



ARCHITECTURE DECISION RECORDS

Una breve historia sobre ADRs

¿QUIÉN SOY?

Pepe Fabra Valverde

Qué hago, cuál es mi experiencia, cuándo y cuántas veces me he topado con este problema.

QUÉ QUIERO QUE APRENDAS

- La importancia de las decisiones, y de documentarlas
- ADR y ASR
- Fitness Function

PROBLEMA

Decisiones inferidas



BUGS

Si no conozco las decisiones, puedo aspirar a inferirlas del código. Cuando el negocio está de por medio, confiar en que se asuman las decisiones adecuadas es demasiado arriesgado

PRS INCONSISTENTES

- Code reviews frustrantes
- Sistema débil
- Negocio comprometido
- etc



MUNDO DE DECISIONES

En un mundo de decisiones, datos y prompts, no conocer con claridad la expectativa es muy dañino



NECESIDAD

Comprender y conocer las decisiones



RESPETAR EL SISTEMA

Respetar al equipo y al negocio



DOCUMENTAR DECISIONES

Pero desde una perspectiva de arquitectura técnica



SOLUCIÓN

Documentarlas, documentamos negocio, documentamos funcionalidades, documentamos soluciones técnicas.

¿Por qué no tenemos la costumbre de documentar decisiones (de arquitectura)?



ADR

Un ADR no es más que un estándar, un concepto, que promueve documentar decisiones de arquitectura con un registro fácil de navegar.

No solo decisiones actuales, también histórico. Esto es así porque...

REQUISITOS DEL SISTEMA

Un sistema puede tenerlo todo, pero siempre habrá cosas que **no** pueda obviar.

En un sistema de banca, el rendimiento y la seguridad son importantes, pero suelen ser excluyentes, si tuviésemos que elegir... **seguramente** tendríamos claro cuál.

Identificar requisitos → Software Architecture: The Hard Parts

ASR

Estos requisitos del sistema son identificados como Architecture Significant Requirement.

Algunos ejemplos (Mark Richards y Neal Fords los llaman los -ilities)

- Maintainability
- Testability
- Deployability
- Security
- Performance
- Scalability

FORMATO DE ADRS

- Estado
 - proposed, accepted, rejected, deprecated, superseded
- Contexto
- Decisión
- Consecuencias
- Recursos

TOOLING DE ADRS

- CLIs
- Herramientas webs
- Herramientas internas
- Wikis y plantillas

QUÉ PUEDE SER UN ADR

- Protocolos de comunicación
- Usar Java para todos los micros
- Composición > Herencia
- Ante X situación usaremos/haremos Y



FITNESS FUNCTIONS

Para cerrar, una breve introducción

O'REILLY®



Fundamentals of Software Architecture

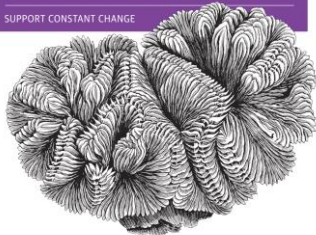
An Engineering Approach

Mark Richards & Neal Ford

O'REILLY®

Building Evolutionary Architectures

SUPPORT CONSTANT CHANGE

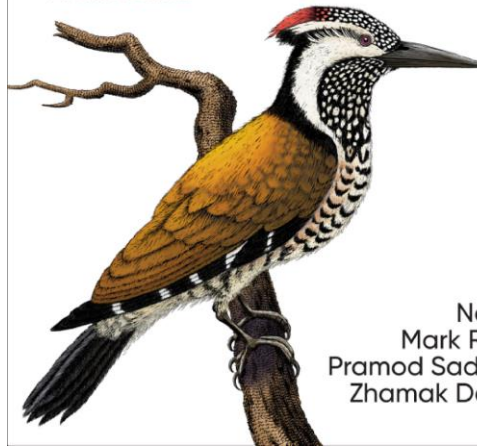


Neal Ford, Rebecca Parsons & Patrick Kua

O'REILLY®

Software Architecture: The Hard Parts

Modern Trade-Off Analyses for Distributed Architectures



Neal Ford,
Mark Richards,
Pramod Sadalage &
Zhamak Dehghani

RECOMENDACIONES
DE LIBROS

BIBLIOGRAFÍA

- <https://adr.github.io/>
- Fundamentals of Software Architecture

CÓMO PROPONÉRSELO A UN CLIENTE

- Cultura de empresa
- Impacto en el negocio y sus productos
- Impacto en la Experiencia de Desarrollo



PREGUNTAS