

# Custom IT Budgeting Best Practices

Waldo Jaquith <[waldo@usdigitalresponse.org](mailto:waldo@usdigitalresponse.org)>

November 2025



# **Case Study: Rhode Island Unified Health Infrastructure Project**



**UHIP debacle: R.I. to extend contract, as Deloitte agrees to more concessions**

Continuing UHIP debacle named R. I. Story of the Year

**RI awards Deloitte \$99 million contract to keep running UHIP**

**Childcare centers say R.I. computer glitches are costly**

**‘We are very sorry’: Deloitte apologizes to RI about UHIP**



# **UHIP repair cost grows**

*State must pay an additional \$30M to solve the computer system problems*

**U.S. fines R.I. \$2 million over UHIP payment errors**

**ACLU sues Rhode Island over computer benefits system delays**

Of all government software development contracts over \$6 million, only 11% are successful (cost, schedule, performance).

46% of systems developed across \$35 billion worth of DoD spending failed to meet real needs even though they met written, contractual specifications.



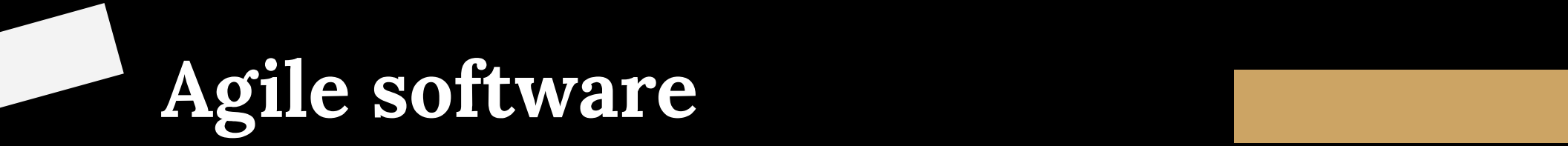
One study of 400 projects found that only 10% of traditionally-developed code was ever actually deployed. Only 2% was ever used.

The average government IT project costs **310%** of the originally estimated price.



There is a better way.



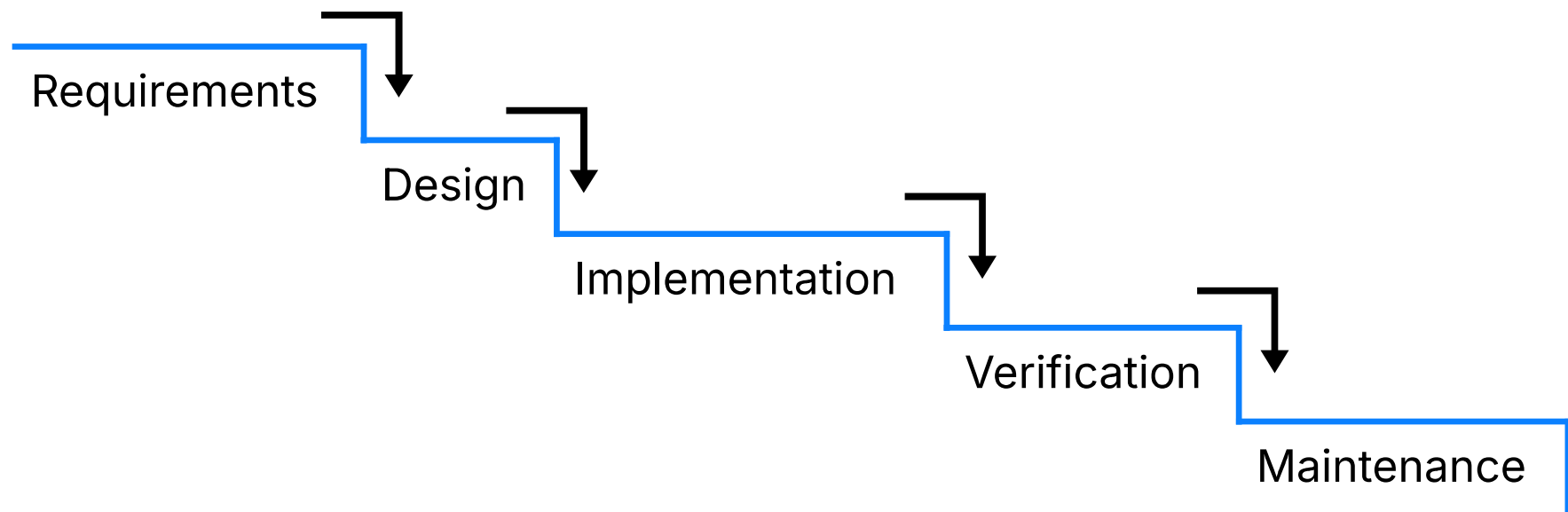


# Agile software development

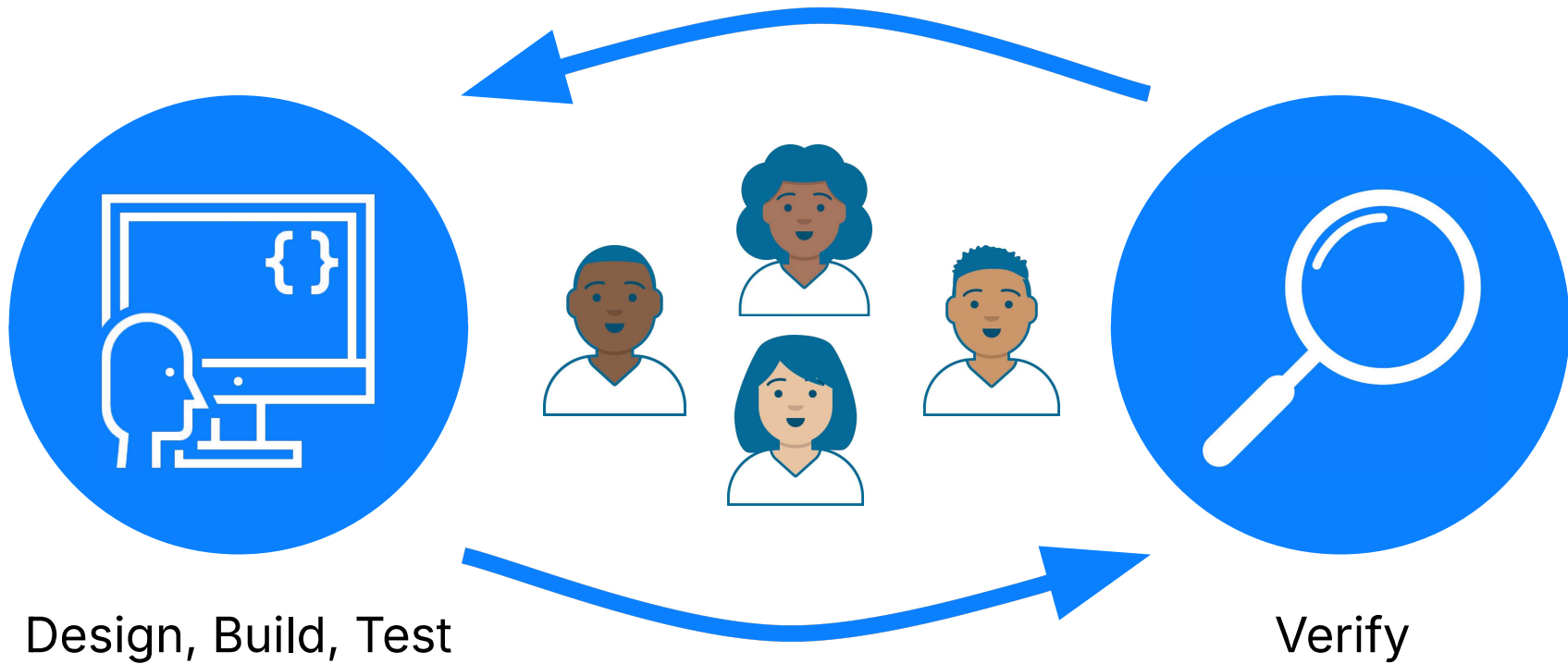
Building software with less  
risk



# Waterfall development model



# The premise of Agile





The Take-Away

**Agile facilitates *extraordinarily* tight contract oversight of vendor activities and ensures that you're getting your money's worth. It's how the whole software industry works.**



# Product ownership

Taking control of  
procurement





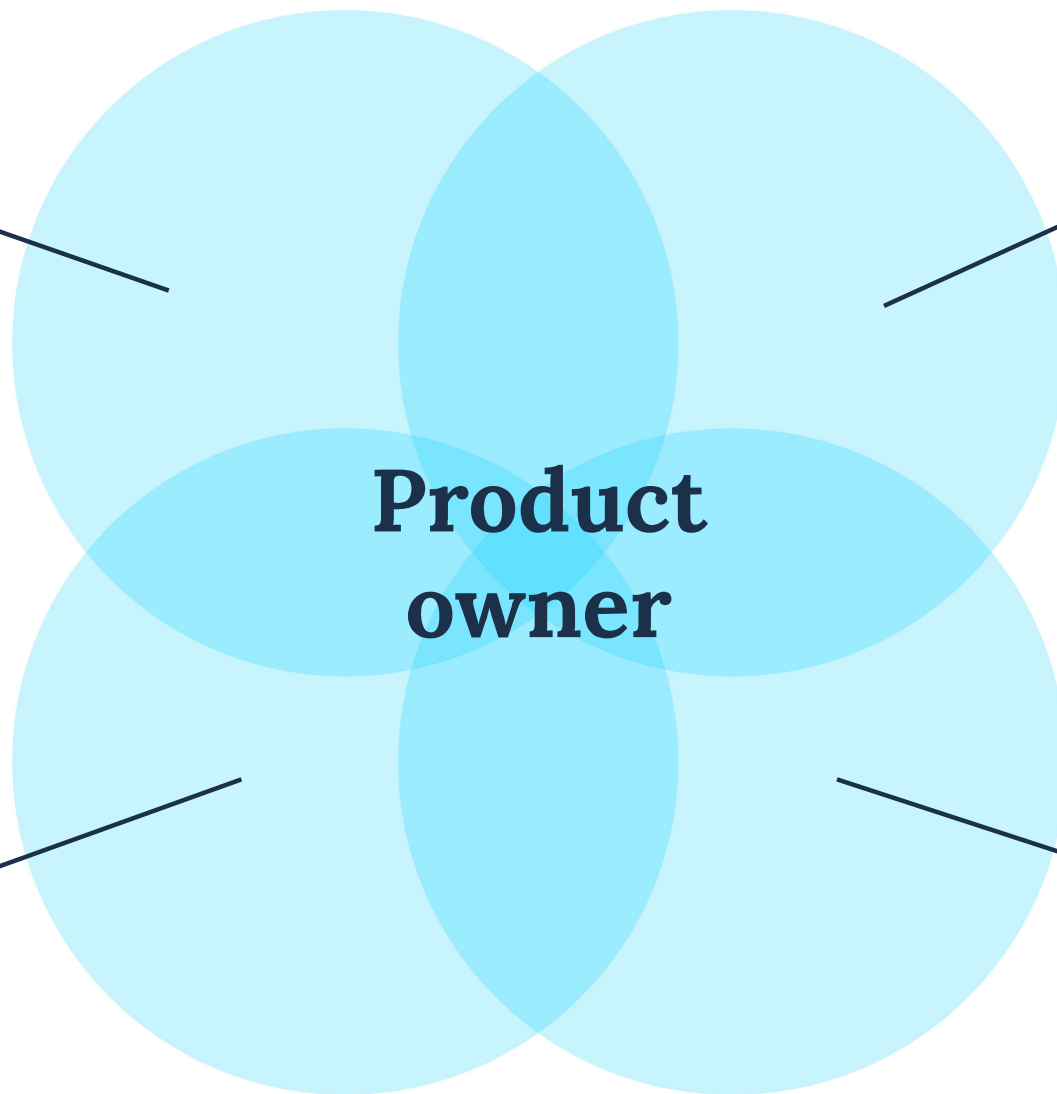
**Users**

**Stakeholders**

**Product  
owner**

**Technology**


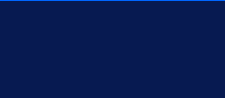
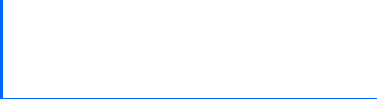
**Business**





### The Take-Away

The product owner is the fulcrum on which the success of the project hinges, the keeper of all system knowledge, the person who knows in exhaustive detail exactly what the vendor is doing.





# User-centered design

Helping us understand where  
to go and why



**“Design is deciding how a thing  
should be.”**

—William Van Hecke

**User-centered design helps us  
design for actual humans **who  
will use what we're making.****





The Take-Away

User research ensures that the vendor is doing the right work and that the system will do what you need it to do.



# **Agile acquisition**

Structuring contracting to  
support development





There are **two principles** to enforce to get better outcomes.

# 1

Keep contract increments **under \$10 million.**



Cost millions	< \$0.5	\$0.5-3	\$3-6	\$6-10	> \$10
Success	61%	24%	12%	11%	6%

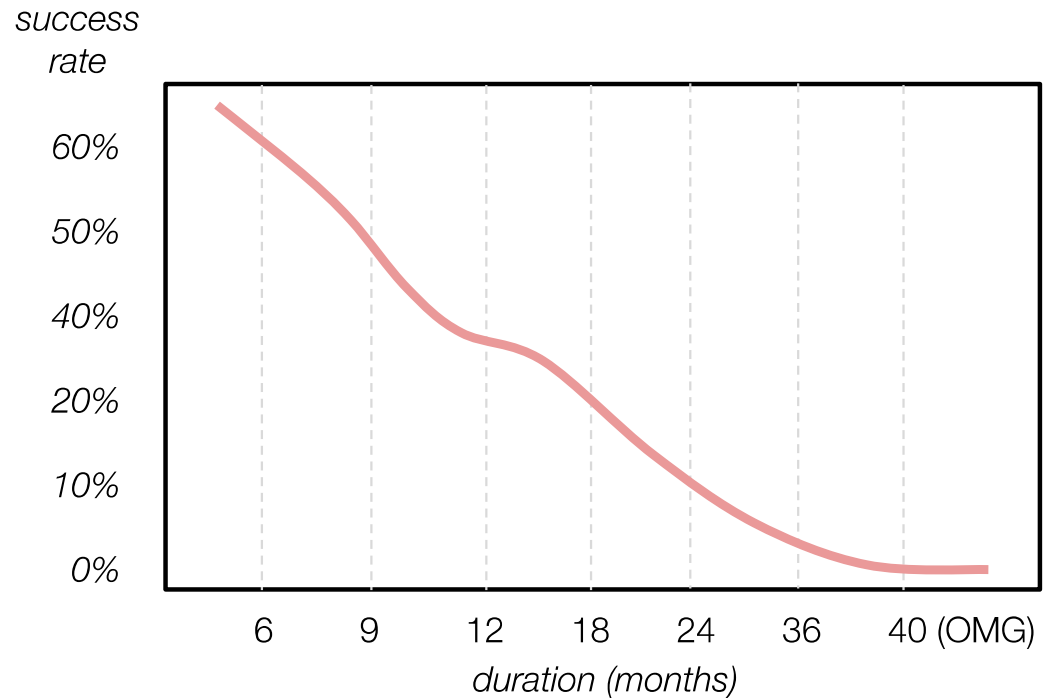
The more you spend, the greater your odds of failure.

[Source:](#) "CHAOS Report 2015," Standish Group

# 2

Keep periods of performance short.

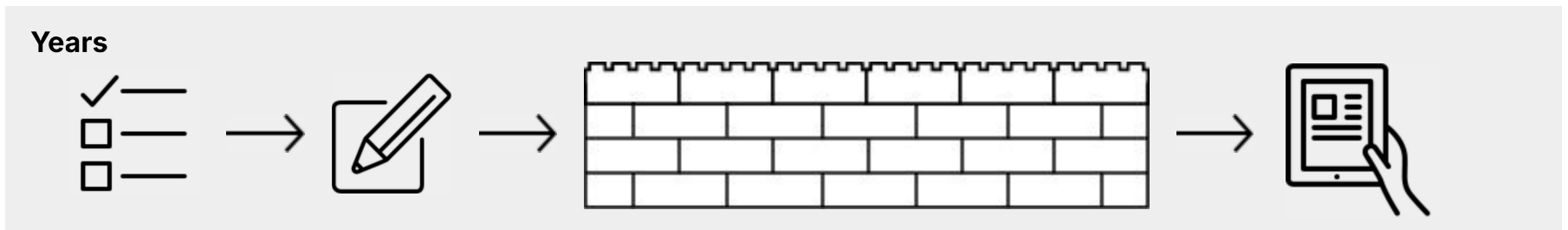
No longer than **3 calendar years** (including options).



Standish Group Study of 23,000 projects comparing project success versus duration.

# Strategy

# The traditional way of approaching “big bang” procurements



# Traditional contracting

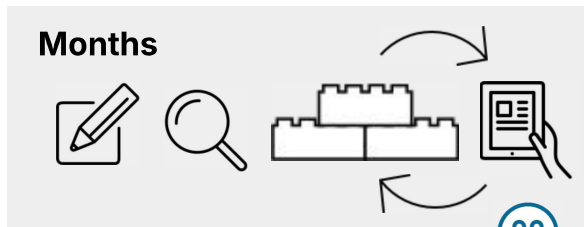
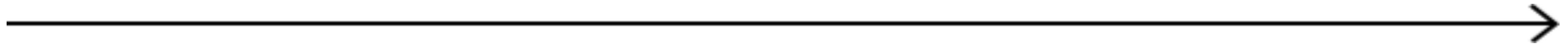
## ATTRIBUTES

- Expensive
- Prescriptive
- Time-consuming
- Waterfall development

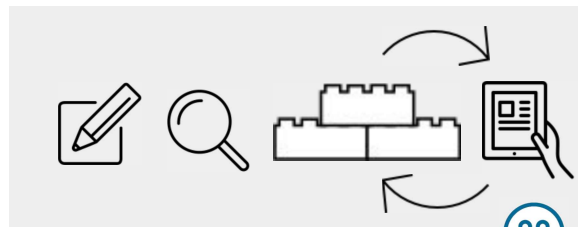
## OUTCOMES

- Risk
- No user value for a long time
- Contract mods are common
- All knowledge is with the vendor

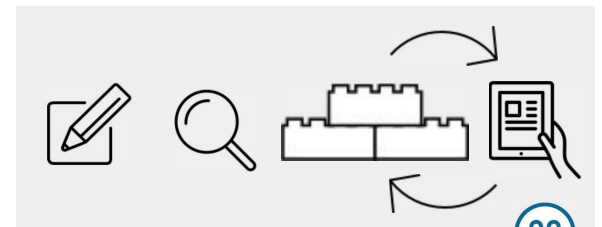
# Agile acquisition



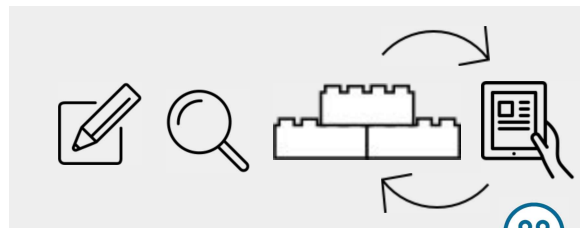
**contract module**



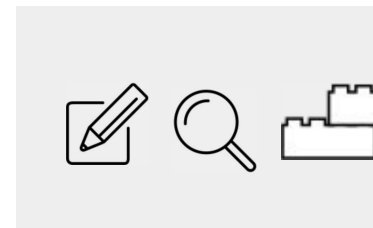
**contract module**



**contract module**



**contract module**



**contract module**

# Agile contracting

## ATTRIBUTES

- Less expensive
- Flexible
- Descriptive
- Quick acquisitions
- Agile development

## OUTCOMES

- Less risk
- Frequent delivery of value to users
- Fewer/no mods
- Government isn't dependent on a single vendor





# What you can do

Legislative actions to improve  
the status quo



# How to fail at software projects

- Don't conduct user research
- Define lots of requirements up front
- Sign a \$50M contract with a single vendor
- Conduct oversight via reports, not live software demos
- Punish projects that fail fast

# Recommended actions

1. Require twice-monthly demos to legislative staff

# Recommended actions

1. Require twice-monthly demos to legislative staff
2. Establish a state digital service

# Recommended actions

1. Require twice-monthly demos to legislative staff
2. Establish a state digital service
3. Allocate funding operationally



In conclusion

**If the technology fails, the  
legislation fails. So don't let the  
technology fail.**

