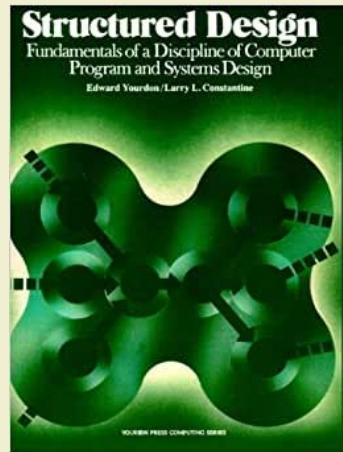


<https://martinfowler.com/bliki/TechnicalDebtQuadrant.html>





*Coupling as an abstract concept — the degree of interdependence between modules — may be operationalized as the probability that in **coding, debugging, or modifying one module**, a programmer will have to take into account **something about another module**.*



- **Constantine & Yourdon, Structured Design, 1979**





Page 1

https://ourcompany.com

```
body {  
  font-family: "Poppins", sans-serif;  
  font-size: 16px;  
  color: ■ #073737;  
}
```

Finance

Billing

Settlement

Reconciliation

Operations

ClaimsEngine

PlanTracking

Claiming

PayInvoice

RaiseInvoice

Admin

Programs



UX Change



Finance

Billing

Settlement

Reconciliation

Operations

ClaimsEngine

PlanTracking

Claiming

PayInvoice

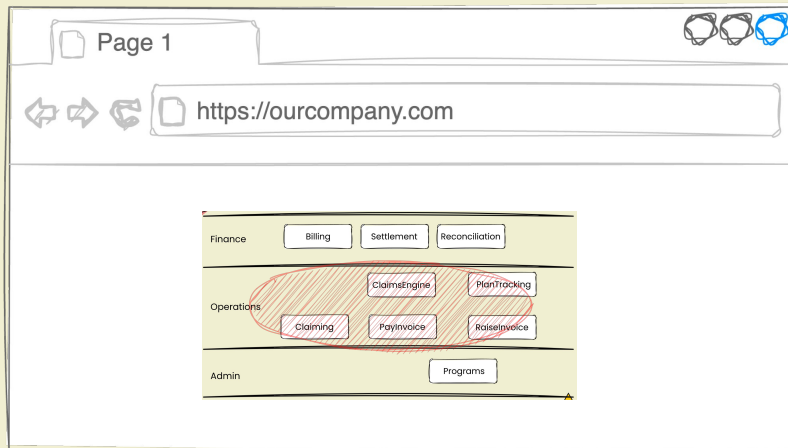
RaiseInvoice

Admin

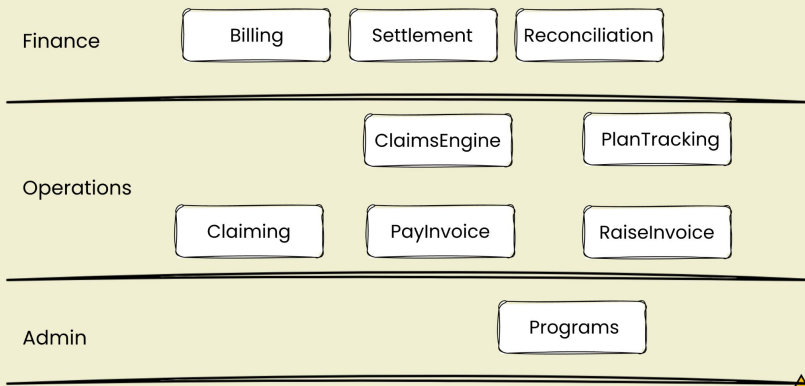
Programs

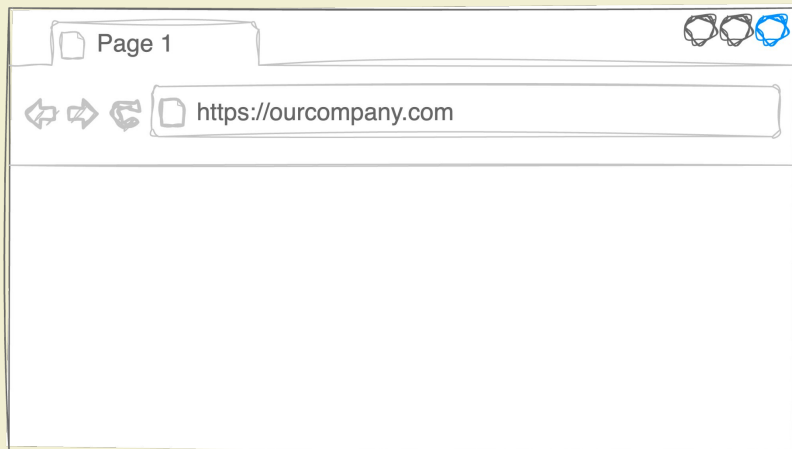


Rules Change



UI Process Change





Finance

Billing

Settlement

Reconciliation

Operations

ClaimsEngine

PlanTracking

Claiming

PayInvoice

RaiseInvoice

Admin

Programs

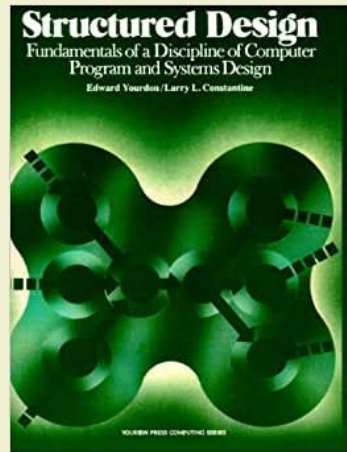


Core Data Model Changes

chrissimon.au[@ChrisSimonAu](https://twitter.com/ChrisSimonAu)@ChrisSimonAu@hachyderm.io



Coupling as an abstract concept — the degree of interdependence between modules — may be operationalized as the probability that in coding, debugging, or modifying one module, a programmer will have to take into account something about another module.



- **Constantine & Yourdon, Structured Design, 1979**



Finance

Billing

Settlement

Reconciliation

Operations

ClaimsEngine

PlanTracking

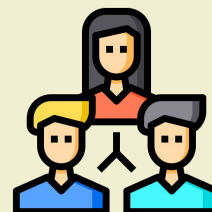
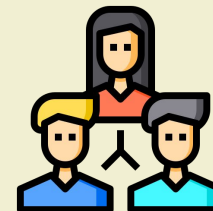
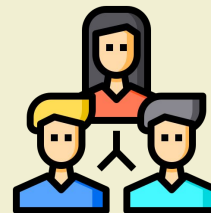
Claiming

PayInvoice

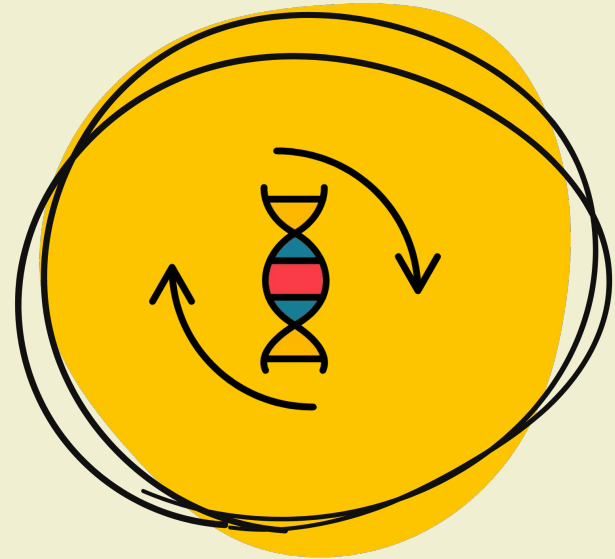
RaiseInvoice

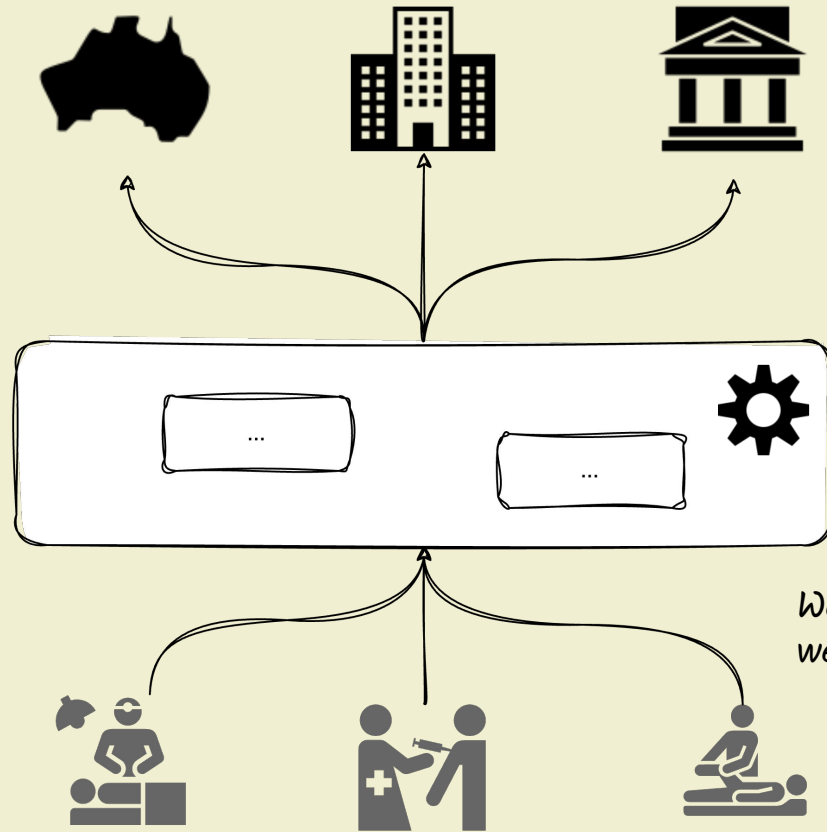
Admin

Programs



Phase 3 - Evolving the Initial Concept



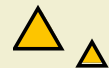
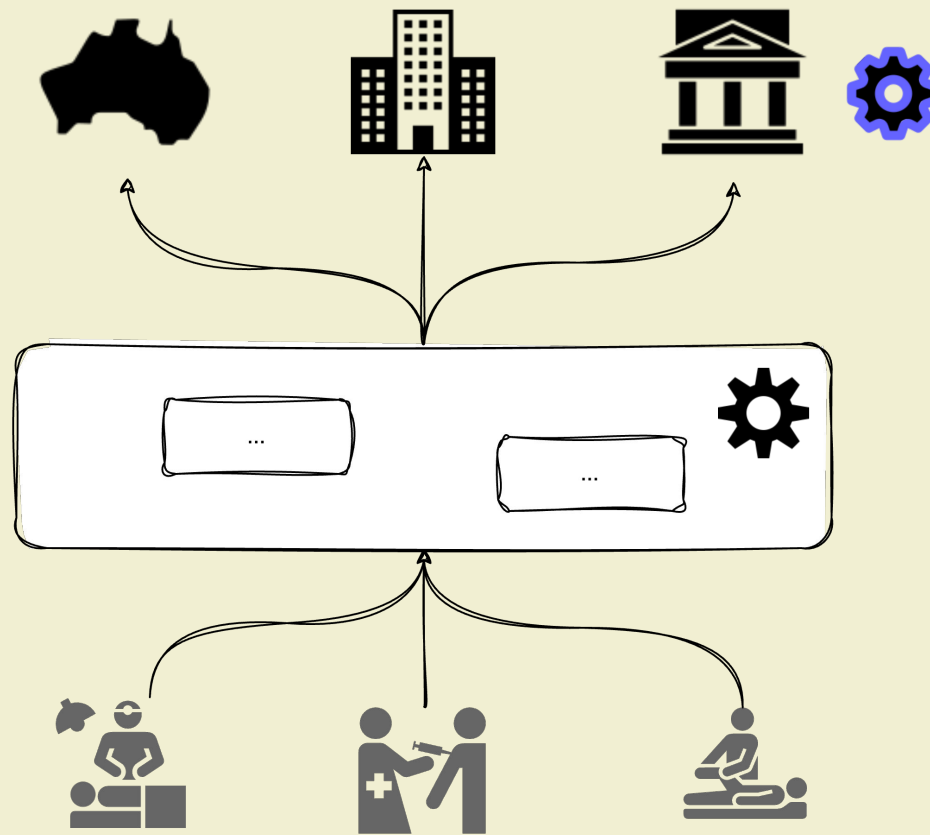


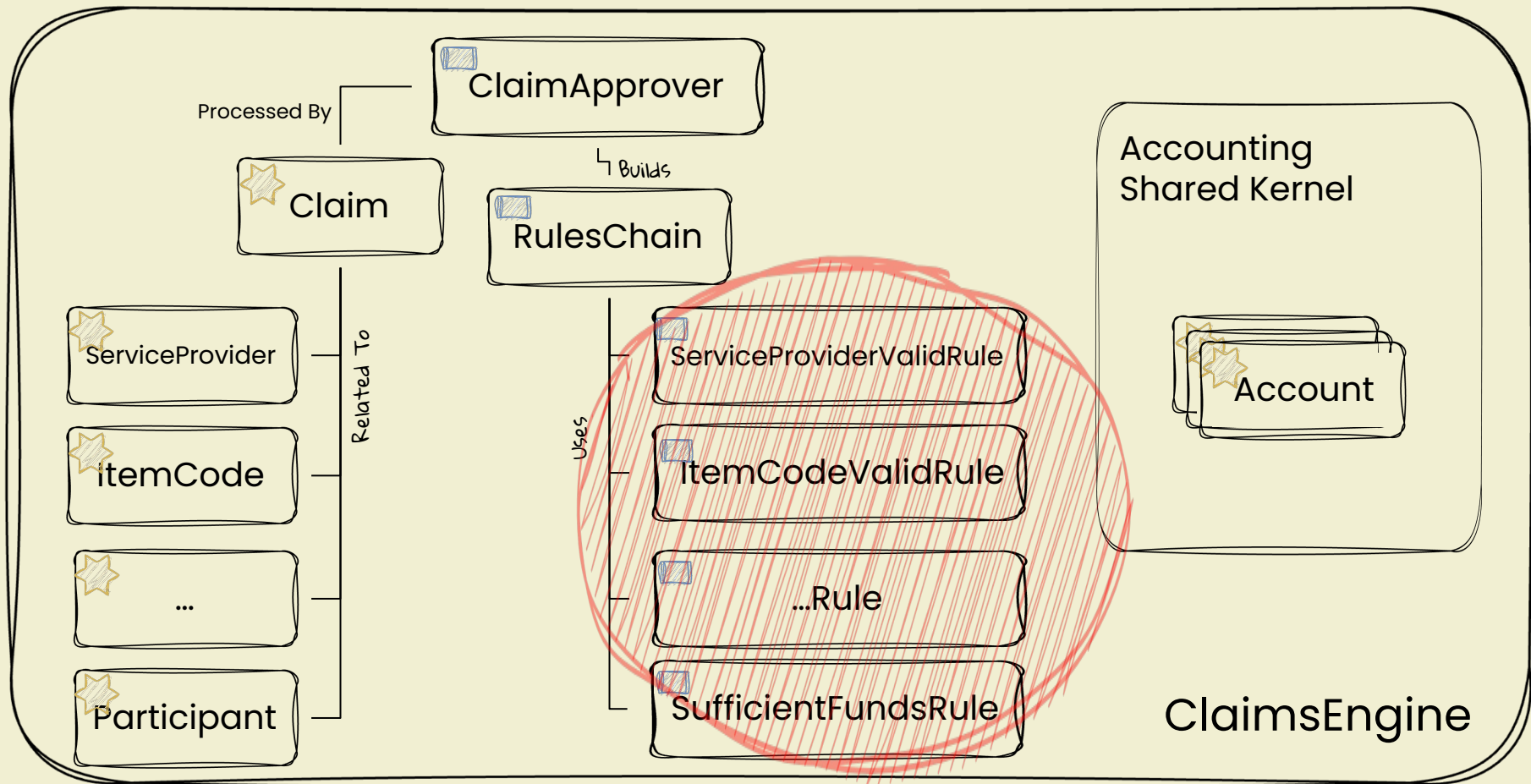
They did XYZ, and we checked all your rules and they are ok, can you send them \$50?

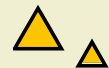
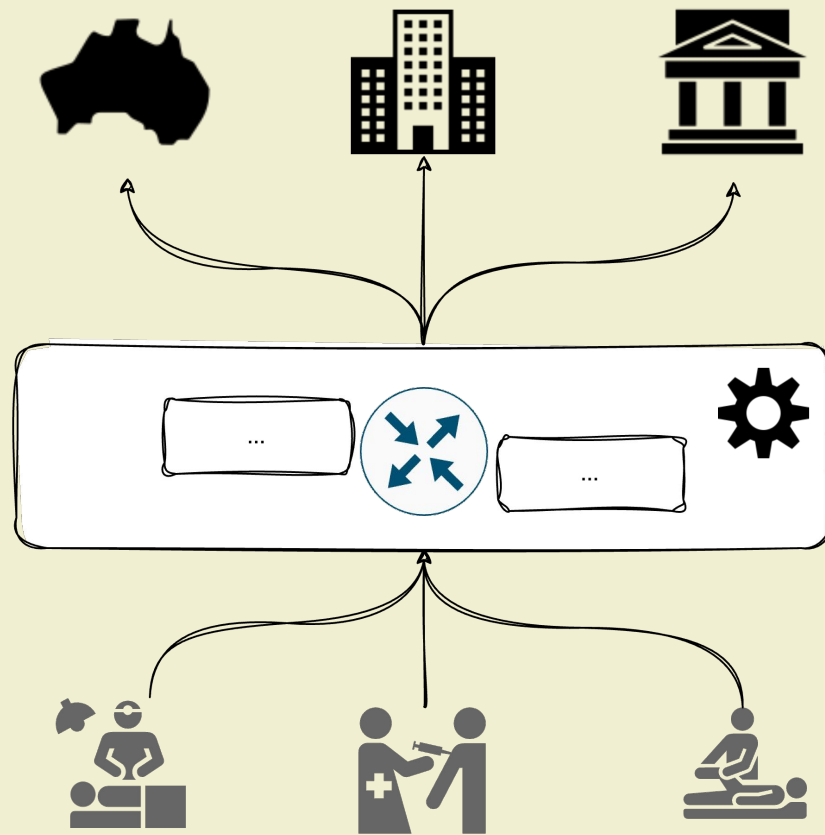
We did XYZ, can we have \$50?

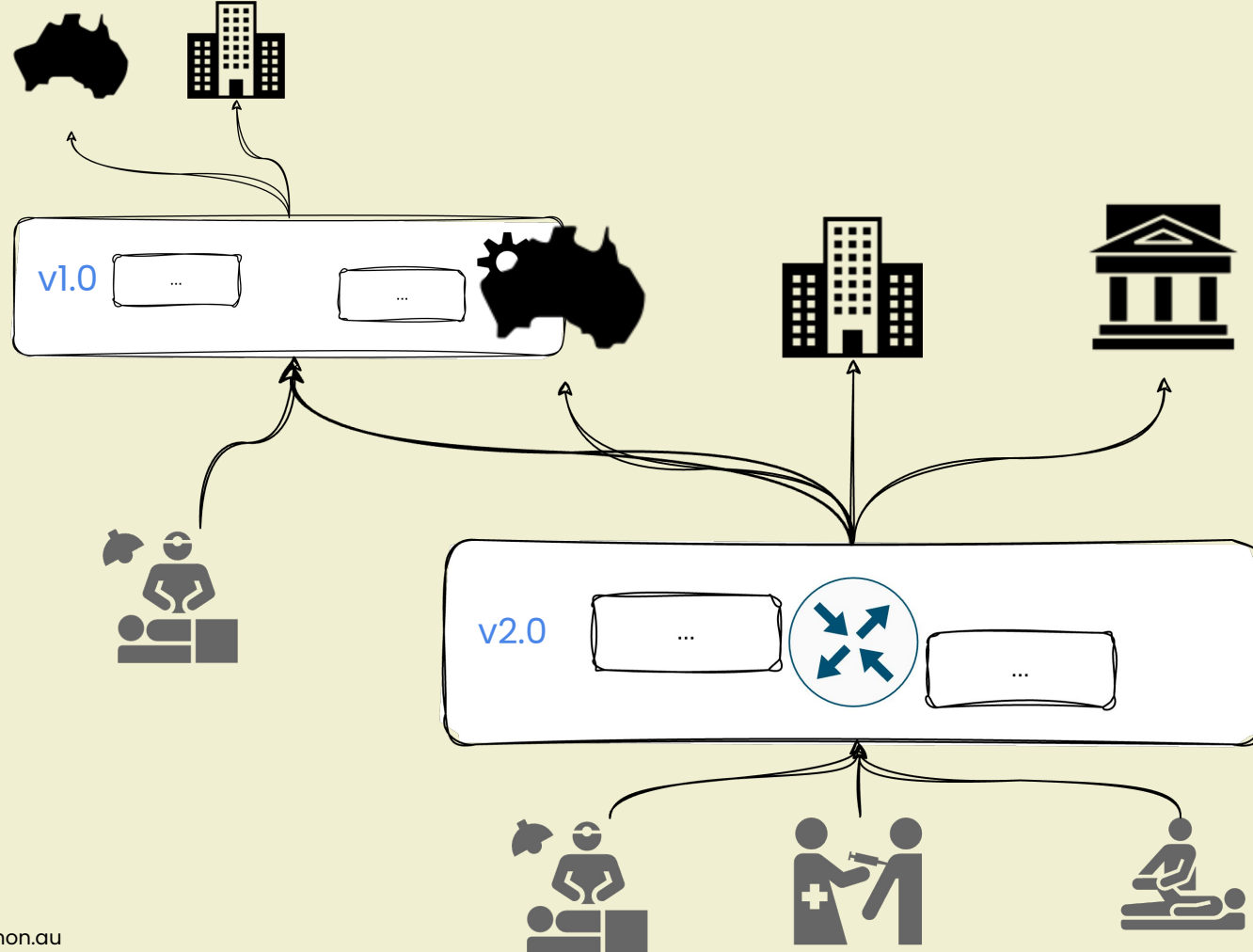
Sorry, we can't pay you because " " ...

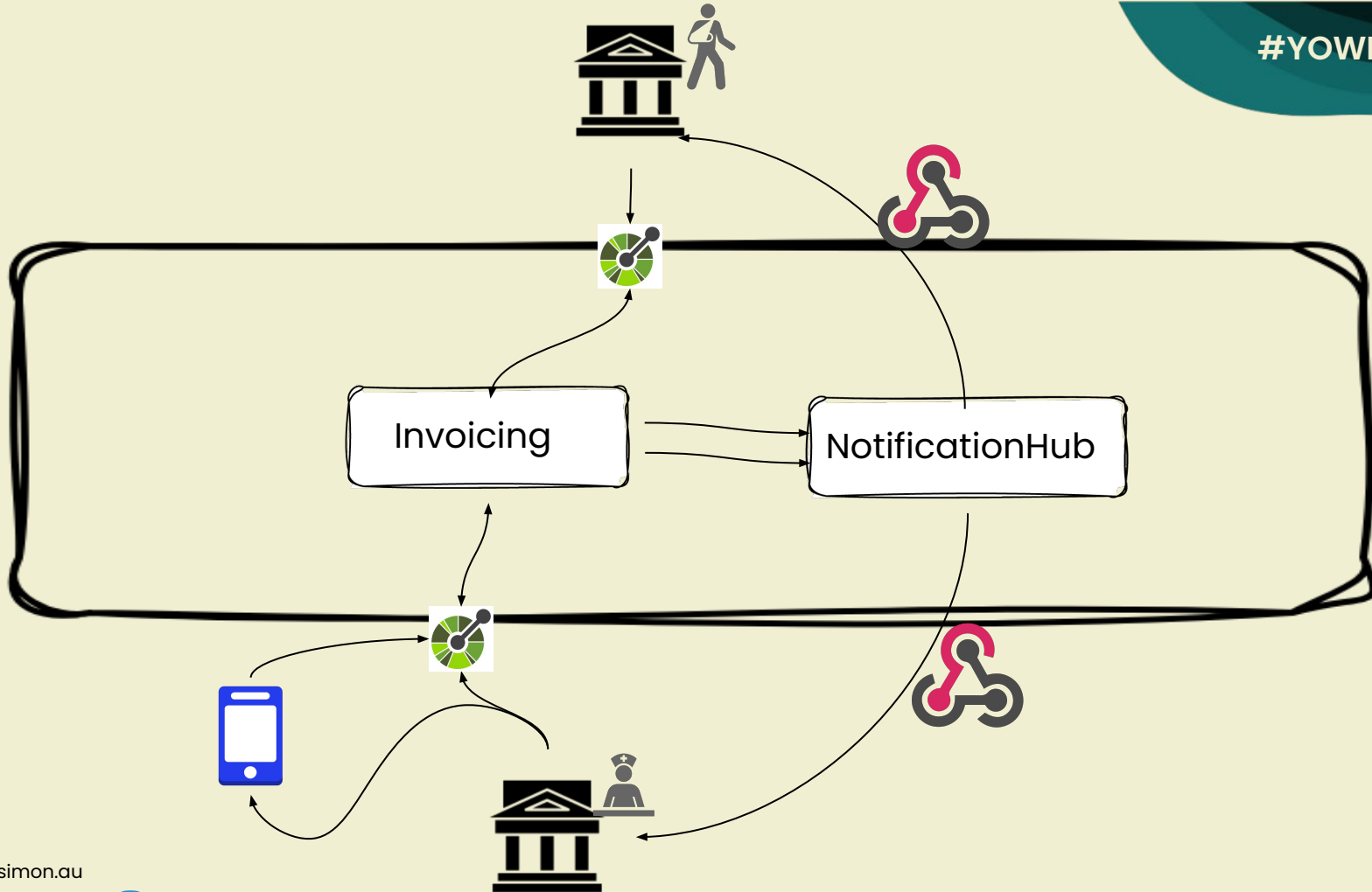








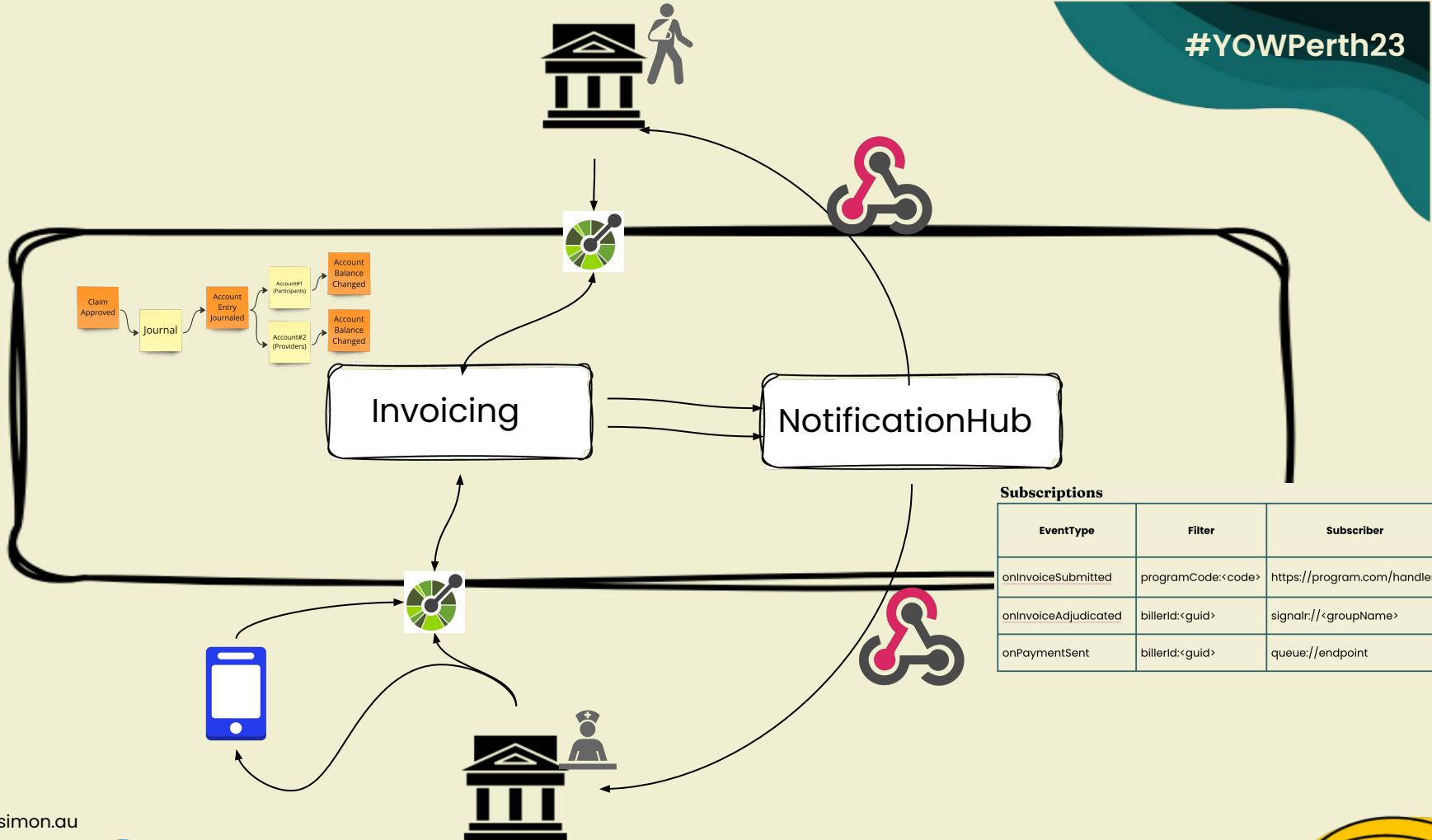




NotificationHub Subscriptions

EventType	Filter	Subscriber
onInvoiceSubmitted	programCode:<code>	https://program.com/handler
onInvoiceAdjudicated	billerId:<guid>	signalr://<groupName>
onPaymentSent	billerId:<guid>	queue://endpoint





Lesson(s) Learnt





A team committed to EVOLVING ORDER must fearlessly rethink the large-scale structure throughout the project life cycle. The team should not saddle itself with a structure conceived of early on, when no one understood the domain or the requirements very well.

*Unfortunately, that evolution means that your final structure will not be available at the start, and that means that you will have to refactor to impose it as you go along. **This can be expensive and difficult, but it is necessary.***

– **Eric Evans, 2003**





#YOWPerth23

DOMAIN DRIVEN DESIGN EUROPE

Strategic Domain-Driven Design

4 half days workshop

Chris Simon




10% Discount Code

<https://ti.to/on3/dddeutr23/with/7jy5nu6hti4/discount/Chris>

 chrissimon.au

 @ChrisSimonAu

 @ChrisSimonAu@hachyderm.io



Domain-Driven Design (DDD) Australia

📍 Sydney, Australia

👤 367 members · Public group

👤 Organized by **Chris Simon** and 1 other



Share:     

[About](#)[Events](#)[Members](#)[Photos](#)[Discussions](#)[More](#)[Join this group](#)

What we're about

Welcome Domain-Driven Design practitioners & aficionados! This group is for software & product people interested in the world of domain-driven design. Covering topics from modeling to implementation, socio-technical systems, patterns, practices & heuristics, we'll have a range of talks, panel discussions and collaborative interactive sessions to collectively grow as practitioners of domain-driven design.

Organizers



Chris Simon and 1 other

[Message](#)


Members (367)

[See all](#)

<https://www.meetup.com/en-AU/domain-driven-design-australia/>

 chrissimon.au

 @ChrisSimonAu

 @ChrisSimonAu@hachyderm.io





Contextive

Get on the same page.



`[ApiController]``[Route("[controller]")]`

0 references

```
public class StudentsController : ControllerBase
{
```

4 references

```
    private readonly UniversityContext context;
```

0 references

```
    public StudentsController(UniversityContext context)
    {
        this.context = context;
    }
```

`[HttpPost]`

0 references


```
    public async Task<ActionResult<Student>> Register([FromBody] RegisterStudentRequest request)
    {
```

```
class ChrisSimonAu.UniversityApi.Students.RegisterStudentRequest
```

University Context


Vision: To allow students to register and enrol in courses.

 **student** : A person who would like to study a course.

 **register** : The process of a student registering with the university. They will not yet be enrolled in a course after registering.

student Usage Examples:

 "The student has registered at the university."

 "The student has enrolled in the course."

register Usage Examples:





chris < 1 minute ago

We're implementing the Leg Policy next, can we have a chat to confirm the prioritisation of rules in the routing specification?

1 reply

New



Contextive APP < 1 minute ago

Cargo Reminders

Leg

The movement of a Cargo on a specific vessel from load location to unload location.

Update Definition

See Usage Examples

Usages in Other Contexts

Policy

A set of rules that the routing service must follow when evaluating legs that confirm to the desired routing specification.

Update Definition

See Usage Examples

Usages in Other Contexts

Reminders for these terms won't be sent for the next 2 weeks.

Change Reminder Frequency



chrissimon.au



@ChrisSimonAu



@ChrisSimonAu@hachyderm.io





Contextive

Get on the same page.



<https://contextive.tech/>



Let's Connect



chrissimon.au



[chrissimon-au](https://www.linkedin.com/company/chrissimon-au)



[@ChrisSimonAu](https://twitter.com/ChrisSimonAu)



[@ChrisSimonAu@
hachyderm.io](mailto:@ChrisSimonAu@hachyderm.io)

