









# The Ward Cunningham Quote

“When taking short cuts and delivering code that is not quite right... a development team incurs Technical Debt. This debt decreases productivity. This loss of productivity is the interest.”



**Technical Debt:** A name that implies a debt owed by tech that should be repaid by tech.



# The Metaphor Purpose

Used to explain the need for "refactoring" to a finance-literate boss at OOPSLA '92.

*That was 34 years ago—context has changed, but the word remains.*

# The Ward Cunningham Quote

“When taking short cuts and delivering code that is not quite right... a development team incurs Technical Debt. This debt decreases productivity. This loss of productivity is the interest.”



**Technical Debt:** A name that implies a debt owed by tech that should be repaid by tech.



# The Metaphor Purpose

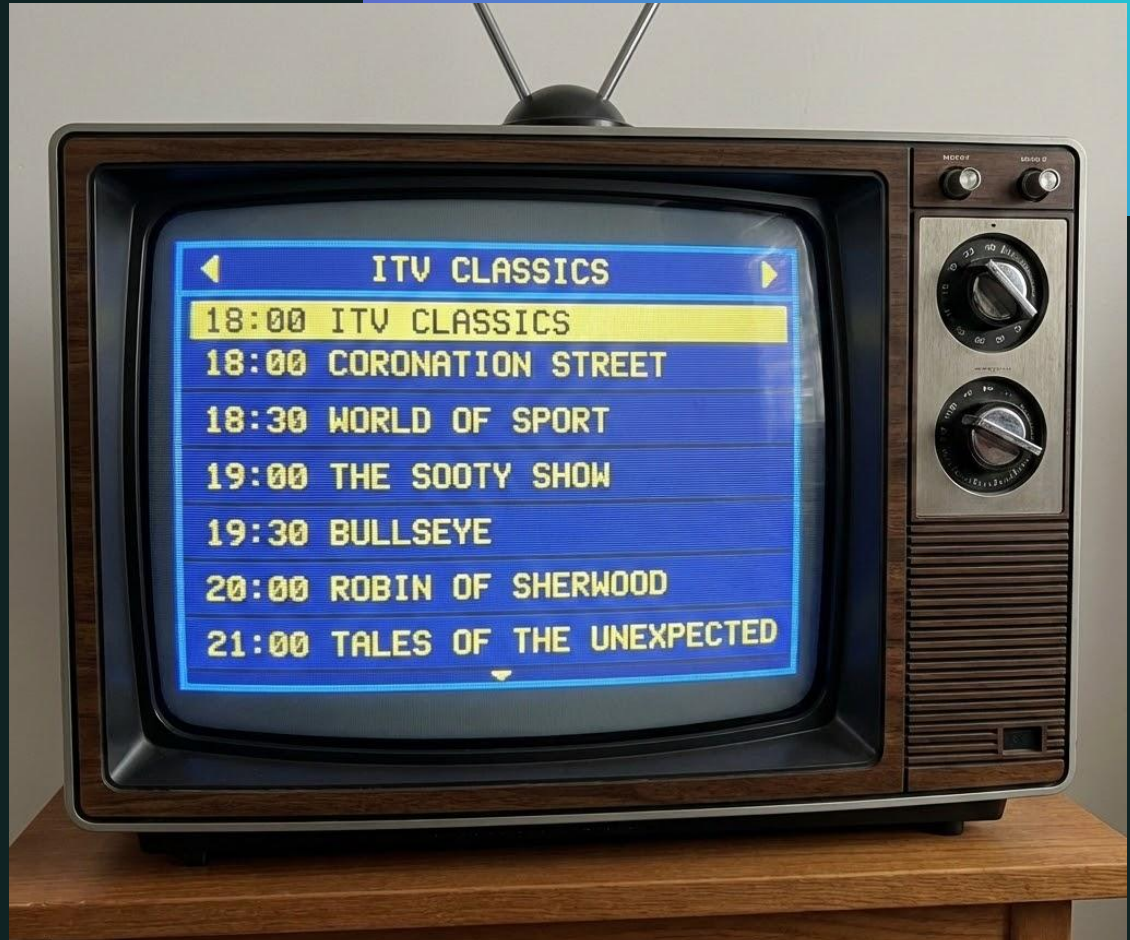
Used to explain the need for "refactoring" to a finance-literate boss at OOPSLA '92.

*That was 34 years ago—context has changed, but the word remains.*

# Legacy Heritage

## 25 Years Old & Mission Critical

- Keeping the screen from going blank



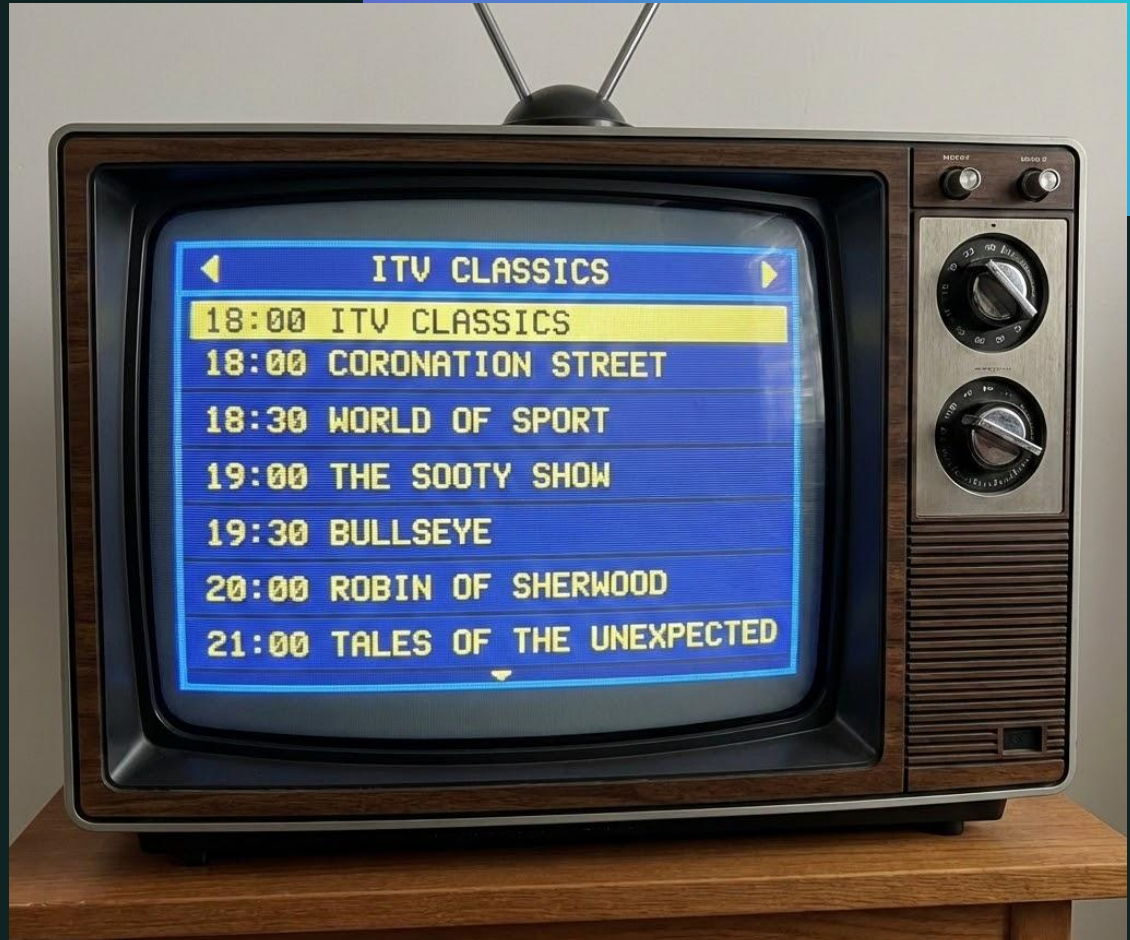
# Legacy Heritage

## 25 Years Old & Mission Critical

- Keeping the screen from going blank

## Non domain relevant

- Non-core modules added over time



# Legacy Heritage

## 25 Years Old & Mission Critical

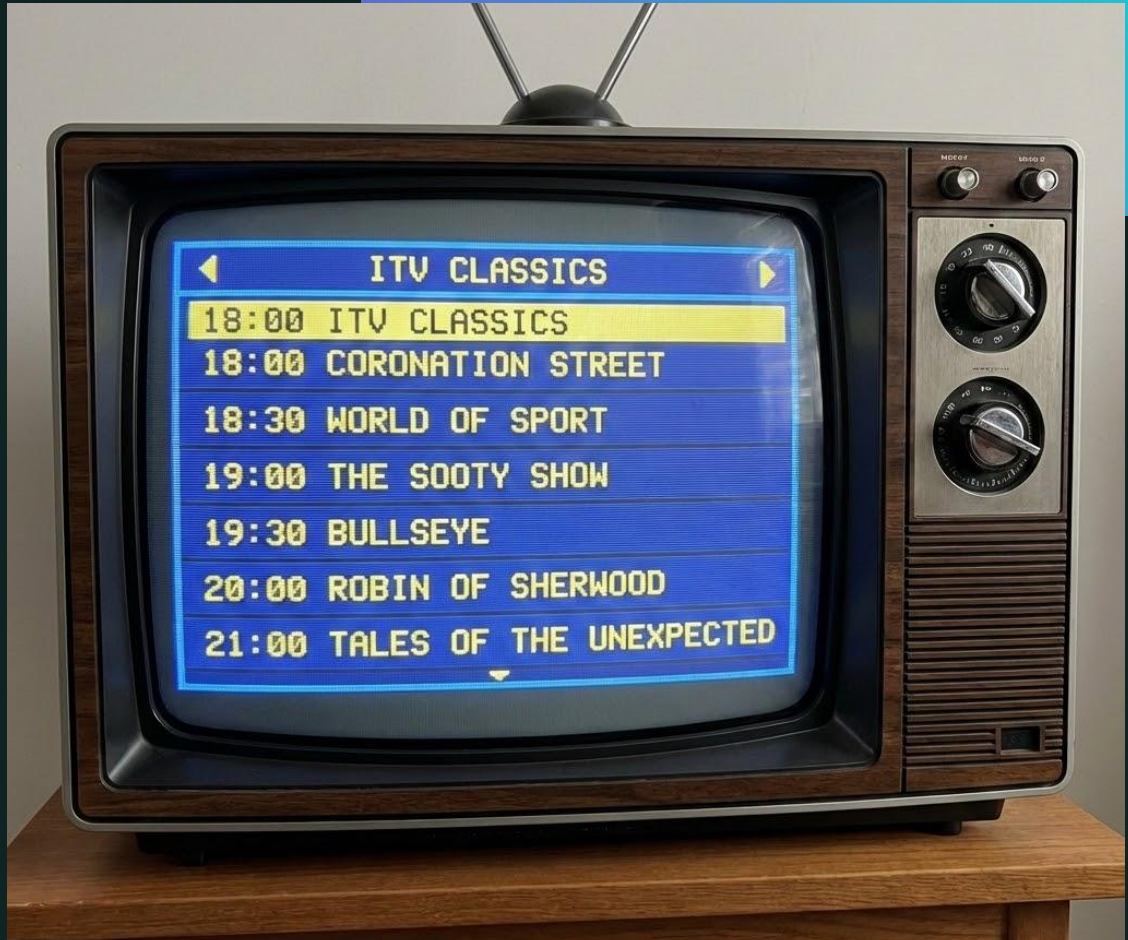
- Keeping the screen from going blank

## Non domain relevant

- Non-core modules added over time

## Innovation blocker

- Complexity actively blocking innovation



# Buy vs Build

## The Ticking Clock

- An aging technology stack hitting End of Support (EoS)



# Buy vs Build

## The Ticking Clock

- An aging technology stack hitting End of Support

## The COTS Illusion

- Customized workflows make a 3rd-party swap impossible



# Buy vs Build

## The Ticking Clock

- An aging technology stack hitting End of Support (EoS)

## The COTS Illusion

- Customized workflows make a 3rd-party swap impossible

## Feature Creep

- A heritage system bloated with non-core functionality, decades of "it's quicker to put it here" decisions



# Buy vs Build

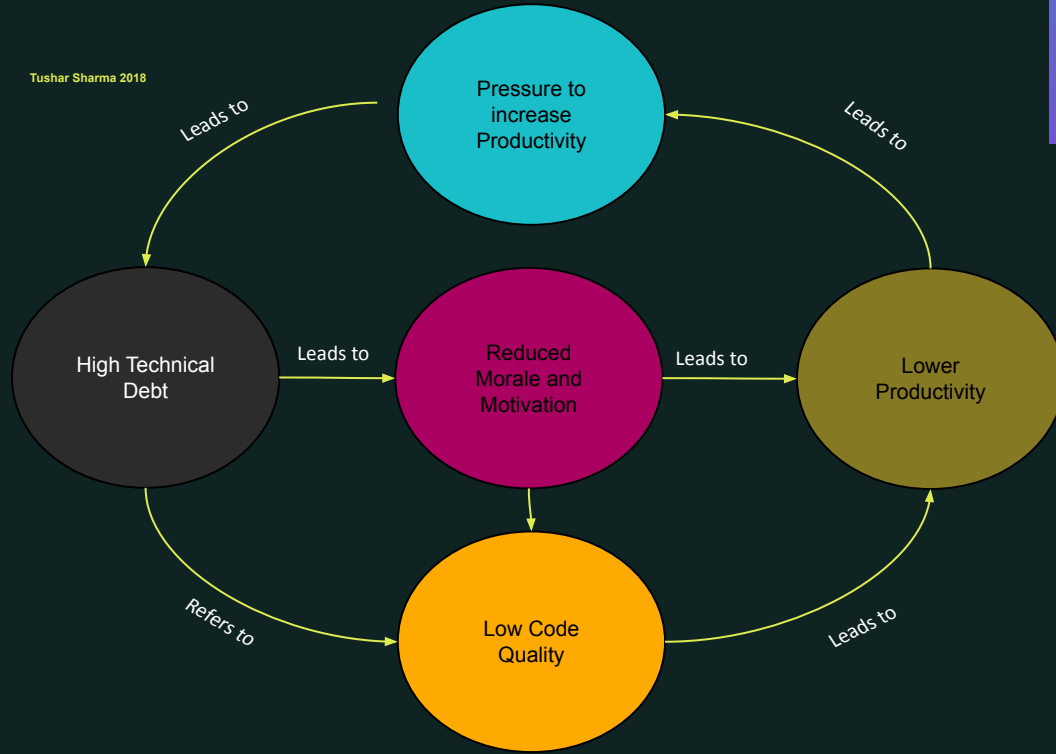
## The Strategy

- Rebuild the core, migrate the rest, and detangle the business.

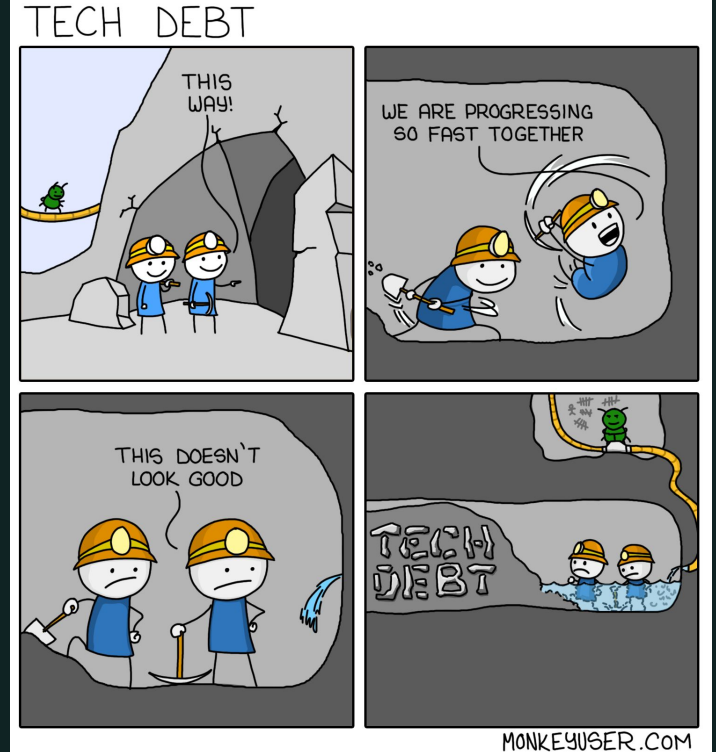


# So why bother thinking about Technical Debt?

Tushar Sharma 2018



Engineering ~~Spiral~~ Cycle of productivity...

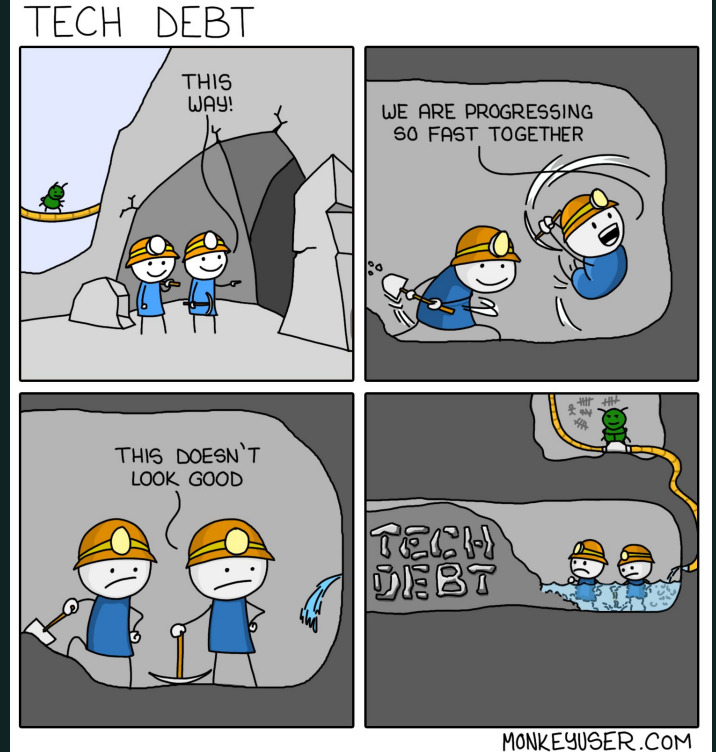


# So why bother thinking about Technical Debt?

Tushar Sharma 2018

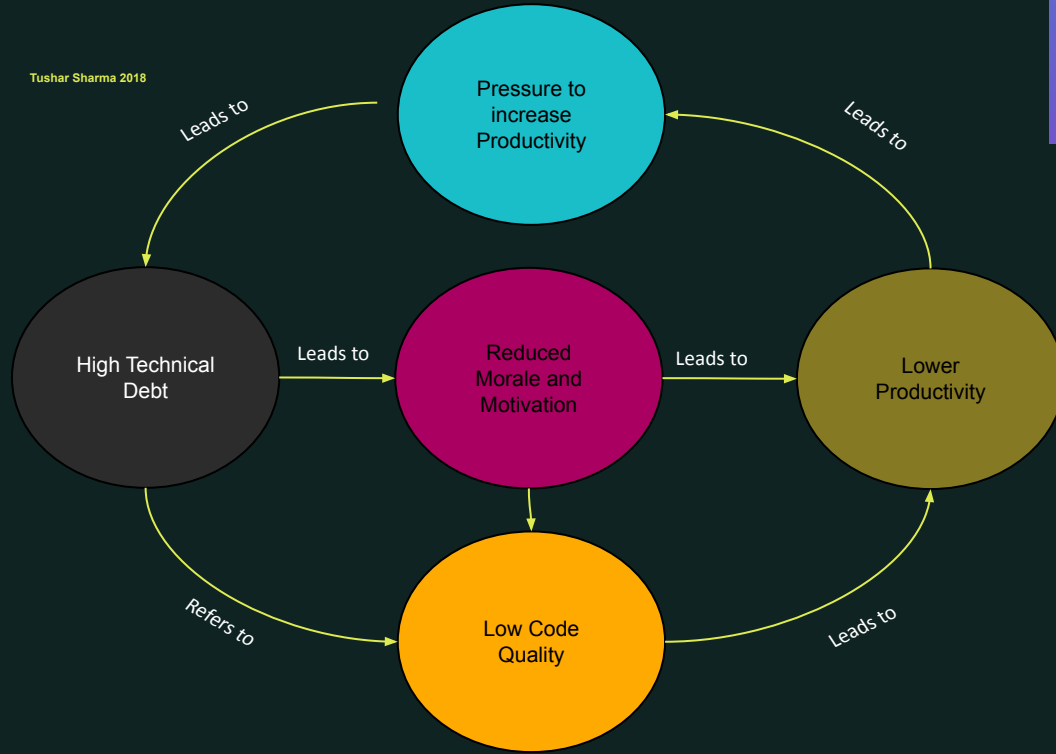


Engineering ~~Spiral~~ Cycle of productivity...

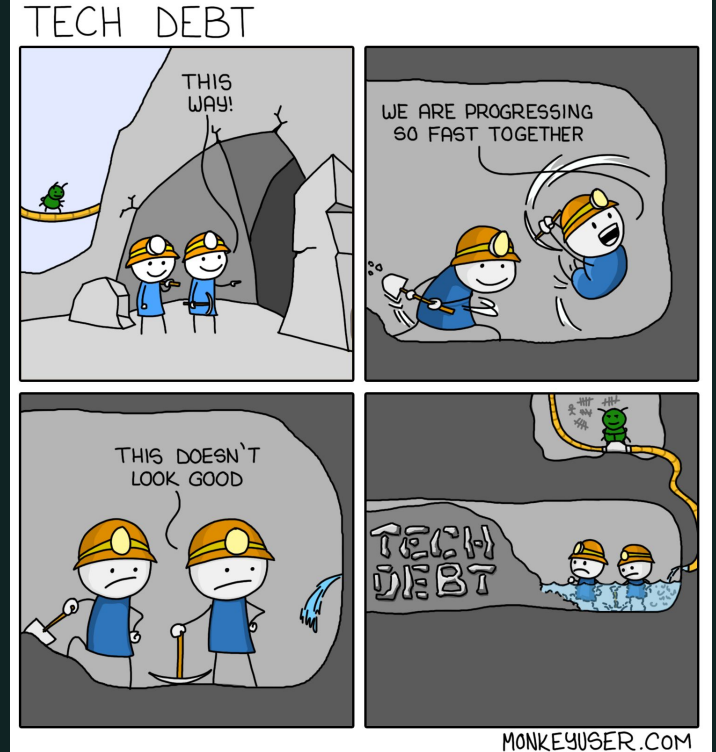


# So why bother thinking about Technical Debt?

Tushar Sharma 2018



Engineering ~~Spiral~~ Cycle of productivity...



# So why bother thinking about Technical Debt?

<i>Value</i>		<i>Visible</i>	<i>Invisible</i>
Positive Value	+	Features	Architecture & Infrastructure
Negative Value	-	Defects	Technical Debt 

Tech Debt - The Visible vs The Invisible

Black hole!  
Invisible, not easy to define any OKRs

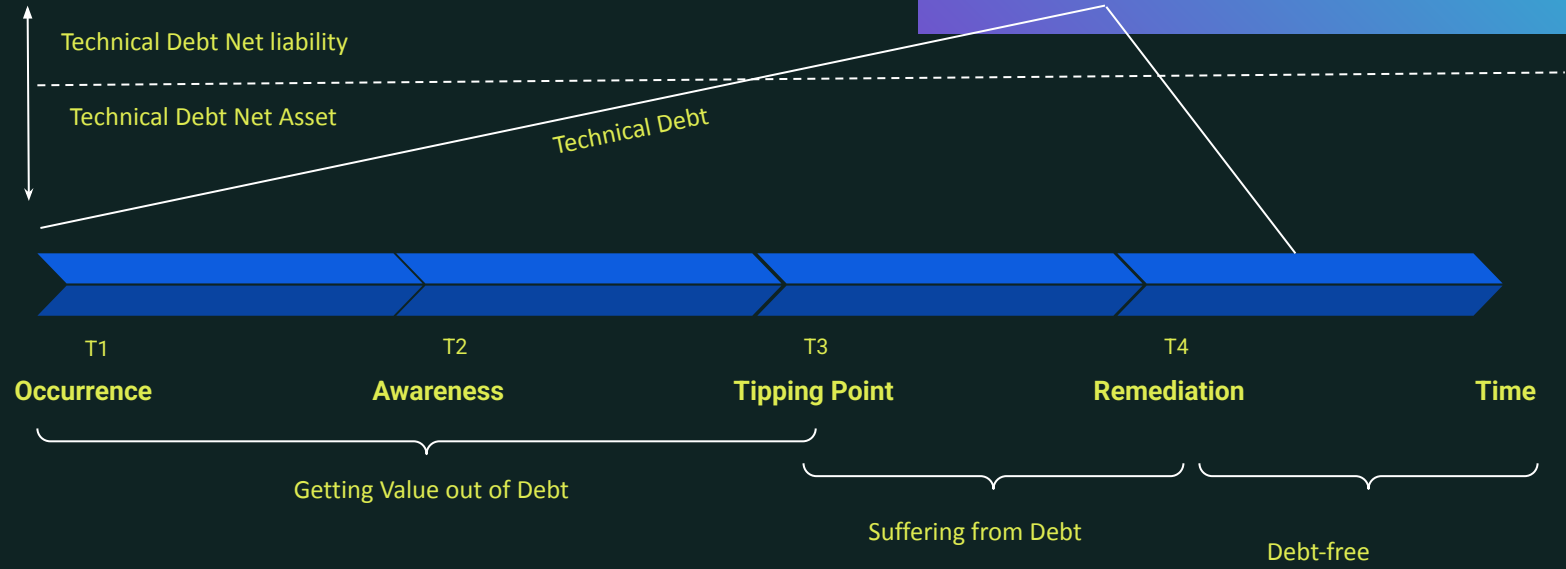
# So why bother thinking about Technical Debt?

<i>Value</i>		<i>Visible</i>	<i>Invisible</i>
Positive Value	+	Features	Architecture & Infrastructure
Negative Value	-	Defects	Technical Debt 

Tech Debt - The Visible vs The Invisible

Black hole!  
Invisible, not easy to define any OKRs

# So why bother thinking about Technical Debt?



Ipek Ozkaya 2019



## Our Approach to Recognizing Technical Debt

# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



Upto 25%!



Team Planning

# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



Upto 25%!



Team Planning



Prioritise

# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



Upto 25%!



Team Planning



Prioritise



Make it Visible

# Our Approach to Recognizing Technical Debt

Where can we start, What can we do today?



Upto 25%!



Team Planning



Prioritise



Make it Visible



Team Objective

# Our Approach to Recognizing Technical Debt

Tracing Tech Debt



# Our Approach to Recognizing Technical Debt

Tracing Tech Debt

Business Context



# Our Approach to Recognizing Technical Debt

## Tracing Tech Debt

Business Context → Assess Artifacts



# Our Approach to Recognizing Technical Debt

## Tracing Tech Debt

Business Context → Assess Artifacts → Record Debt Items



# Our Approach to Recognizing Technical Debt

## Tracing Tech Debt

Business Context → Assess Artifacts → Record Debt Item → Repeat



# Example Template!

<b>Name/Subject</b>	<b>(What)</b> - A standard Jira subject Ideally focused on shorthand “Technical Debt Item”
<b>Summary</b>	<b>(Where)</b> - “Where do we observe the technical debt in the affected development artifacts, and where do we expect it to accumulate?”
<b>Consequences</b>	<b>(Why)</b> - “Why is it important to address this technical debt item? Consequences include immediate benefits and costs as well as those that accumulate later, such as additional rework and testing costs as the issue stays in the system and costs due to reduced productivity, induced defects, or loss of quality incurred by building software that depends on an element of technical debt.”
<b>Remediation Approach</b>	<b>(When)</b> Describe the rework needed to eliminate the debt, if any. When should the remediation occur to reduce or eliminate the consequences?”
<b>Reporter / Owner / Team</b>	<b>(Who)</b> Who is responsible for servicing the debt? Assign a person or team. While in most cases the who aspect can be trivial, in some situations the debt resolution may need to be assigned to external parties. If remediation is significantly postponed, this field can communicate that decision.”

## Sub categories

In addition to the broader tag, a sub category can be useful to help teams identify hotspots or areas in our backlog that may later indicate a wider issues.

Suggested sub categories:

- **techdebt-design**
- **techdebt-coding**
- **techdebt-architecture**
- **techdebt-Infrastructure**
- **techdebt-accessibility**
- **techdebt-deprecated**
- **techdebt-testing**
- **techdebt-security**
- **techdebt-operations**
- **techdebt-deployment**

# Our Approach to Recognizing Technical Debt



The first step toward recognizing technical debt is to investigate the chain of **cause and effect in reverse**

Symptoms

# Our Approach to Recognizing Technical Debt



The first step toward recognizing technical debt is to investigate the chain of **cause and effect in reverse**

Symptoms



Consequences

# Our Approach to Recognizing Technical Debt



The first step toward recognizing technical debt is to investigate the chain of **cause and effect in reverse**

Symptoms



Consequences



Technical Debt

# Our Approach to Recognizing Technical Debt



The first step toward recognizing technical debt is to investigate the chain of **cause and effect in reverse**

**Symptoms**



**Consequences**



**Technical Debt**



**Causes**

# Business context and Factors to consider



## Size

### Factor

Size is a major factor, it can drive size of the teams, coordination, impact of changes, size of code and usually the larger the systems the more complexity is involved.



## Architecture

### Factor

Did we follow agreed patterns and due diligence when planning architecturally what we need? Technical debt for Architecture is difficult to identify and can be costly to repay.



## Business Model

### Factor

How is it funded? Is it internal only?, bespoke for an external dept?, a component of a larger system?, a commercial product?



## Team Distribution

### Factor

How many teams involved increase complexity and coordination, communication silos contribute to accumulation of debt.



## Rate of Change

### Factor

How stable is it? What are the risks and unknowns?, do incoming requirements exceed architecture limitations?



## Criticality

### Factor

Is Verification and validation needed constantly? This often leads to struggling with modernizing legacy or arcane services.



## Age of system

### Factor

No surprise here, more opportunities to accrue debt on large or long lived systems, Often carry hidden assumptions, constraints often prevent modernisation.



## Governance

### Factor

How do we make critical decisions? How do we steer? What do we do when things go wrong? What does success/failure look like?

# Business context and Factors to consider



## Size

### Factor

Size is a major factor, it can drive size of the teams, coordination, impact of changes, size of code and usually the larger the systems the more complexity is involved.



## Architecture

### Factor

Did we follow agreed patterns and due diligence when planning architecturally what we need? Technical debt for Architecture is difficult to identify and can be costly to repay.



## Business Model

### Factor

How is it funded? Is it internal only?, bespoke for an external dept?, a component of a larger system?, a commercial product?



## Team Distribution

### Factor

How many teams involved increase complexity and coordination, communication silos contribute to accumulation of debt.



## Rate of Change

### Factor

How stable is it? What are the risks and unknowns?, do incoming requirements exceed architecture limitations?



## Criticality

### Factor

Is Verification and validation needed constantly? This often leads to struggling with modernizing legacy or arcane services.



## Age of system

### Factor

No surprise here, more opportunities to accrue debt on large or long lived systems, Often carry hidden assumptions, constraints often prevent modernisation.



## Governance

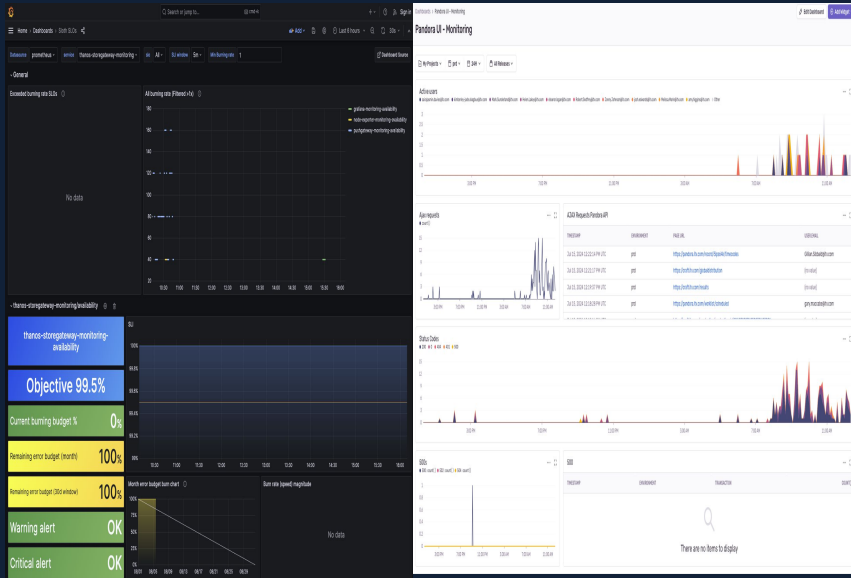
### Factor

How do we make critical decisions? How do we steer? What do we do when things go wrong? What does success/failure look like?

# Observability, Data-Driven Value & Benefits

## Telemetry Over "Gut-Feel"

- Spot hotspots instantly without relying on a single individual







# Observability, Data-Driven Value & Benefits

## Telemetry Over "Gut-Feel"

- Spot hotspots instantly without relying on a single individual

## Safe Decommissioning

- Confidently turn off features to shrink estate costs.

## Measurable Value

- Prove impact on both heritage systems and microservices.

## Predictable Scaling

- Remove guesswork around infrastructure capability.



# Culture, confidence & Debt

## Culture Over Tools

- Practices influence outcomes as much as technology does



# Culture, confidence & Debt

## Culture Over Tools

- Practices influence outcomes as much as technology does.

## Defining "Reliable Enough"

- Aligning engineering SLOs directly with product OKRs



# Culture, confidence & Debt

## Culture Over Tools

- Practices influence outcomes as much as technology does.

## Defining "Reliable Enough"

- Aligning engineering SLOs directly with product OKRs

## Technical Psychological Safety

- Building deployment confidence instead of codebase fear



# Culture, confidence & Debt

## Culture Over Tools

- Practices influence outcomes as much as technology does.

## Defining "Reliable Enough"

- Aligning engineering SLOs directly with product OKRs

## Technical Psychological Safety

- Building deployment confidence instead of codebase fear

## From Survival to Mastery

- Combining observability with ring-fenced capacity for true autonomy



# Domain View

## Platform

Linear Scheduling

Company  
Management

Compliance

Finance

Reporting

Rights

Catalogue

Modernisation  
tasks

Migration tasks

# Domain View

## Platform

Linear Scheduling

Company  
Management

Compliance

Finance

Reporting

Rights

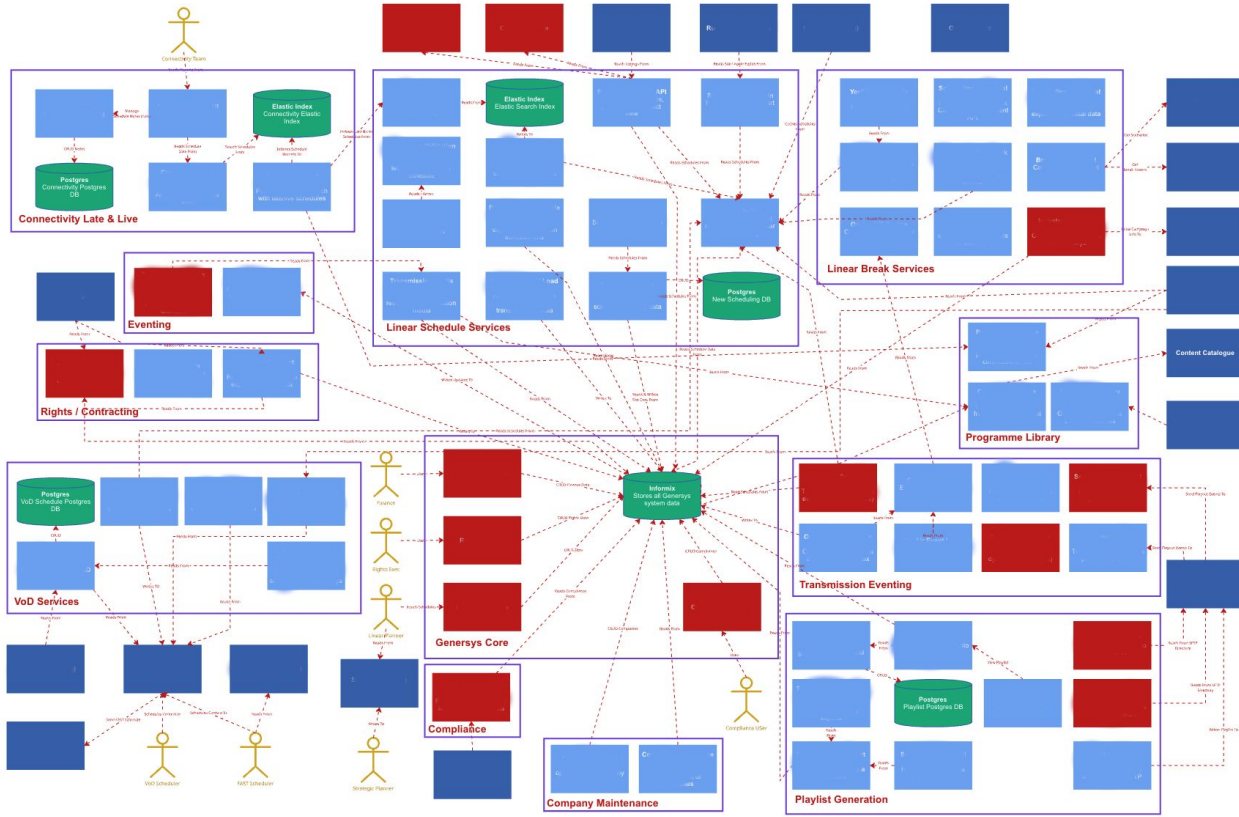
Catalogue

Modernisation  
tasks

Migration tasks

# Piece of Cake

Start of SM

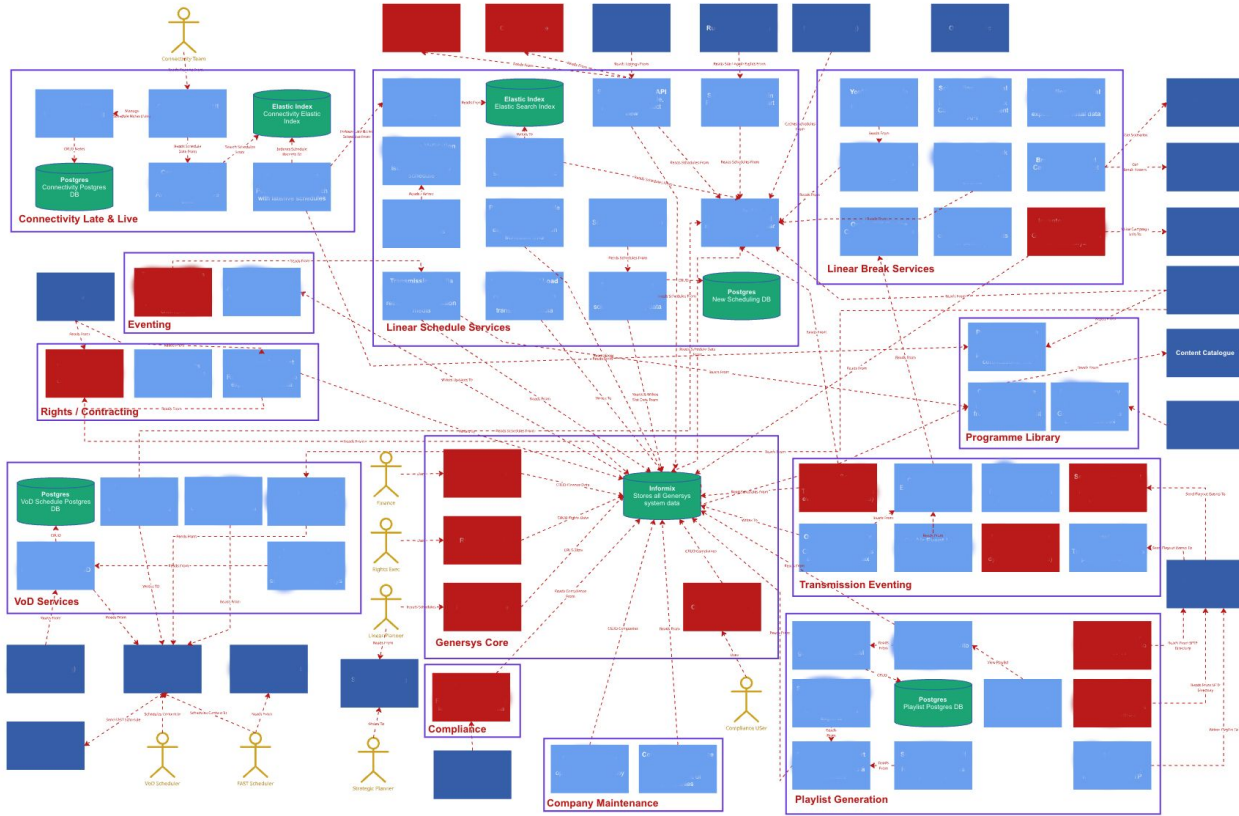


58 Services  
7 Databases

? procedures  
? function  
? raw lines of  
code

# Piece of Cake

Start of SM



58 Services  
7 Databases

297 procedures  
856 function  
257,896 raw  
lines of code

# Phased Modernisation

- Phase 1:  
The Sidecar UI (planning Grid)



# Phased Modernisation

- Phase 1  
The Sidecar UI (planning Grid)
- Phase 2 & 3  
Decoupling Core Logic  
(Forecasting & Planning)



# Phased Modernisation

- Phase 1  
The Sidecar UI (planning Grid)
- Phase 2 & 3  
Decoupling Core Logic  
(Forecasting & Planning)
- Phase 4 & 5  
Migrating Views (Presentation &  
Media)



# Phased Modernisation

- Phase 1  
The Sidecar UI (planning Grid)
- Phase 2 & 3  
Decoupling Core Logic  
(Forecasting & Planning)
- Phase 4 & 5  
Migrating Views (Presentation & Media)
- Decouple database (retire informix)





# Here is the old

Update Programme Stripe

VERA

Q	L	Lock	Date	Year	Wk	Day	Bill	Rep	Epis	Ver	Title	Prod No	Finance Ref(s)	Sub	Ind	Aud	Vis	Split	R
✓		Cost Log	02/12/2025	2025	49	Tue	2100	6/R	5	1	VITAL SIGNS	1/7314/0049H001	AC/09190/3-1	Y	Y				
✓		Cost Log	03/12/2025	2025	49	Wed	2100	6/R	6	1	THE WAY THE WIND ...	1/7314/0050H001	AC/09190/3-1	Y	Y				
✓		Cost Log	04/12/2025	2025	49	Thu	2100	25/R	1	2	THE GHOST POSITION	1/7314/0007H003	A/10417/3-2	Y	Y				
✓		Cost Log	06/12/2025	2025	49	Sat	2100	7/R	5	1	VITAL SIGNS	1/7314/0004H001	AC/09190/3-1	Y	Y				
✓		Cost Log	06/12/2025	2025	49	Sat	2300	7/R	6	1	THE WAY THE WIND ...	1/7314/0050H001	AC/09190/3-1	Y	Y				
✓		Cost Log	07/12/2025	2025	49	Sun	2200	26...	1	2	THE GHOST POSITION	1/7314/0007H003	A/10417/3-2	Y	Y				
✓		Cost Log	09/12/2025	2025	50	Tue	2000	22/R	2	2	SILENT VOICES	1/7314/0009H003	A/10417/3-2	Y	Y				
✓		Cost Log	10/12/2025	2025	50	Wed	2000	23/R	3	1	A CERTAIN SAMARIT...	1/7314/0010H001	A/10417/3-2	Y	Y				
✓		Cost Log	11/12/2025	2025	50	Thu	2000	22/R	4	1	SANDANCERS	1/7314/0008H001	A/10417/3-2	Y	Y				
✓		Cost Log	13/12/2025	2025	50	Sat	2100	23...	2	2	SILENT VOICES	1/7314/0009H003	A/10417/3-2	Y	Y				
✓		Cost Log	13/12/2025	2025	50	Sat	2300	24...	3	1	A CERTAIN SAMARIT...	1/7314/0010H001	A/10417/3-2	Y	Y				
✓		Cost Log	14/12/2025	2025	50	Sun	2200	23...	4	1	SANDANCERS	1/7314/0008H001	A/10417/3-2	Y	Y				
✓		Cost Log	16/12/2025	2025	51	Tue	2000	26/R	1	1	CASTLES IN THE AIR	1/7314/0012H001	A/10417/3-3	Y	Y				
✓		Cost Log	17/12/2025	2025	51	Wed	2000	25/R	2	2	POSTER CHILD	1/7314/0011H003	A/10417/3-3	Y	Y				
✓		Cost Log	18/12/2025	2025	51	Thu	2000	25/R	4	2	PRODIGAL SON	1/7314/0014H003	A/10417/3-3	Y	Y				
✓		Cost Log	20/12/2025	2025	51	Sat	2100	27...	1	1	CASTLES IN THE AIR	1/7314/0012H001	A/10417/3-3	Y	Y				
✓		Cost Log	21/12/2025	2025	51	Sun	2200	26...	2	2	POSTER CHILD	1/7314/0011H003	A/10417/3-3	Y	Y				
✓		Cost Log	22/12/2025	2025	52	Mon	2000	25/R	1	1	ON HARBOUR STREET	1/7314/0015H001	A/10417/3-4	Y	Y				
✓		Cost Log	22/12/2025	2025	52	Mon	2300	26...	4	2	PRODIGAL SON	1/7314/0014H003	A/10417/3-3	Y	Y				
✓		Cost Log	23/12/2025	2025	52	Tue	2000	24/R	2	4	PROTECTED	1/7314/0016H004	A/10417/3-4	Y	Y				
✓		Cost Log	24/12/2025	2025	52	Wed	2000	22/R	3	4	THE DEER HUNTERS	1/7314/0017H004	A/10417/3-4	Y	Y				
✓		Cost Log	25/12/2025	2025	52	Thu	2100	3/R	5	2	THE RISING TIDE - C...	1/7314/0055H006	AC/110296/3-1	Y	Y				
✓		Cost Log	26/12/2025	2025	52	Fri	2000	3/R	1	1	AGAINST THE TIDE	1/7314/0051H001	AC/110296/3-1	Y	Y				
✓		Cost Log	27/12/2025	2025	52	Sat	2000	3/R	2	1	FOR THE GRACE OF ...	1/7314/0052H001	AC/110296/3-1	Y	Y				
✓		Cost Log	28/12/2025	2025	52	Sun	2000	3/R	3	1	BLUIE	1/7314/0053H001	AC/110296/3-1	Y	Y				
✓		Cost Log	29/12/2025	2025	01	Mon	2030	3/R	4	1	THE DARKEST EVEN...	1/7314/0054H001	AC/110296/3-1	Y	Y				
✓		Cost Log	30/12/2025	2026	01	Tue	2030	4/R	1	1	FAST LOVE	1/7314/0057H001	AC/110926/3-1	Y	Y				
✓		Cost Log	31/12/2025	2026	01	Wed	2000	4/R	2	1	TENDER	1/7314/0058H001	AC/110926/3-1	Y	Y				

28 Event(s)

Validate Library Build State Save OK Cancel

Channel(s)	Finance Ref	Pack	Title	Ver	Epis	Rights
ITV3	A/110615/3	50	VERA (S7)	2017	4	01-Jan
ITV3	A/110296/3	2	VERA (S8)	2018	4	01-Jul
ITV3	A/10415/3	7	VERA (S9)	2017	4	31-Dec
ITV3	A/10615/3	49	VERA (S10)	2019	4	25-Nov
ITV3	AC/09190/3	1	VERA (S11)	2020	6	01-Feb
ITV3	AC/110296/3	1	VERA (S12)	2021	5	??-Feb

Standard Drop Mode Highlight Drop Mode Show Only VS Versions Hide VS Versions

Epis	Episode Title	Ver	Version Description	Cert	Prod No
1	WITNESS	2	VERSION 2 - 2000 - WITNESS	2000	1/7314/0045H005
2	RECOVERY	2	VERSION 2 - 2000 - RECOVE...	2000	1/7314/0046H004
3	TYGER TYGER	1	VERSION 1 - 2000 - TYGER...	2000	1/7314/0048H001
4	AS THE CROW FLIES	1	VERSION 1 - 1900 - EP 4 AS...	1900	1/7314/0047H001
5	VITAL SIGNS	1	VERSION 1 - 2000 - VITAL SI...	2000	1/7314/0049H001
6	THE WAY THE WIND BLOWS	1	VERSION 1 - 2000 - THE WA...	2000	1/7314/0050H001

6 Ep(s) x 6(s)

Black = SD, Orange = HD.





# Wrap up

- Unmanaged debt affects engineering productivity



# Wrap up

- Unmanaged debt affects engineering productivity
- COTS Illusion



# Wrap up

- Unmanaged debt affects engineering productivity
- COTS Illusion
- Extract the unrelated domain technology



# Wrap up

- Unmanaged debt affects engineering productivity
- COTS Illusion
- Extract the unrelated domain technology.
- Open stakeholder support



# Wrap up

- Unmanaged debt affects engineering productivity
- COTS Illusion
- Extract the unrelated domain technology.
- Open stakeholder support
- Product, Delivery and Engineering must plan debt together (not just a technology problem)



# Wrap up

- Unmanaged debt affects engineering productivity
- COTS Illusion
- Extract the unrelated domain technology.
- Open stakeholder support
- Product, Delivery and Engineering must plan debt together (not just a technology problem)
- Use Observability to provide better visibility and safety



THANK  
*you*